Analyse du discours conversationnel dans le cadre de communications médiées par ordinateur

Jeremy Auguste

▶ **To cite this version:**

Jeremy Auguste. Analyse du discours conversationnel dans le cadre de communications médiées par ordinateur. Informatique et langage [cs.CL]. Aix-Marseille Université, 2020. Français. NNT : . tel-03356464

HAL Id: tel-03356464
https://theses.hal.science/tel-03356464

Submitted on 28 Sep 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

AIX-MARSEILLE UNIVERSITÉ
ECOLE DOCTORALE 184
LABORATOIRE D'INFORMATIQUE ET SYSTÈMES

Thèse présentée pour obtenir le grade universitaire de docteur

Discipline : Informatique

Jeremy AUGUSTE

Analyse du discours conversationnel dans le cadre de communications médiées par ordinateur

Conversational discourse parsing of chat conversations

Soutenue le 14/10/2020 devant le jury composé de :

Benoît CRABBÉ	LLF, Université de Paris	Rapporteur
Yannick ESTÈVE	LIA, Université d'Avignon	Rapporteur
Christophe CERISARA	LORIA, CNRS	Examinateur
Géraldine DAMNATI	Orange Labs, Lannion	Examinatrice
Sophie ROSSET	LIMSI, CNRS	Examinatrice
Frédéric BÉCHET	LIS, Aix-Marseille Université	Directeur de thèse
Alexis NASR	LIS, Aix-Marseille Université	Directeur de thèse

Numéro national de thèse/suffixe local : 2020AIXM0228/011ED184

Cette œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 4.0 International.

Résumé

Les dialogues ont une place importante dans la société et celle-ci s'accroît au fur et à mesure que la technologie progresse. Il existe de plus en plus d'outils pour dialoguer à distance permettant la collecte d'une masse importante de données, utilisables pour réaliser différentes analyses et divers systèmes automatiques.

L'analyse du discours conversationnel est une réponse partielle pour comprendre certains aspects de la production du langage dans les dialogues. Une telle analyse permet de caractériser les interactions entre les messages d'un dialogue et ainsi faire ressortir les différents enjeux ou d'identifier les échanges nécessaires pour faire progresser le dialogue.

Produire ces analyses est une tâche complexe. Le nombre important de théories d'analyse du discours illustre bien la complexité pour un humain à définir des structures discursives modélisant l'ensemble des interactions. Ceci rend la production d'un grand corpus annoté très couteuse et le peu de données annotées rend difficile l'utilisation d'algorithmes d'apprentissage supervisés.

Dans cette thèse, je propose de produire des représentations du discours conversationnel en s'appuyant sur peu de données annotées discursivement. La thèse s'inscrit dans le cadre de l'ANR DATCHA me donnant accès à un grand corpus de tchats provenant de l'entreprise Orange. Ce corpus me permet d'explorer plusieurs stratégies pour produire des représentations du discours : s'appuyer sur un modèle bout-en-bout prédisant la satisfaction des clients ; se fonder sur des annotations en actes de dialogue pour produire des plongements de phrases ; utiliser des algorithmes supervisés sur un corpus enrichi automatiquement.

Mots-clefs : analyse du discours, dialogues, tchats, représentations distributionnelles, structures dialogiques, actes de dialogue.

Abstract

Dialogues are a central part of human society, and technological improvements only strengthen their use in more and more situations. Additional tools used to communicate from a distance allow the collection of large amounts of data, which can be used to produce various analyses and automatic systems.

Conversational discourse analysis is a partial response to understand some aspects of language production in dialogues. It is used to characterize the different interactions between the messages of a dialogue, and thus highlight the different issues or identify the exchanges that are needed to solve the dialogue's main objectives.

Discourse parsing is a challenging task. The high number of existing theories of discourse analysis shows that humans have a hard time defining discursive structures that model all possible interactions. This difficulty makes the production of annotated corpora expensive and the low amount of discursively annotated data makes the use of supervised learning algorithms impractical.

In this thesis, I propose to produce representations of conversational discourse based on data that is partially annotated with discourse structures. The thesis is part of the DATCHA project which allowed me access to a large corpus of dialogues owned by the Orange company. This corpus allows us to explore different strategies in order to produce discourse representations: rely on an end-to-end model that predicts customer satisfaction; rely on dialogue acts to produce sentence embeddings; using supervised algorithms on an automatically enriched corpus.

Keywords: discourse analysis, dialogues, written chats, distributional representations, dialogical structures, dialogue acts.

Remerciements

Il y a treize ans — au collège — j'avais pris la grande décision (à l'époque c'était important en tout cas !) de ne plus chercher à devenir astrophysicien car cela m'aurait imposé de faire de longues études. Effectivement, je ne suis pas devenu astrophysicien... mais me voilà aujourd'hui — treize ans plus tard — à la fin de mon doctorat. Il est clair que sans les nombreuses rencontres que j'ai pu faire durant ces nombreuses années, mon cursus aurait été très différent (et peut-être un peu moins long !). Il me parait donc important de remercier les différentes personnes ayant contribuées à modeler celui-ci.

Tout d'abord, j'aimerais remercier mes directeurs de thèse, Alexis Nasr et Frédéric Béchet, pour m'avoir montré, dès le M1, ce qu'est le monde merveilleux du TAL et pour m'avoir suivi durant ces quatre années de thèse. J'aimerais aussi remercier Benoit Favre qui, même s'il n'a pas été officiellement mon directeur, m'a également suivi de près. Leurs conseils m'ont été précieux — leurs complémentarités fait que j'avais toujours quelqu'un vers qui me tourner — et j'ai passé des moments inoubliables en leur compagnie lors des différentes conférences et autres déplacements.

Je remercie Yannick Estève et Benoît Crabbé qui ont accepté d'être les rapporteurs de ce manuscrit, mais aussi Géraldine Damnati, Sophie Rosset et Christophe Cerisara pour avoir accepté de faire partie de mon jury de thèse.

Je remercie également Georges Linares et Carlos Ramisch pour avoir été les membres de mon comité de suivi de thèse. Leurs remarques et conseils lors des comités annuels m'ont été indispensables afin d'améliorer la qualité de mes travaux lors de ces quatre années.

Je remercie Géraldine Damnati et Delphine Charlet pour les nombreuses collaborations qu'on a pu avoir durant ma thèse. Celles-ci m'ont été très enrichissantes et j'espère avoir de nouveau l'occasion de travailler avec vous deux dans le futur.

Je souhaite également remercier les différentes personnes que j'ai pu cotoyer durant ces quatres années dans les locaux du laboratoire. Je pense en particulier à l'ensemble de l'équipe TALEP qui permet à chacun de s'épanouïr dans sa recherche dans la bonne humeur, au personnel administratif du LIS qui est toujours présent pour aider les doctorants en détresse, ou encore aux personnes présentes dans l'appartement durant les nombreuses mais indispensables pauses avec qui j'ai passé d'incroyables moments : Makki, Jérémy, Jérémie, José-Luis, Thibault, Damien, Florian, Franck, Cindy, Raph, Théodore, Amélia, Manon « la jeune », Matthieu, Jean-Marc, Arnaud, Benjamin, et tous les autres.

Parmi « les autres », certains ont une place particulière, car ils ont eu la lourde

et difficile tâche d'être mes co-bureaux : Olivier, Mickaël, les deux Sébastien, Cédric et Manon « la vieille ». Sans vous, mes journées n'auraient pas été aussi enrichissantes, passionnantes et amusantes !

Bien évidemment, je souhaite remercier ma famille et mes amis qui ont été là pour me soutenir, m'encourager ou pour simplement s'amuser. Je pense en particulier à Maman et Papa, Lucas et Matthieu, Gran et Bobby, mes oncles, mes tantes, mes cousins et ma belle-famille, qu'ils soient en France, en Écosse, en Angleterre, en Australie ou en Floride, mais aussi à Hadrian, Arnaud, Eddy, Amandine, Caroline et Anthony.

Et enfin, je n'oublie pas les deux personnes les plus importantes à mes yeux. Tout d'abord, Tiffen qui, même avec la distance (on peut entendre un « échoooo » dû à la distance !), est toujours là pour me soutenir et passer des moments inoubliables avec moi. Et enfin Alexie, avec qui je partage ma vie depuis presque sept ans, je te remercie pour tous ce que tu as pu m'apporter en soutien, en amour, en complicité et dans tous les aspects de ma vie en général.

Table des matières

Résumé 3

Abstract 4

Remerciements 5

Table des matières 7

Table des figures 11

Liste des tableaux 13

Liste des acronymes 15

Introduction générale 17

I Dialogues et modélisation du discours 26

1 Le dialogue et l'analyse du discours 27

1.1 Introduction . . . 27
1.2 Le dialogue . . . 29
1.2.1 Définitions et vocabulaire . . . 29
1.2.2 Différents types de dialogues . . . 30
1.3 L'analyse de surface du discours conversationnel . . . 34
1.3.1 Théorie des actes de langage . . . 35
1.3.2 Les actes de dialogue . . . 37
1.3.3 Étiquetage automatique en actes de dialogue . . . 43
1.3.4 Conclusion . . . 45
1.4 Structure locale du dialogue . . . 46
1.4.1 Paires adjacentes . . . 46
1.4.2 Actes de la conversation . . . 47
1.5 Analyse profonde du discours conversationnel . . . 49
1.5.1 L'analyse du discours dans le cadre de monologues . . . 49
1.5.2 Schéma d'annotation fondé sur la SDRT : le corpus STAC . . . 52
1.5.3 Schéma d'annotation fondé sur le PDTB . . . 54
1.5.4 Construction d'une structure hiérarchique . . . 55

1.6 Discussion 56

2 Apprentissage de représentations 58
2.1 Introduction 58
2.2 Du mot au plongement de mots 60
2.3 Amélioration des plongements de mots 64
2.3.1 Les mots hors vocabulaire 65
2.3.2 Des solutions pour la polysémie 66
2.4 Les plongements pour d'autres types d'unités linguistiques 68
2.5 Propriétés attendues et évaluation des représentations 70
2.5.1 Évaluation des plongements de mots 71
2.5.2 Évaluation des plongements de phrases 76
2.6 Discussion 78

II Représenter implicitement le discours à partir d'un corpus de tchats 79

3 Corpus Datcha : description et annotations de départ 80
3.1 Introduction 80
3.2 DATCHA : un corpus de tchats d'assistance clientèle 81
3.2.1 Le corpus DATCHA 81
3.2.2 Descriptions des différents sous-corpus 83
3.3 Comparaison avec des conversations orales 85
3.3.1 La silhouette des dialogues 86
3.3.2 Les erreurs dans le langage 87
3.4 Annotation en actes de dialogue 88
3.4.1 Le jeu d'étiquettes utilisé 88
3.4.2 Le sous-ensemble du corpus annoté manuellement 92
3.5 Étiquetage automatique en actes de dialogue 93
3.6 Discussion 94

4 Influence du langage tchat sur des tâches de TAL 96
4.1 Introduction 96
4.2 Corrections manuelles du corpus 98
4.3 Correction automatique du corpus 99
4.3.1 Définir le lexique de mots corrects 101
4.3.2 Correction automatique basées sur des distances d'édition . 101
4.3.3 Utilisation de plongements de mots pour réordonner les corrections 102
4.4 Étiqueteur en parties du discours sur les données DATCHA 103
4.5 Évaluation de l'influence des erreurs 104
4.6 Discussion 109

5 Prédire la qualité des interactions avec une méthodologie bout en bout 111
5.1 Introduction . . . 111
5.2 Des questionnaires de satisfaction utilisés comme supervision . . . 113
5.2.1 Liens entre les différentes questions . . . 114
5.2.2 Le *Net Promoter Score* . . . 115
5.3 La satisfaction client comme support pour des modèles bout en bout 116
5.3.1 Estimation de la difficulté de la tâche . . . 118
5.3.2 Étude de la nature des erreurs de prédiction . . . 125
5.3.3 Limiter les confusions entre classes extrêmes . . . 127
5.3.4 Conclusion . . . 134
5.4 Influence du contenu du dialogue sur les prédictions . . . 135
5.4.1 Influence des scripteurs . . . 135
5.4.2 Étude de l'importance du lexique . . . 137
5.5 Conclusion . . . 139

III Étudier les interactions à partir de représentations explicites du discours 142

6 Représentation vectorielle des tours de parole 143
6.1 Introduction . . . 143
6.2 Étude de la qualité des plongements de phrases sur des tâches de discours conversationnel . . . 144
6.2.1 Les plongements de phrases et les dialogues . . . 145
6.2.2 Entrainement de plongements de phrases . . . 146
6.2.3 Évaluation des plongements de phrases . . . 147
6.3 Prendre en compte explicitement les interactions dans les représentations . . . 151
6.3.1 Apprentissage de plongements de phrases reposant sur les actes de dialogue . . . 152
6.3.2 Évaluation des plongements de phrases . . . 155
6.4 Conclusion . . . 163

7 Analyse profonde du discours conversationnel avec peu d'annotations 166
7.1 Introduction . . . 166
7.2 Annotation du discours conversationnel en dépendance . . . 168
7.2.1 Spécificités du corpus DATCHA . . . 168
7.2.2 Schéma d'annotation utilisé . . . 169
7.2.3 Annotation du corpus DATCHA . . . 176
7.3 Entrainer un analyseur du discours avec peu de données . . . 176
7.3.1 Induire une grammaire hors-contexte . . . 178
7.3.2 Apprendre une grammaire hors-contexte probabiliste . . . 186

7.3.3 Induire les arbres discursifs à partir d'une PCFG 188
7.3.4 Expérimentations 189
7.4 Utiliser davantage de données à l'aide d'annotations automatiques . 195
7.4.1 Résoudre les problèmes de couverture des PCFG 196
7.4.2 Prise en compte du lexique par un analyseur en dépendance 197
7.4.3 Expérimentations 199
7.5 Conclusion 206

Conclusion générale **208**

Bibliographie **212**

Index **229**

ANNEXES **232**
A Exemple de dialogue avec annotations 233
B Lexique LEX5 234
C Lexique LEX6 235
D Caractéristiques utilisées par l'analyseur en transition 236

Table des figures

1.1 Exemple d'un dialogue . . . 30
1.2 Fonctions de communication prospectives du schéma d'annotation DAMSL . . . 41
1.3 Fonctions de communication rétrospectives du schéma d'annotation DAMSL . . . 41
1.4 Caractéristiques de contenu et de forme des énoncés du schéma d'annotation DAMSL . . . 42
1.5 Exemple d'arbre pouvant servir à représenter un dialogue guidé par une tâche . . . 56

2.1 Idées du fonctionnement des modèles CBOW et SKIP-GRAM . . . 64

3.1 Illustration du phénomène d'enchevêtrement de tours de parole . . . 83
3.2 Distribution des différents types de canaux d'assistance clientèle dans DATCHAFÉVRIER . . . 85
3.3 Récapitulatif des différentes variantes du corpus DATCHA . . . 86
3.4 Distribution des actes de dialogue dans le sous-ensemble d'apprentissage du corpus DATCHAACT . . . 93

5.1 Corrélation de Spearman entre les réponses données aux différentes questions . . . 115
5.2 Les différents schémas de classification utilisés . . . 132
5.3 Performances (MacroF1) des SVM, CNN et LSTM en fonction de la taille du lexique. . . . 139

6.1 Schémas des RNN utilisés pour l'évaluation des plongements de phrases . . . 148
6.2 Les différentes tâches de support utilisées pour construire des plongements de phrases prenant en compte le contexte dialogique . . . 155
6.3 Schéma du réseau utilisé pour apprendre les plongements de phrase prenant en compte le discours conversationnel. . . . 156
6.4 Schéma du réseau utilisé pour apprendre les vecteurs Skip-Act. . . . 157
6.5 Scores F1 obtenus par les différents plongements sur la tâche d'évaluation Acte Courant . . . 160
6.6 Scores F1 obtenus par les différents plongements sur la tâche d'évaluation Acte Suivant . . . 161

7.1 Exemple d'analyse syntagmatique . . . 177

7.2 Exemple d'analyse en dépendance 177
7.3 Exemple d'un arbre en dépendance discursif projectif tel qu'il peut être trouvé dans le corpus DATCHAREL 180
7.4 Exemple d'un sous-arbre en dépendance discursif non-projectif . . . 180
7.5 Exemple d'un sous-arbre en dépendance discursif projectivisé 181
7.6 Arbre syntagmatique équivalent à l'arbre en dépendance de la figure 7.3 182
7.7 Resultat de la binarisation de l'arbre de la figure 7.6 183
7.8 Contexte capturé (en rouge) par les règles (5) des tables 7.3 et 7.4 permettants de produire les sous-arbres des séquences $\texttt{PRO}_2 \ldots \texttt{STA}_5$ 187
7.9 Évolution de la couverture et des scores de rattachements en fonction de la taille du corpus de développement 194
7.10 Score de précision des différents analyseurs discursifs par type de relation discursive 204
7.11 Score de rappel des différents analyseurs discursifs par type de relation discursive 205

Liste des tableaux

1.1 Exemples d'énoncés associées à leurs actes de langage illocutoires . 36
1.2 Sous-ensemble de la taxinomie d'actes de dialogue du projet VERBMOBIL 39
1.3 Performances des modèles états de l'art sur la tâche de prédiction des actes de dialogue 45

2.1 Corrélations de Spearman entre les temps de réaction provenant du jeux de données SPP et les jugements d'association/similarité provenant d'autres jeux de données. 74
2.2 Évaluation sur le jeu de données SPP de plongements de mots en libre accès 75

3.1 Analyse de la forme des dialogues écrits et oraux 87
3.2 Jeu d'étiquettes en actes de dialogue utilisé pour DATCHA 89

4.1 Typologie des différentes erreurs possibles sur les mots 99
4.2 Le jeu d'étiquettes utilisé pour l'annotation en parties du discours ainsi que la proportion de chaque étiquette dans le corpus 100
4.3 Exactitudes (en %) des prédictions des parties du discours des différents étiqueteurs 106
4.4 Erreurs de prédictions en parties du discours en fonction des types d'erreurs d'orthographe 108

5.1 Les différentes questions du questionnaire de satisfaction 114
5.2 Résultats des classifieurs pour la prédiction des indicateurs de satisfaction. 124
5.3 Comparaison des différents modèles de classification sur la tâche Recommander 127
5.4 Comparaison d'une approche par classification avec une approche par régression sur la tâche Recommander. 129
5.5 Comparaison d'une approche mono-tâche avec une approche multi-tâche sur la tâche Recommander. 130
5.6 Comparaison des modèles utilisant des schémas de classification à 3 classes et à 2 classes avec rejet. 133
5.7 Comparaison des différents modèles en utilisant la MacroF1 en fonction de l'entrée utilisée. 136

6.1 Résultats (en %) des évaluations des plongements de phrases sur les tâches de prédictions de l'acte courant et de l'acte suivant 150

6.2 Évaluation des différents plongements de phrase 158
6.3 Évaluation des différents plongements de phrases sur la tâche « Acte Courant » en ne conservant que les tours d'un des deux scripteurs. . 162
6.4 Évaluation des différents plongements de phrases sur la tâche « Acte Suivant » en ne conservant que les tours d'un des deux scripteurs. . 163

7.1 Étiquettes des relations dialogiques 170
7.2 Distribution des relations dialogiques dans DATCHAREL 176
7.3 Une grammaire Γ_{basic} induite obtenue à partir de l'arbre de la figure 7.7 184
7.4 Une grammaire $\Gamma_{sibling}$ induite obtenue à partir de l'arbre de la figure 7.7 186
7.5 Tailles des CFG Γ_{basic} et $\Gamma_{sibling}$ extraites depuis DATCHAREL 190
7.6 Évaluation des arbres discursifs inférés par les PCFG sur les ensembles de test 192
7.7 Distribution des relations dialogiques du corpus DATCHAACT+R annoté avec $\Gamma^{(25)}_{sibling}$ comparée à DATCHAREL25 201
7.8 Évaluation des arbres discursifs inférés par l'ensemble des analyseurs discursifs sur les ensembles de test 202

Liste des acronymes

BPE
encodage par paires d'octets (*Byte Pair Encoding* en anglais).

CFG
grammaire hors-contexte.

CNN
réseau de neurones convolutifs.

CRF
champ aléatoire conditionnel.

CYK
Cocke Younger Kasami.

DDL
distance de Damerau-Levenshtein.

EM
Espérance-Maximisation (*Expectation-Maximisation* en anglais).

FNC
forme normale de Chomsky.

GRU
Gated Recurrent Unit.

LAS
score de rattachements étiquetés (*Labeled Attachement Score* en anglais).

LDA
allocation de Dirichlet latente (*Latent Dirichlet Allocation* en anglais.

LSA
analyse sémantique latente (*Latent Semantic Analysis* en anglais).

LSTM
Long-Short Term Memory.

MAE
erreur absolue moyenne (*Mean Absolute Error* en anglais).

NPS
Net Promoter Score.

PCFG
grammaire hors-contexte probabiliste.

PDTB
Penn Discourse Treebank.

RNN
réseau de neurones récurrents.

RST
théorie de la structure rhétorique (*Rhetorical Structure Theory* en anglais).

SDRT
théorie de la représentation du discours segmenté (*Segmented Discourse Representation Theory* en anglais).

SPP
Semantic Priming Project.

SVD
décomposition en valeurs singulières.

SVM
machine à vecteurs de support.

TAL
traitement automatique des langues.

TEM
taux d'erreurs mots (*Word Error Rate* en anglais).

TES
taux d'erreurs sérieuses.

TMS
taux de messages avec substitution.

UAS
score de rattachements non-étiquetés (*Unlabeled Attachement Score* en anglais).

UD
unité du discours.

Introduction générale

Les dialogues ont toujours été au cœur des interactions entre humains, et de ce fait, ils ont depuis toujours été un objet d'étude que ce soit en philosophie, en linguistique, dans les sciences humaines ou encore dans les sciences cognitives. Avec l'apparition de l'informatique, le besoin de comprendre la manière dont se déroule et s'organise un dialogue est une question fondamentale afin d'automatiser productions et analyses du langage.

Très tôt dans l'histoire de l'informatique, des systèmes de dialogue entre humain et machine ont été développés, un exemple connu étant ELIZA [Wei66]. Ces agents conversationnels ont récemment acquis une très grande visibilité avec les très nombreux assistants personnels virtuels qui ont été mis sur le marché. En outre, du fait de l'utilisation massive de la téléphonie mais également de l'apparition d'internet, les contraintes physiques de distance et de temps n'imposent plus autant de limites qu'auparavant dans la production de dialogues. En effet, que ce soit en utilisant le téléphone, des SMS, des messageries instantanées, des forums de discussion, des visioconférences ou simplement en face-à-face, dans une seule journée nous sommes souvent amenés à dialoguer plusieurs fois, avec de nombreux interlocuteurs différents et dans des contextes très variés (familial, professionnel, commercial, etc.).

Que ce soit dans le contexte des agents conversationnels ou de l'analyse automatiquement du contenu des conversations (comparaison de dialogues, déterminer les sujets de discussions, etc.), la compréhension parfaite des dialogues par un ordinateur est encore un objectif lointain. Néanmoins, un moyen de s'en approcher est de produire des modèles d'apprentissage automatique prédisant des structures (par exemple, séquentielles ou arborescentes) qui caractérisent les différentes interactions dans le dialogue. En particulier, un aspect de la compréhension d'un dialogue réside dans le fait d'être capable de déterminer le rôle de chaque message dans le dialogue, mais également de pouvoir déterminer les différents enjeux mis en avant dans le dialogue par les locuteurs. Afin de répondre à ces problèmes, une solution est d'analyser le discours conversationnel, qui a pour but d'étudier la façon dont le discours s'organise dans le dialogue afin de résoudre divers enjeux dialogiques.

L'analyse du discours est un champ de recherche relativement ancien et ayant produit une quantité très importante de théories et de schémas d'annotations permettant d'analyser le discours dans des monologues. Certains de ces schémas peuvent également être utilisés pour des dialogues après avoir réalisé quelques adaptations afin de correctement modéliser les interactions entre plusieurs par-

ticipants. En particulier, les schémas doivent modéliser des phénomènes d'enchevêtrements de sous-dialogues, de questions-réponses, de spécifications de nouveaux enjeux dialogiques ou encore des actions sur l'environnement externe ayant une influence sur le dialogue (pause, action sur un objet ayant un lien avec la discussion, etc.). Le dialogue suivant permet d'illustrer différents phénomènes à modéliser :

— **Alice**: Salut, ça fait longtemps que tu ne t'étais pas connecté !

— **Alice**: Qu'est-ce que tu deviens ?

— **Bob**: Salut Alice, ça fait très longtemps oui !

— **Bob**: Je vis tranquillement à Brisbane et toi ?

— **Alice**: Toujours à Alicante.

Dans ce dialogue, on peut constater un enchevêtrement entre un premier sous-dialogue d'ouverture (en rouge) et un deuxième sous-dialogue introduisant et développant un enjeu dialogique (en bleu). Ces phénomènes linguistiques n'existent pas dans les monologues.

Contrairement à l'analyse du contenu, l'analyse du discours n'a pas pour objectif de modéliser le sens des dialogues. Bien entendu, étudier le contenu sémantique est utile pour correctement prendre en compte certains aspects du discours, mais l'analyse du discours se veut plus générale (on ne souhaite pas identifier des thèmes) et doit permettre de faire apparaitre la structure discursive d'un dialogue indépendamment du sens de celui-ci (en s'appuyant sur des étiquettes très génériques : spécification d'un enjeu, réponse, acquiescement, élaboration, etc.). Ces structures discursives permettent de faire abstraction du sens des dialogues et peuvent être utilisées dans le cadre de diverses applications pratiques s'appuyant sur l'organisation interne des conversations.

La thèse s'inscrivant dans le cadre de l'ANR DATCHA, et donc en lien avec l'entreprise Orange, les structures discursives ont diverses applications en lien avec les services d'assistance clientèle. En effet, afin de pouvoir analyser à posteriori le déroulement des conversations, ces structures permettent de produire des groupements de dialogues ayant des schémas de résolution et d'accompagnement similaires. Ces groupements sont, par exemple, utiles pour aider les téléconseillers à améliorer l'accompagnement de futurs clients, en évitant de reproduire des schémas de conversations où les clients étaient mécontents. Par ailleurs, ces structures peuvent être un support intéressant pour développer des systèmes de dialogues qui produisent des messages s'inscrivant correctement dans l'organisation discursive des conversations.

Un inconvénient majeur pour le traitement automatique des langues (TAL) des théories d'analyse du discours est qu'elles se fondent sur des schémas d'annotations complexes qui sont difficiles à mettre en place par des humains et qui le sont encore plus par des machines. En effet, l'annotation nécessite une compréhension complète du dialogue et chaque message peut potentiellement être lié à

des messages en début de conversation. Voici un exemple en apparence simple où il est pourtant indispensable d'analyser et de comprendre le contenu des messages pour correctement lier ensemble questions et réponses :

— **Bob**: Tu fais encore des études ?

— **Bob**: Et Carole est toujours avec toi en colocation ?

— **Alice**: Oui, mais pas au même endroit !

— **Alice**: Et non, je suis désormais embauchée dans une boite d'informatique.

Dans cet exemple, on observe que Bob pose une deuxième question avant d'avoir la première réponse. Ensuite, on peut noter que la première réponse d'Alice pourrait être une réponse valable pour les deux questions. De ce fait, il est indispensable d'analyser le contenu de la deuxième réponse pour connaitre les liens questions-réponses.

Ces difficultés liées à annoter précisément le discours conversationnel font que l'on se contente généralement d'analyser manuellement qu'une petite quantité de conversations. Afin de pouvoir utiliser ces analyses pour d'autres applications du TAL, il faut être capable de prédire automatiquement ces structures discursives. Or, la faible quantité de donnée disponible rend difficile l'utilisation d'algorithmes d'apprentissage automatique — ceux-ci nécessitant généralement beaucoup de données.

Une première solution à ces problèmes d'annotation manuelle est de se passer de celle-ci en construisant des modèles bout en bout, par exemple en construisant des systèmes de dialogue fondés sur la sélection de réponses [Wu+17]. En effet, pour sélectionner la bonne réponse en fonction d'un historique de messages, il est nécessaire d'avoir une compréhension implicite du discours conversationnel. Néanmoins, une telle approche repose à la fois sur le contenu sémantique des messages et sur l'organisation discursive du dialogue. Il est alors difficile de déterminer ce qui est, en pratique, pris en compte par les systèmes de dialogue. En particulier, il devient très difficile de faire ressortir explicitement la structure discursive du document.

Une autre solution est de n'utiliser qu'une annotation de surface du discours conversationnel, c.-à-d. en n'annotant que les fonctions de communication des messages sans expliciter les liens entre les messages. Cette approche a l'avantage de produire une représentation séquentielle (et donc simplifiée) du discours mais en sacrifiant l'identification des différentes interactions qui existent entre les messages des dialogues.

Afin de prendre en compte les liens discursifs entre les messages d'un dialogue, on peut se demander s'il n'est pas envisageable d'utiliser diverses techniques existantes de représentations des mots ou des phrases afin de modéliser le contexte de production dialogique des messages. En effet, ces représentations (par exemple les plongements de mots ou de phrases) sont obtenues dans le but de modéliser diverses propriétés linguistiques telles que la similarité sémantique et la prise en compte du contexte de production.

Pour déterminer si ces représentations permettent de prendre en compte le contexte de production des unités linguistiques, divers cadres d'évaluation ont été mis au point. Néanmoins, les cadres existants s'intéressent peu aux dialogues et en particulier ne permettent pas de faire le lien avec les annotations du discours conversationnel — ce qui est pourtant indispensable pour évaluer la prise en compte des interactions dans les dialogues par ces représentations.

Dans cette thèse, j'ai pour objectif de prédire automatiquement des représentations du discours conversationnel en s'appuyant sur peu de données annotées avec des structures discursives. Trois types d'approches sont développées :

— la production de représentations distributionnelles du discours conversationnel produites à partir d'annotations indirectes et implicites des interactions entre locuteurs ;

— la production de représentations distributionnelles des messages du dialogue en s'appuyant uniquement sur des annotations du discours de surface ;

— la production de représentations explicites (sous forme d'arbre) en utilisant des méthodes d'enrichissement automatique de corpus.

Enfin, je proposerai des cadres d'évaluations permettant de déterminer si une représentation permet de correctement prendre en compte certains aspects des interactions entre locuteurs en s'appuyant sur des tâches en lien direct avec le discours conversationnel.

Ma thèse et le projet ANR DATCHA

Le but du projet DATCHA est de permettre l'extraction de connaissances à partir de très vastes corpus de conversations de type « tchat » entre des clients et des opérateurs. Afin d'extraire des connaissances dans ce contexte, le but est de se fonder sur des approches prenant en compte la dimension interactive et les propriétés propres aux tchats — type de production se trouvant à l'intersection du langage écrit et parlé. Le projet DATCHA cherche à répondre à ces problèmes à travers diverses analyses profondes des conversations, en particulier au niveau discursif, et en définissant des mesures de similarités sémantiques et discursives.

Ma thèse se positionne directement sur ces points-ci en cherchant à développer des approches d'analyse du discours conversationnel, peu couteuses, et qui permettent de prendre en compte les spécificités des tchats et les différentes interactions qui s'y trouvent.

Un autre aspect du projet DATCHA, qui n'est pas étudié dans cette thèse mais qui s'appuie sur celle-ci, est d'utiliser les différentes solutions que j'ai développées dans le contexte des centres de relation clientèle de l'entreprise Orange dans différents cadres applicatifs tels que la génération de rapports, la prédiction de succès d'un dialogue et l'aide en ligne.

Structure et sommaire détaillé de la thèse

La thèse est structurée en trois parties. La première me permet de manière générale d'introduire les théories, concepts et outils sur lesquels la thèse se fondent. Dans le détail, j'y introduis les notions de dialogues et d'analyse de discours, les schémas d'annotations permettant de modéliser cette dernière, et également la notion de représentation dans le contexte du TAL.

La seconde partie présente le corpus DATCHA et l'utilisation d'une approche bout en bout s'appuyant sur une annotation « gratuite » afin de produire des représentations distributionnelles du dialogue prenant en compte de manière implicite les interactions entre locuteurs.

Dans la troisième partie, je propose de s'appuyer explicitement sur les structures discursives afin de produire des représentations (distributionnelles ou arborescentes) des dialogues. En plus du fait de produire des représentations du discours conversationnel, les approches proposées permettent d'être très peu confronté au problème de la production manuelle d'annotations discursives complexes (c.-à-d. les liens discursifs explicites entre les messages du dialogue).

Partie I : Dialogues et modélisation du discours

Avant de s'intéresser à la production de représentations du discours conversationnel à partir de peu de données annotées discursivement, il est important de bien introduire les différentes notions fondamentales : qu'est-ce qu'un dialogue, qu'est-ce que l'analyse du discours dans un dialogue, comment produire des représentations d'unités linguistiques. Cette première partie est dédiée à présenter ces notions et à comprendre les différences entre représentations explicites et implicites.

Chapitre 1 : Le dialogue et l'analyse du discours Ce premier chapitre introduit les deux notions fondamentales de la thèse : le dialogue et l'analyse du discours conversationnel. Je commencerai par définir ce qu'est un dialogue étant donné que ce terme regroupe des productions de natures très variées dues aux modes de communication utilisés, au nombre de participants ou encore aux objectifs des conversations.

À partir de cette définition des dialogues, je définirai ce qu'est l'analyse du discours conversationnel. Cette définition est présentée chronologiquement, ce qui permet de voir que l'analyse du discours conversationnel n'a pas immédiatement suivi le même chemin que l'analyse du discours sur les monologues. En effet, sur les dialogues, les travaux ont surtout été guidés par des tâches applicatives où le besoin était de caractériser la fonction des messages dans le dialogue (question, assertion, ouverture, etc.). Au contraire, sur les monologues, les travaux ont davantage été portés par des linguistes qui se sont principalement intéressés aux interactions entre les phrases (élaboration d'un propos précédent, correction,

propos opposés, etc.). Ces deux approches du problème me permette de définir deux notions : la notion d'analyse de surface du discours — fondée sur les actes de dialogues — et celle d'analyse profonde du discours — fondée sur les théories d'analyse du discours sur les monologues.

Je termine le chapitre par une discussion me permettant d'introduire la problématique principale de la thèse pour laquelle je vais proposer diverses solutions dans les deuxième et troisième parties.

Chapitre 2 : Apprentissage de représentations Ce chapitre permet d'introduire deux autres notions importantes : les représentations distributionnelles et l'évaluation de ces représentations. Les représentations distributionnelles sont un des modes de représentation que j'utilise pour modéliser le discours conversationnel. Par conséquent, ce chapitre me permet de définir les différents enjeux que l'on trouve derrière ces représentations. Par ailleurs, je montre également que l'évaluation de telles représentations est un problème très difficile et qu'un cadre d'évaluation permet rarement d'évaluer toutes les propriétés linguistiques que l'on pourrait souhaiter avoir.

Partie II : Représenter implicitement le discours à partir d'un corpus de tchats

La deuxième partie de la thèse est dédiée à la description du corpus de dialogues étudié et à voir comment je l'utilise pour produire des représentations implicites des interactions entre locuteurs. Dans un premier temps, je décris les dialogues étudiés et les annotations disponibles dans le corpus. Le langage tchats ayant ses propres spécificités, j'étudie l'influence de ce langage sur des tâches de TAL — cette étude permettant d'identifier certains prétraitements utiles à réaliser sur les tchats. Enfin, je proposerai de tirer parti de certaines annotations disponibles « gratuitement » dans le corpus afin d'étudier une approche bout en bout qui pourrait permettre de produire des représentations des interactions dans les dialogues.

Chapitre 3 : Corpus `Datcha` : description et annotations de départ Ce premier chapitre de la deuxième partie me permet d'introduire le corpus DATCHA sur lequel mes travaux se fondent. J'y décris les « tchats » et les annotations qui ont été réalisées et collectées. J'introduis aussi le schéma d'annotations en actes de dialogue utilisé, ainsi que la portion du corpus qui a été annotée en actes de dialogue. Ce corpus est central à ma thèse étant donné que l'ensemble de mes contributions s'appuie sur celui-ci.

Chapitre 4 : Influence du langage tchat sur des tâches de TAL Avant d'entrer directement dans le vif du sujet, je m'intéresse dans un premier temps à la forme

particulière des dialogues étudiés. Ceux-ci étant des « tchats », à la frontière entre oral et écrit, je propose d'étudier l'influence qu'a cette forme du langage sur certaines tâches du TAL. En effet, une particularité de ces dialogues est qu'ils contiennent énormément d'erreurs orthographiques et grammaticales. Dans ce chapitre, j'essaie de répondre à la question de savoir s'il est nécessaire de réaliser des prétraitements avant de pouvoir manipuler les données du corpus DATCHA dans mes expériences en lien avec l'analyse du discours.

Chapitre 5 : Prédire la qualité des interactions avec une méthodologie bout en bout Un intérêt du corpus DATCHA est qu'il contient une quantité très importante de conversations. En complément, celles-ci étant produites dans le contexte de l'assistance clientèle, de nombreuses métadonnées en lien avec cette assistance sont également disponibles. En particulier, pour un grand nombre de dialogues, on dispose des réponses du client à un questionnaire de satisfaction.

À partir de cette annotation « gratuite », je produis des modèles d'apprentissage bout en bout pour prédire automatiquement la satisfaction client en ne se fondant que sur le contenu des dialogues. Une première question est alors de déterminer si le contenu seul est suffisant pour produire des prédictions de bonnes qualités. Pour cela, je propose d'utiliser différents algorithmes d'apprentissage et différents schémas de classification.

En outre, je propose d'étudier si la prise en compte des interactions entre locuteurs est nécessaire à la prédiction de la satisfaction client. En effet, si cette prise en compte est utile, de tels modèles bout en bout modéliseront implicitement certains aspects du discours conversationnel et il sera alors possible d'extraire des représentations distributionnelles des dialogues ou des messages prenant en compte ces aspects-là.

Partie III : Étudier les interactions à partir de représentations explicites du discours

Une limite des approches entièrement bout en bout est qu'elles ne permettent pas de faire ressortir explicitement les structures discursives des dialogues. Or, obtenir des représentations prenant explicitement en compte les interactions dans les dialogues est indispensable pour comprendre le déroulement de ceux-ci. De plus, certaines tâches applicatives peuvent utiliser de telles structures explicites, par exemple pour analyser le comportement de téléconseillers ou pour réaliser des groupements de dialogues ou sous-dialogues ayant des déroulements similaires.

Dans cette dernière partie, je propose deux approches produisant des représentations qui prennent explicitement en compte les interactions entre locuteurs, tout en s'appuyant sur peu d'annotations complexes à réaliser manuellement. La première approche se fonde sur une analyse de surface du discours (les actes

de dialogue) pour produire des représentations distributionnelles des messages d'un dialogue. Le deuxième approche produit une analyse profonde du discours en s'appuyant sur une très petite quantité de dialogues annotés manuellement.

Chapitre 6 : Représentation vectorielle des tours de parole Dans ce chapitre, je pars du constat que les cadres d'évaluation des représentations distributionnelles de phrases ne permettent pas de mettre en évidence la prise en compte du contexte de production, en particulier dialogique, par les modèles. En effet, les algorithmes de productions de plongements de phrases sont généralement conçus pour être utilisés sur des documents produits par une seule personne (et donc pas des dialogues). Par ailleurs, les cadres d'évaluation des plongements produits par ces algorithmes se limitent à l'évaluation de contexte de production non dialogiques tels que l'inférence ou les associations légendes-images.

Je propose donc dans un premier temps dans ce chapitre un cadre d'évaluation se fondant sur les actes de dialogue pour déterminer si des modèles de représentations existants permettent de capturer des phénomènes dialogiques de base. Dans un deuxième temps, je propose de nouveaux modèles de représentations distributionnelles de phrases qui permettent de prendre en compte explicitement le contexte dialogique lors de la création des modèles. Ceux-ci permettent en particulier de constater que les représentations distributionnelles existantes sont peu adaptées aux dialogues et qu'il est possible d'améliorer ces modèles pour mieux prendre en compte le discours conversationnel.

Chapitre 7 : Analyse profonde du discours conversationnel avec peu d'annotations Un inconvénient de l'approche présentée dans le chapitre 6 est qu'elle ne permet pas d'identifier explicitement les relations discursives qui sont modélisées. Dans ce dernier chapitre, je propose d'utiliser une annotation profonde du discours d'un ensemble très restreint de conversations. L'objectif d'une telle annotation est de pouvoir développer un analyseur du discours conversationnel. Deux méthodes sont étudiées : une première où le corpus DATCHA est enrichi automatiquement afin d'avoir davantage de données pour réaliser un apprentissage, et une deuxième où les représentations distributionnelles développées dans le chapitre 6 sont utilisées en entrée de l'analyseur. En complément de l'analyse automatique du discours obtenues par ces méthodes, elles permettent d'évaluer la capacité à modéliser les interactions des représentations obtenues jusqu'à présent.

Publications en relation avec la thèse

Parmi les différentes contributions présentées dans les différents chapitres de la thèse, plusieurs d'entre elles ont été présentées dans des conférences avec comités de lecture.

Chapitre 2 (section 2.5.1.4)

— Jeremy AUGUSTE, Arnaud REY et Benoit FAVRE. « Evaluation of Word Embeddings against Cognitive Processes : Primed Reaction Times in Lexical Decision and Naming Tasks ». In : *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*. Copenhagen, Denmark, 2017

Chapitre 3 (section 3.5)

— Robin PERROTIN, Alexis NASR et Jeremy AUGUSTE. « Dialog Acts Annotations for Online Chats ». In : *25e Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN)*. Rennes, France, 2018

— Catherine THOMPSON, Nicholas ASHER, Philippe MULLER et Jeremy AUGUSTE. « Weakly Supervised Dialog Act Analysis ». In : *Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN - PFIA 2019)*. Toulouse, France, 2019

Chapitre 4

— Géraldine DAMNATI, Jeremy AUGUSTE, Alexis NASR, Delphine CHARLET, Johannes HEINECKE et Frédéric BECHET. « Handling Normalization Issues for Part-of-Speech Tagging of Online Conversational Text ». In : *Eleventh International Conference on Language Resources and Evaluation (LREC)*. Miyazaki, Japan, 2018

Chapitre 5

— Jeremy AUGUSTE, Delphine CHARLET, Géraldine DAMNATI, Benoit FAVRE et Frédéric BECHET. « Customer Satisfaction Prediction with Attention-Based RNNs from a Chat Contact Center Corpus ». In : *25e Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN)*. Rennes, France, 2018

— Jeremy AUGUSTE, Delphine CHARLET, Geraldine DAMNATI, Frédéric BÉCHET et Benoit FAVRE. « Can We Predict Self-Reported Customer Satisfaction from Interactions ? » In : *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Brighton, United Kingdom, 2019

Chapitre 6

— Jeremy AUGUSTE, Frédéric BÉCHET, Geraldine DAMNATI et Delphine CHARLET. « Skip Act Vectors : Integrating Dialogue Context into Sentence Embeddings ». In : *Proceedings of the Tenth International Workshop on Spoken Dialogue Systems (IWSDS)*. Syracuse, Italy, 2019

Première partie

Dialogues et modélisation du discours

Chapitre 1.

Le dialogue et l'analyse du discours

Sommaire

1.1 Introduction 27
1.2 Le dialogue 29
1.2.1 Définitions et vocabulaire 29
1.2.2 Différents types de dialogues 30
1.3 L'analyse de surface du discours conversationnel 34
1.3.1 Théorie des actes de langage 35
1.3.2 Les actes de dialogue 37
1.3.3 Étiquetage automatique en actes de dialogue 43
1.3.4 Conclusion 45
1.4 Structure locale du dialogue 46
1.4.1 Paires adjacentes 46
1.4.2 Actes de la conversation 47
1.5 Analyse profonde du discours conversationnel 49
1.5.1 L'analyse du discours dans le cadre de monologues 49
1.5.2 Schéma d'annotation fondé sur la SDRT : le corpus STAC . . 52
1.5.3 Schéma d'annotation fondé sur le PDTB 54
1.5.4 Construction d'une structure hiérarchique 55
1.6 Discussion 56

1.1. Introduction

Avant de pouvoir mettre en avant les contributions de ma thèse, il est primordial de bien définir les objets qui m'intéressent. Le domaine du TAL s'appuie sur de nombreuses unités linguistiques différentes, et en particulier sur des documents de natures très variées.

Ce premier chapitre va me permettre de définir le type de document sur lequel je vais réaliser tous mes travaux : les dialogues (ou conversations). Les dialogues sont des documents produits par des échanges entre plusieurs participants, ces échanges ayant lieu afin de progresser vers un objectif commun (résoudre un problème concret, en apprendre plus sur les autres, etc.).

Se limiter à cette définition des dialogues serait assez incomplet. En effet, comme tous types de document, il existe une grande variété de dialogues différents, que ce soit par le mode de communication (face à face, téléphone, communications médiées par ordinateur, etc.), par les objectifs du dialogue (réservation, réunion, discussion amicale, etc.) ou par le nombre de participants. Ces différences vont avoir une influence sur les registres de langue utilisés, la longueur des dialogues ou encore le contenu des messages. Dans la section 1.2, je vais détailler les différents types de dialogues qui peuvent être produits.

Un deuxième point qu'il est important de définir est la tâche que je souhaite réaliser sur les dialogues. En effet, un des objectifs du domaine du TAL est d'automatiser de nombreuses tâches portant sur le langage naturel avec pour objectif de faire aussi bien que l'humain. Ces tâches peuvent être aussi bien applicatives (traduction automatique, *chatbots*, reconnaissance de la parole, etc.) qu'analytiques (analyses morphosyntaxiques, syntaxiques, sémantiques, discursives, etc.). Ces dernières peuvent être des étapes à des tâches applicatives ou un moyen de comprendre le processus de production du langage.

Dans ma thèse, je vais m'intéresser à l'analyse du discours conversationnel. L'*analyse du discours* permet de déterminer la manière dont les différents énoncés d'un document, écrit ou oral, fonctionnent ensemble afin de permettre la production d'un discours cohérent. Contrairement à l'analyse de contenu où le but est de comprendre un texte et d'en extraire les idées principales, l'analyse du discours s'intéresse en particulier à l'organisation de la narration et des interactions qui se jouent entre les différents éléments du document.

Dans le dialogue, les interactions concernent également celles entre participants et on cherche alors à déterminer les liens qui existent entre les différents messages du dialogue, l'exemple le plus simple étant les liens questions-réponses. Dans ce chapitre, je vais présenter différentes approches permettant de réaliser des analyses du discours dans le contexte des dialogues. En particulier, nous verrons qu'il existe deux niveaux d'analyse : un premier niveau portant sur les fonctions de communication des différents énoncés du dialogue (section 1.3) et un deuxième niveau portant sur les liens discursifs entre les énoncés, c.-à-d. en réaction à quels énoncés est-ce qu'un énoncé a été produit et dans quel but (section 1.5). Par ailleurs, je présenterai également des analyses discursives à mi-chemin entre les deux niveaux où l'objectif est d'identifier des structures locales du discours conversationnel (section 1.4).

Ce chapitre est la brique de base de ma thèse. L'ensemble de mes contributions s'appuie sur les notions qui y sont présentées et nous discuterons en fin de chapitre des limites de l'analyse automatique du discours conversationnel. Ceci me permettra d'introduire la problématique principale de la thèse.

1.2. Le dialogue

Dans cette thèse, je m'intéresse à des documents bien précis qui sont les dialogues. Bien que ceux-ci soient constitués de mots et phrases comme la majorité des documents étudiés en TAL, il existe de très nombreuses particularités aux dialogues qu'il est important d'identifier. Ces différences avec les autres catégories de documents auront une influence sur les analyses linguistiques à étudier. Dans cette section, je définirai ce qu'est un dialogue de manière générale en donnant également le vocabulaire utilisé afin d'étudier cet objet. Ensuite, je sortirai de ce cadre général afin de détailler les différents types de dialogues qui existent.

1.2.1. Définitions et vocabulaire

Afin de pouvoir correctement étudier les dialogues, il est important dans un premier temps de définir les termes utilisés permettant de désigner différentes portions et unités, les divers phénomènes liés aux dialogues, ainsi que les relations qu'elles entretiennent.

Un *dialogue*, que je nommerai également *conversation*, est constitué d'échanges oraux ou écrits de prises de parole pendant lesquelles les individus réalisent leurs énoncés. Un *énoncé* est une séquence de mots, généralement une phrase, qui est produite par un énonciateur dans un contexte donné. Ces énoncés ont pour objectif de faire progresser une connaissance commune entre les divers individus. Les prises de parole et les énoncés produits avant la prochaine prise de parole sont appelés *tours de parole*. Ces tours de parole peuvent donc être constitués de plusieurs phrases. Il est également possible qu'il y ait plusieurs tours de parole d'un même individu qui se succèdent, s'il y a un acte de silence volontaire par exemple. Dans le contexte de conversations écrites, ces tours de paroles sont parfois également nommés *messages*. Certains tours de parole peuvent être regroupés au sein d'une unité plus vaste appelée *sous-dialogue* qui est constitué de tours de parole du dialogue liés par un sujet ou un sous-objectif commun. Des sous-dialogues peuvent eux-mêmes contenir plusieurs sous-dialogues plus précis. La figure 1.1 présente un exemple de dialogue avec une annotation présentant les différents niveaux.

On appelle *locuteur* la personne qui a produit un tour de parole donné dans le contexte de conversations orales. On appelle *scripteur* cette même personne dans le contexte de conversations écrites. Dans le cas où le mode de communication n'est pas spécifié ou important, j'utilise généralement le terme *locuteur*. La personne à qui est destiné le message provenant du locuteur est appelée *destinataire* ou *allocuteur*. De manière plus générale, les participants au dialogue non locuteurs du tour de parole courant sont appelés *interlocuteurs*.

Dans un dialogue se déroulant de manière idéale, on obtient un enchainement de tours de parole produits par plusieurs locuteurs, et où chaque tour de parole est lié au tour de parole précédent (réponse, élaboration, correction, etc.).

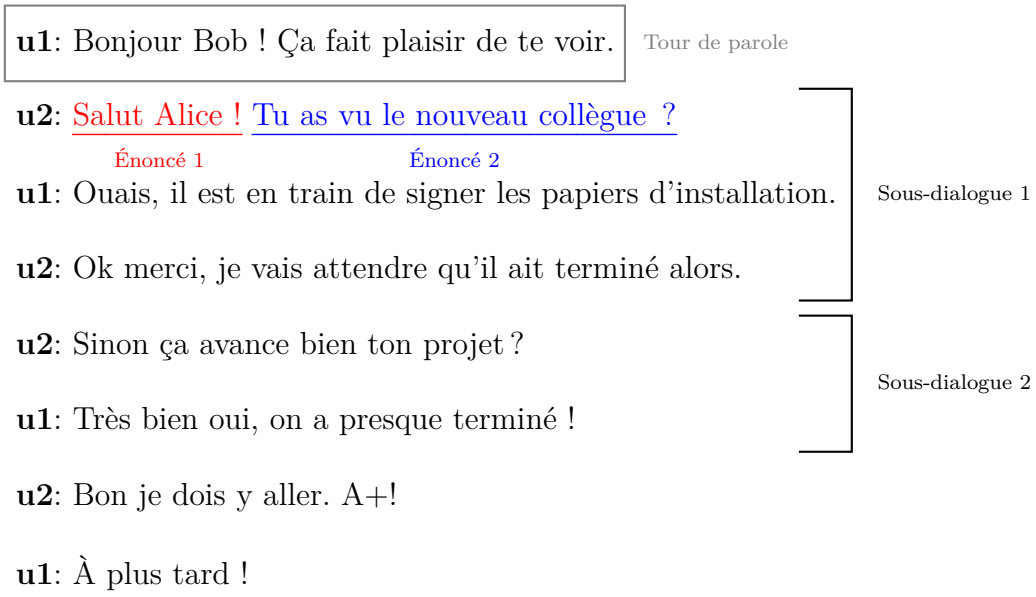


FIGURE 1.1. – Exemple d'un dialogue

Cependant, que ce soit à l'oral ou à l'écrit, on observe très fréquemment des phénomènes d'*enchevêtrements de sous-dialogues*. Lors de ces enchevêtrements, deux sous-dialogues sont ouverts en même temps et on se retrouve alors avec un ou plusieurs tours de parole sur un sujet A suivi d'un ou plusieurs tours de parole sur un sujet B suivi de nouveau par un ou plusieurs tours de parole sur le sujet A. Les notions d'enchevêtrement et de sous-dialogue permettent d'introduire une autre notion : le fait qu'il existe des *relations entre tours de parole*. En effet, les tours de parole sont produits en fonction de ce qui a été énoncé précédemment. Il est donc possible d'associer ensemble les tours de parole ayant des liens entre eux, l'exemple le plus classique étant celui de la question et de sa réponse.

1.2.2. Différents types de dialogues

Le terme « dialogue » est en réalité très vague et permet de désigner un nombre très important d'objets. En effet, on peut considérer des échanges dans des forums de discussion, des échanges entre un humain et une machine ou encore une discussion par téléphone comme étant tous des dialogues. Or il est évident que ces différents types de dialogues n'impliquent pas les mêmes types d'interactions entre les participants, traitent de tâches différentes ou encore utilisent des méthodes de communications différentes ayant des incidences sur la production du dialogue. Il est donc important de bien caractériser ces différents types de dialogues et de comprendre quels sont les points communs et les différences entre ces derniers afin de pouvoir correctement les étudier et les comparer. Le détail des différents types de dialogues sera fait en trois parties. Dans un premier

temps, je m'intéresserai à l'influence de la nature et du nombre des participants sur la forme des dialogues. Dans un second temps, la raison d'être des conversations, influençant les rôles tenus par les participants et donc la forme et le contenu des dialogues, sera étudié. Enfin, la dernière partie s'intéressera à l'impact que peuvent avoir les différents modes de communication sur la production des dialogues.

1.2.2.1. Nature et nombre des participants

Ce qui fait qu'un dialogue en est un est avant tout la présence d'au moins deux participants qui échangent, verbalement ou non, dans le but de faire progresser une connaissance commune. La nature de ces participants est importante à identifier puisque c'est eux qui vont alimenter le contenu du dialogue.

De nos jours, les participants à des dialogues sont généralement des humains discutant ensemble. Cependant, de plus en plus, on voit apparaitre des agents conversationnels non humains dans divers contextes de la vie de tous les jours : chatbots d'assistance en ligne ou assistants personnels par exemple. Le but pour ces agents est bien entendu d'imiter le mieux possible l'humain, cependant ce n'est pas encore parfait, loin de là, et cela a pour conséquence que les dialogues ainsi produits peuvent être très différents des dialogues produits entre humains. En effet, les systèmes de dialogues ont généralement deux problèmes majeurs [SA07] :

— ils ne conservent pas réellement un historique de la conversation, et ne savent pas forcément correctement s'en servir s'ils en ont un à disposition ;

— ils ne sont pas réellement capables de comprendre ce qui est dit, ils ne font qu'extraire des informations d'une base de données ou répéter (plus ou moins bien) des séquences déjà rencontrées.

Ces deux aspects font que les utilisateurs humains se rendent compte assez rapidement que leur interlocuteur n'est pas humain. De ce fait, les humains vont potentiellement changer la façon dont ils s'expriment afin de s'adapter à la machine. En effet, leurs messages deviennent moins longs et les utilisateurs utilisent un vocabulaire plus limité [HFF15]. De plus, les utilisateurs de ces systèmes peuvent également être moins ouverts ou agréables qu'avec d'autres humains [MX17]. Ces différents aspects font que les conversations produites dans ces contextes-là auront tendances à être différentes, pouvant même donner l'impression d'être plus « simples », que des conversations entre humains.

Le nombre de participants est également un facteur important dans un dialogue car il va avoir une influence sur sa forme et sa nature, en particulier entre les dialogues bipartis et les dialogues multi-parties [Tra04]. En effet, lorsqu'il n'y a que deux participants, il est évident pour chacun d'entre eux que les dires de l'un sont destinés à être entendus ou lus par l'autre. Au contraire, lorsqu'il y a au moins trois participants, cette évidence disparait. En effet, un énoncé peut

être destiné à tous les autres participants, ou uniquement certains d'entre eux. Afin de que les participants puissent savoir à qui cet énoncé est destiné, il est nécessaire de donner des indices, verbaux ou non-verbaux sur le ou les *destinataires*, les autres participants ayant alors à ce moment-là un rôle secondaire passif d'*auditeur* ou de *lecteur*. Cet aspect-là a un impact direct sur certaines tâches du TAL où l'identification des locuteurs [BF95] et des destinataires [SSJ78 ; Jo04] est primordiale, telles que l'analyse du discours [Afa+15], la compréhension de réunions [Tur+08 ; BSL16] ou encore dans les systèmes de dialogues multiparties [TR02 ; de +17].

1.2.2.2. Raison d'être du dialogue

Un autre aspect important à prendre en compte lors de travaux sur des dialogues est l'objectif de ces derniers. En effet, tout dialogue a lieu pour une certaine raison : simple discussion entre connaissances, réunion de travail, assistance clientèle, réservation dans un établissement, etc. Ces différentes raisons vont avoir pour conséquences d'induire les types d'énoncés rencontrés. En effet, lors d'une réservation d'une table dans un restaurant, le dialogue sera principalement constitué de questions et réponses et qui seront très spécifiques au domaine de la restauration. Au contraire, dans une réunion de travail, les énoncés seront très variés comprenant des questions, des argumentations, des affirmations, des opinions, etc.

Il est donc évident que le vocabulaire utilisé, les types de phrases employées ou encore la longueur des énoncés et des dialogues ne seront pas les mêmes. Un autre aspect qui sera très influencé est la variété des rôles occupés par les participants dans la conversation. En effet, bien que dans chaque conversation les différents participants vont pouvoir régulièrement passer de locuteur à auditeur et inversement, ces participants ne vont pas forcément tous poursuivre le même objectif personnel dans le dialogue. Ces objectifs personnels variés vont provoquer l'apparition de rôles potentiellement « asymétriques » où un ou plusieurs participants deviennent meneurs de la conversation, c.-à-d. qu'ils décident quels vont être les orientations des sous-dialogues et les sujets à discuter, alors qu'au contraire les autres participants auront des rôles beaucoup plus passifs en se contentant de continuer un sous-dialogue démarré par le meneur, par exemple en répondant aux questions. En fonction du type de dialogue, les rôles seront variablement marqués, et on peut facilement dégager deux types opposés de dialogues qui correspondent à deux extrêmes d'un continuum :

— les dialogues ouverts où les sujets de discussion, et donc les types de messages utilisés, sont assez libres et où les rôles des participants vont être amenés à évoluer, voire à ne pas vraiment exister, au fil de la discussion ;

— les dialogues orientés vers une tâche où le sujet de discussion tourne autour d'une tâche bien précise et donc les rôles sont également bien définis et statiques.

Bien entendu, il est rare de rencontrer des dialogues correspondant à ces deux extrêmes. Généralement, un dialogue considéré ouvert aura également des objectifs latents qui influenceront légèrement les rôles des participants et les sujets de discussions. De même, un dialogue orienté vers une tâche pourra avoir des séquences de tours de paroles pendant lesquelles les participants discutent de sujets plus libres, ou tout du moins qui ne sont pas en lien avec la tâche à résoudre.

1.2.2.3. Mode de communication

Un dernier aspect très important à prendre en compte est le vecteur de communication utilisé dans le dialogue. Le moyen le plus naturel est de dialoguer en face à face avec ses interlocuteurs. Cependant, et plus particulièrement de nos jours, il existe de nombreux moyens de communication permettant aux différents participants de ne pas être dans le même lieu. Ces différents modes de communication vont chacun introduire leurs particularités et modifier la forme des dialogues.

Lorsque deux humains communiquent ensemble, ils n'utilisent pas uniquement les mots pour faire passer leurs messages et intentions. Ils peuvent aussi communiquer des informations à l'aide d'expressions du visage, de gestes, d'intonations ou du bruit [ANA92 ; Tay+98], par exemple pour faire des retours à l'interlocuteur. De manière évidente, si le canal de communication ne permet pas d'utiliser la vision ou la parole, il ne sera pas possible de faire passer l'information à son interlocuteur de ces manières-là et un autre moyen doit être trouvé. De ce fait, afin de faire passer la même information, le contenu du dialogue devra potentiellement être adapté au mode de communication.

Que ce soit dans des contextes oraux ou écrits, le mode de communication peut avoir une influence sur la latence du temps de réponse. En effet, que ce soit pour des raisons techniques (par exemple une mauvaise connexion internet) ou pour des raisons humaines (par exemple une personne correspondant avec des personnes différentes dans plusieurs conversations simultanément par SMS), cette latence pourra provoquer des attentes de réponse incomprises par l'interlocuteur. Ceci pourra se traduire par des messages de relances ou encore des continuations de propos précédents pouvant conduire à l'apparition d'enchevêtrements par la suite (l'interlocuteur pouvant vouloir réagir aux propos précédents).

Dans le contexte de conversations écrites, il existe aussi un autre aspect à considérer qui est le support de communication utilisé. En effet, un attribut intéressant de l'écrit par rapport à l'oral est qu'il permet de garder beaucoup plus facilement un historique de la conversation visible par les divers participants. Le fait que l'historique soit disponible permet aux différents participants de faire beaucoup plus facilement référence à des messages précédents, ce qui permet d'éviter certaines répétitions d'informations ou certains quiproquos qui peuvent beaucoup plus facilement apparaitre à l'oral. Il existe plusieurs types de supports permettant de dialoguer par écrit, divisés en deux catégories : les supports *synchrones*

et les supports *asynchrones*. Dans les asynchrones, tels que les forums de discussion ou les courriels, les différents participants peuvent participer au dialogue sans qu'il y ait besoin que leurs interlocuteurs soient présents au même moment. Étant donné qu'il n'est pas nécessaire de construire des réponses rapidement, il est beaucoup plus facile de construire de très longs messages, des monologues, pouvant faire passer un nombre très important d'informations dans un même temps. Dans les supports synchrones, tels que les messageries instantanées ou « tchats », les différents participants s'attendent à ce que tout le monde reçoive, lise et réponde dans une période de temps très courte. Les supports synchrones sont donc des dialogues proches de ceux qui sont produits à l'oral. On pourra aisément assister à des phénomènes d'enchevêtrements par exemple, contrairement aux supports asynchrones qui peuvent presque être vus comme étant une succession de monologues liés entre eux par un sujet commun.

1.3. L'analyse de surface du discours conversationnel

Maintenant que la notion de dialogue est définie, ainsi que les différentes formes qu'ils peuvent prendre, il est désormais possible de s'intéresser à l'analyse du discours conversationnel.

Dans l'analyse du discours de documents de manière générale, l'objectif est d'identifier la manière dont chaque partie du document s'inscrit dans le discours. Dans ce but-là, il est important de distinguer le rôle des unités du discours, c.-à-d. quel type d'information est véhiculé par cette unité, et les relations discursives entre unités du discours, c.-à-d. comment et avec quelles autres unités du discours interagit une unité du discours. Ceci est donc également vrai dans le contexte de conversations où il faut distinguer pour les énoncés (et par extension les tours de paroles) les rôles qu'ils ont au sein du discours conversationnel (par exemple une question ou une affirmation) et les relations discursives qu'ils ont avec les autres énoncés (par exemple une spécification ou une réponse).

Dans cette section, nous allons nous concentrer sur le rôle des énoncés dans le discours. Ce premier niveau d'analyse du discours conversationnel nous permet de construire une structure séquentielle du discours qui est la séquence des rôles des énoncés de la conversation. On considère alors que cette structure est une *structure de surface du discours* puisqu'elle ne modélise pas explicitement les relations et les interactions entre énoncés. La section est divisée en trois sous-sections. Dans la sous-section 1.3.1, je m'intéresserai à définir et à décrire la théorie fondatrice des actes de langages qui est une première approche où le but est d'identifier les intentions derrière les énoncés dans le cadre de conversation. Dans la sous-section 1.3.2, je m'intéresserai tout d'abord à identifier les limites de la théorie des actes de langages et je décrirai la notion d'actes de dialogue ainsi que les différents modèles d'annotation existants qui sont une extension des actes de langage qui mettent davantage l'accent sur l'aspect communicatif

des énoncés. Enfin, dans la sous-section 1.3.3, j'introduirai les problématiques liées à l'étiquetage automatique en actes de dialogue, pour ensuite présenter des étiqueteurs existants obtenant des performances à l'état de l'art.

1.3.1. Théorie des actes de langage

La problématique de la définition des rôles des énonciations est un problème ancien et qui a été en particulier mis en avant par le philosophe AUSTIN [Aus62] dans *How to Do Things with Words* et a introduit la *théorie des actes de langage*. Dans cet ouvrage, il définit la notion de *performativité* qui est le fait, pour un énoncé, de réaliser lui-même ce qu'il énonce en opposition avec la notion de *constativité* où l'énoncé fait simplement un constat sans répercussion sur le monde. Un exemple classique d'énoncé performatif est celui du juge qui va déclarer une sentence déterminant l'avenir d'un accusé. Lorsqu'il énonce la phrase « Je vous déclare coupable », en plus d'énoncer un fait, l'état de l'individu s'en retrouve modifié en passant du statut d'accusé à celui de coupable. AUSTIN considère dans un premier temps qu'il existe uniquement ces deux types d'énoncés (constatifs et performatifs). Il a par la suite développé ces travaux afin de définir trois niveaux d'*actes de langage* :

— *locutoire* : le fait de produire un énoncé ;

— *illocutoire* : intention du message réellement transmis quel que soit le sens immédiat de l'énoncé ;

— *perlocutoire* : l'effet psychologique de l'énoncé sur le destinataire.

Dans mes travaux, je m'intéresse aux *actes illocutoires* des énoncés. En effet, ces actes permettent de caractériser la nature des messages que les locuteurs cherchent à transmettre à leur destinataire. Ceci correspond à donner le rôle d'un énoncé au sein de la conversation et donc de construire un premier niveau d'analyse du discours conversationnel. AUSTIN propose une taxinomie préliminaire et fondatrice des actes illocutoires permettant d'après lui de classer les énoncés. Ces classes d'actes illocutoires sont construites à partir d'une liste de verbes de la langue anglaise et les classes sont les suivantes :

— *verdictifs* : ce sont des jugements, des avis prononcés par des juges, des jurés, des arbitres, etc. Cela correspond à dire une découverte, un constat sur un fait pour lequel il est difficile d'avoir des certitudes ;

— *exercitifs* : cela correspond à utiliser des pouvoirs, des droits ou des influences, par exemple le vote, l'ordre ou le conseil ;

— *promissifs* : cela concerne les engagements à réaliser quelque chose dont les promesses mais également les déclarations et annonces d'intentions ;

— *comportatifs* : celui-ci inclus les énoncés ayant un lien avec l'attitude et le comportement social tel que les excuses, les félicitations ou les insultes ;

— *expositifs* : ces derniers exposent clairement comment nos propos s'intègrent dans une argumentation ou conversation et sont généralement explicatifs. Ces énoncés commencent généralement par des groupes verbaux comme « Je réponds », « J'illustre » ou « Je concède ».

AUSTIN précise clairement que ces cinq classes ne couvrent pas nécessairement tous les cas possibles, en particulier des cas marginaux ou étranges, et qu'il est également possible qu'il y ait des intersections entre certaines classes.

Afin d'approfondir ces travaux, SEARLE [Sea75] propose une taxinomie des actes illocutoires basée sur les classes précédentes. Il définit cinq classes d'actes de langage illocutoires. Des exemples d'énoncés associés à leurs actes de langage sont proposés dans la table 1.1. Les classes sont les suivantes :

— *assertifs* (ou *représentatifs*) : le but de ces énoncés est d'engager le locuteur sur quelque chose qu'il pense être une vérité ;

— *directifs* : dans ces énoncés, le locuteur essaie (de la simple invitation jusqu'à l'ordre en passant par le conseil) de faire en sorte que le destinataire fasse quelque chose ;

— *promissifs* : dans ces énoncés, le locuteur s'engage sur des plans d'actions futurs, tels que des promesses ou des serments ;

— *expressifs* : le but illocutoire de ces énoncés est d'exprimer l'état psychologique vis-à-vis d'une proposition, incluant par exemple les remerciements et les félicitations ;

— *déclaratifs* : ces énoncés modifient la réalité en accord avec la proposition de la déclaration.

Énoncé	Acte de langage
Il fait beau.	Assertif
Tu devrais penser à changer de moyen de transport.	Directif
Je vais terminer le projet dès que possible.	Promissif
C'est génial ! (l'interlocuteur accepte une invitation)	Expressif
Oui je le veux ! (dans le contexte d'un mariage)	Déclaratif
Je vous déclare coupable !	Déclaratif
Je te parie 5 euros qu'il va pleuvoir !	Directif et Promissif

TABLE 1.1. – Exemples d'énoncés associées à leurs actes de langage illocutoires

Ces travaux de SEARLE sont à leurs tours approfondis par HANCHER [Han79] qui propose dans un premier temps de construire explicitement de nouvelles classes afin de prendre en compte des énoncés pouvant être inclus dans de multiples classes (actes *promissifs directifs* et *assertifs déclaratifs*). Dans un second

temps, il propose également de distinguer les actes dits *coopératifs* et les actes dits *collectifs* et *multiples*. En effet, les actes de langage *coopératifs* correspondent aux actes de langage où le fait d'avoir plusieurs locuteurs est essentiel à la nature de l'acte comme, par exemple, l'établissement d'un contrat ou le fait de réaliser un vote. Ceci est à distinguer des énoncés « accidentellement » simultanés de la part de plusieurs locuteurs (actes *multiples*) et les énoncés dans lequel le locuteur parle également au nom d'une autre personne (actes *collectifs*).

Ces différents travaux restent assez abstraits et sont généralement uniquement étudiés sur des exemples construits manuellement par ces trois auteurs, et sur des énoncés isolés. En particulier, ils ne se confrontent jamais à un nombre important d'exemples concrets de dialogues produits par des humains. La théorie des actes de langage a cependant rapidement gagné en popularité dans plusieurs domaines différents tel que la psychologie [Bru74], la linguistique [Fil71 ; Sad74] ou encore la littérature [Ohm71].

1.3.2. Les actes de dialogue

Une des grandes limites de la théorie des actes de langage est que la fonction de communication entre plusieurs locuteurs est assez peu prise en compte lors de l'analyse des énoncés. En effet, les énoncés sont généralement étudiés de manière isolée sans prendre en compte le fait que les différents participants à la conversation la construisent dans le but de répondre à divers objectifs, que ce soit pour échanger des idées, informer ou argumenter. Afin de mieux prendre en compte ces aspects problématiques, différents travaux en parallèles essaient de construire des actes illocutoires mais davantage focalisés sur la dimension communicative des conversations. Dans cette sous-section, je passe dans un premier temps en revue différents travaux qui ont cherché à faire évoluer la notion d'acte de langage afin que ces actes prennent mieux en compte la dimension communicative des conversations. Dans une seconde partie de la sous-section, je présente deux approches cherchant à définir de manière précise ce que sont des actes de dialogue, tout en faisant en sorte que ces actes de dialogue ne soient pas spécifiques à un domaine particulier.

1.3.2.1. Premiers travaux pour définir les actes de dialogue

Le but de ce qui suit est de décrire la manière dont a été introduit le concept d'actes de dialogue et les besoins auxquels ils répondent. Un *acte de dialogue* est similaire à un acte de langage du fait qu'ils s'intéressent tout deux aux intentions du locuteur. Cependant, contrairement aux actes de langage, les actes de dialogue mettent davantage l'accent sur la dimension communicative des dialogues, c-à-d. qu'ils ont pour but de détailler le type de communication que le locuteur cherche à réaliser avec ses interlocuteurs, en plus de ses intentions.

Les travaux de KUME et al. [KSY89] font partie des premiers travaux s'intéressant à ces aspects-là. Ils proposent de se concentrer sur les *forces illocutoires* des énoncés, c.-à-d. uniquement les intentions du locuteur. Ces forces illocutoires étant en pratique des actes de dialogue, j'utiliserai ce dernier terme dans le paragraphe qui suit, même si ce n'est pas le terme utilisé par les auteurs. Ils définissent 9 types d'actes de dialogue qui correspondent globalement aux classes de SEARLE [Sea75] mais s'intéressent davantage aux différents types d'interactions que vont produire les énoncés. Au lieu d'avoir un seul acte directif, on obtient alors un acte de dialogue pour les requêtes, un autre pour les questions appelant des précisions, ou encore un autre pour les questions introduisant un nouveau problème. Ces besoins distincts proviennent du fait que les auteurs souhaitent réaliser une traduction automatique en se basant sur les intentions des locuteurs et non sur le contenu des énoncés qui peuvent manquer d'informations sur les contextes dans lesquels ils sont produits. Or les actes de langage restent trop abstraits pour réaliser cette tâche. En dehors des énoncés directifs, la majorité des actes de dialogue restent très similaires aux actes de langage classiques. NAGATA et MORIMOTO [NM94] approfondissent ce travail et définissent 15 actes de dialogue. Ces derniers sont encore une fois une volonté de principalement modéliser les types d'interactions produites par le locuteur, en plus de ses intentions. Ce modèle-là permet de ne pas confondre tous les énoncés assertifs utilisés en guise de réponses en créant plusieurs classes détaillées (`rejet`, `réponse simple`, `acquiescement`, `permission`). De même, les énoncés directifs utilisés comme questions sont également divisés en plusieurs classes (`offre`, `invitation`, `suggestion`, `confirmation`). À partir de ces nouveaux actes, ils s'intéressent à construire un modèle de dialogue afin de prédire les actes de dialogue de l'énoncé suivant en se basant sur les actes de dialogue des énoncés passés.

Le projet VERBMOBIL [Kay92] qui a pour but de permettre à des personnes parlant des langues différentes de dialoguer ensemble repose sur une modélisation du dialogue fondée sur la théorie des actes de langage [AMR95]. Pour cela, dans le cadre du projet, les auteurs définissent des actes de langage se focalisant plus particulièrement sur la communication dans le dialogue. Ils nommeront explicitement ces actes de langage comme étant des *actes de dialogue*. Les actes de dialogues interviennent à plusieurs niveaux, que ce soit pour aider la tâche de reconnaissance de la parole en contraignant le lexique ou pour construire un historique du discours afin de mieux comprendre un énoncé faisant référence à un ou plusieurs énoncés passés. Dans le cadre de ce projet, une taxinomie d'actes de dialogue est également établie [Jek+95]. Elle est construire sur deux niveaux avec le premier niveau qui identifie 18 catégories d'actes de dialogue générales. Le deuxième niveau apporte 36 actes additionnels mais qui ont pour but de détailler les catégories d'actes de dialogue du premier niveau. La table 1.2 illustre un sous-ensemble de cette taxinomie.

La grande quantité d'actes de dialogue permet d'obtenir une analyse plus fine du discours dans le dialogue. Il est tout de même important de noter que les

Actes de premier niveau	Actes de second niveau
Clarification	Clarification-Requête Clarification-Réponse Clarification-État
DemanderSuggestion	DemanderSuggestion-Date DemanderSuggestion-Localisation DemanderSuggestion-Durée
Acceptation	Acceptation-Date Acceptation-Localisation Acceptation-Durée
Motivation	Motivation-RendezVous
Confirmation	–

TABLE 1.2. – Sous-ensemble de la taxinomie d'actes de dialogue du projet VERBMOBIL

dialogues étudiés dans le cadre de VERBMOBIL sont plutôt des dialogues ayant lieu lors de réunions de travail. Par conséquent, les actes de dialogue, même s'ils restent pour la plupart assez génériques, sont tout de même construits afin de bien correspondre à ce type de dialogues là.

Dans le but de construire un agent conversationnel pouvant réaliser de la planification de trajets, le projet TRAINS [All+95] se base également sur les actes de dialogue. L'objectif est d'analyser et de comprendre les actes de dialogue produit par l'utilisateur dans le but de permettre à l'agent conversationnel de déterminer l'acte de dialogue à produire pour l'énoncé suivant. Ceci permet ensuite de générer une phrase en langage naturel. Dans TRAINS, les 8 actes de dialogues utilisés sont très spécifiques dans le but de correspondre un maximum au type de dialogues présents dans le corpus utilisé. Ces actes sont : `informer`, `question-oui-non`, `vérification`, `suggestion`, `requête`, `acceptation`, `refus`, `information-complémentaire`. On peut donc bien voir que les actes *assertifs* et *directifs* définis par SEARLE [Sea75] sont beaucoup plus présents que les autres. Ceci est dû à la nature des conversations dans le corpus utilisé qui offre très peu de variations dans le déroulement des conversations.

1.3.2.2. Constructions de schémas d'annotation génériques

Comme on a pu le voir, les différents corpus créés jusque-là ont tous des jeux d'étiquettes différents, certains sont plus précis que d'autres, et certains sont plus spécifiques à des domaines particuliers. Il est cependant important de noter que même si les jeux d'étiquettes sont différents, on retrouve tout de même beaucoup

de classes en commun, les principales différences se retrouvant surtout au niveau de la précision et de la portée des classes. On peut donc se demander s'il n'est pas envisageable de construire des schémas d'annotation d'actes de dialogue pouvant s'adapter à tout type de dialogue.

CORE et ALLEN [CA97] proposent dans ce contexte-là un nouveau schéma d'annotation nommé DAMSL basé sur les travaux du *Multiparty Discourse Group* lors des réunions de la *Discourse Research Initiative*. Ce schéma d'annotation a deux objectifs : proposer un ensemble flexible d'actes de dialogue pouvant s'appliquer à n'importe quel type de dialogue et faire en sorte que chaque énoncé puisse avoir plusieurs actes de dialogue multidimensionnels. De la même manière que pour les actes de dialogue définis dans le cadre du projet VERBMOBIL, le schéma d'annotation DAMSL est construit de manière hiérarchique avec plusieurs niveaux permettant de préciser la nature des actes de dialogue. Cependant, dans DAMSL le premier niveau n'est pas réellement constitué d'actes de dialogue à proprement parler mais est plutôt utilisé afin de distinguer les trois grandes catégories d'actes de dialogue existants :

— les fonctions de communication prospectives ;

— les fonctions de communication rétrospectives ;

— les caractéristiques de contenu et de forme des énoncés.

Ces trois catégories sont définies et décrites dans les paragraphes suivants.

Dans DAMSL, une grande distinction est faite entre les énoncés réagissant à des énoncés passés et les énoncés qui ont pour but d'influencer le futur de la conversation. Ces premiers énoncés ont ainsi donc une *fonction rétrospective* alors que ces derniers ont une *fonction prospective*. Dans chacune de ces catégories, plusieurs classes sont définies et appelées *dimensions*, correspondant à un premier niveau d'actes de dialogue très généraux. Ces dimensions sont construites de manières à être indépendantes les unes des autres. Elles peuvent ensuite elles-mêmes avoir des sous-niveaux composés d'actes de dialogue davantage précis. Les fonctions de communication prospectives incluent les actes de langage *assertifs*, *directifs*, *promissifs* et *déclaratifs*, cependant un énoncé peut dans ce schéma d'annotation être les quatre à la fois.

La figure 1.2 présente les différents actes de dialogue ayant une fonction prospective dans le schéma d'annotation DAMSL. On peut en particulier y voir les différentes dimensions (en gras) avec les différents actes de dialogue plus précis détaillant celles-ci.

Les énoncés *assertifs* constituent la dimension nommée Affirmation (*Statement* en anglais), elle-même divisée en plusieurs sous-catégories permettant de représenter le fait que le locuteur peut faire passer une nouvelle information ou réitérer une affirmation.

Les énoncés *directifs* sont inclus dans la dimension Influencer-Action-Future-Destinataire (*Influencing-Addressee-Futur-Action* en anglais) qui inclue tous les énoncés qui évoquent des actions potentielles du destinaire.

- **Affirmation** :
  - Assertion
  - Reassertion
  - Autre Affirmation
- **Engager Action Future Locuteur**:
  - Offre
  - Engagement
- **Influencer Action Future Destinataire**:
  - Option Ouverte
  - Directif :
    - Requête Information
    - Action Directive
- **Performatif**
- **Autres Fonctions Prospectives**

FIGURE 1.2. – Fonctions de communication prospectives du schéma d'annotation DAMSL

Les énoncés *promissifs* sont directement représentés par la dimension Engager-Action-Future-Locuteur (*Committing-Speaker-Futur-Action* en anglais) et est divisée afin de distinguer les offres des engagements.

Les énoncés *déclaratifs* sont représentés par la dimension Performatif qui regroupe les énoncés qui réalisent eux-mêmes ce qu'ils énoncent.

Pour les cas où un énoncé n'entre pas dans les catégories précédentes mais que l'énoncé a une fonction prospective, la catégorie Autres-Fonctions-Prospectives est définie.

Les fonctions de communication rétrospectives permettent de couvrir un aspect qui est assez peu pris en compte par la théorie des actes de langage. En effet, même si la plupart des énoncés concernés peuvent être inclus dans les classes *assertifs* ou *expressifs*, ces classes ne permettent pas de mettre en évidence le fait que les énoncés correspondant sont une réaction à un ou plusieurs énoncés précédents, et en particulier, elles ne permettent pas d'identifier la nature exacte de la réaction. DAMSL définit donc plusieurs catégories permettant de prendre en compte ces différentes réactions possibles. La figure 1.3 présente les différents actes de dialogue ayant une fonction rétrospective dans le schéma d'annotation DAMSL.

- **Acceptation** :
  - Accord
  - Accord Partiel
  - Peut-être
  - Rejet Partiel
  - Rejet
  - Attente
- **Réponse**
- **Compréhension** :
  - Signal Incompréhension
  - Signal Compréhension :
    - Acquiescement
    - Reformulation/Répétition
    - Complétion
  - Correction

FIGURE 1.3. – Fonctions de communication rétrospectives du schéma d'annotation DAMSL

La dimension Affirmation est utilisée lorsqu'un locuteur indique son degré d'accord à une proposition d'un interlocuteur. Afin de bien prendre en compte

les différents degrés d'accord, plusieurs sous-catégories allant de l'acceptation jusqu'au rejet sont définis.

La dimension Compréhension capture le phénomène où le locuteur cherche à indiquer à son interlocuteur s'il a compris ou non son énoncé. La catégorie est donc elle-même divisée afin de distinguer les signaux de non-compréhension, les signaux de compréhension (acquiescements, complétion, répétition) ou encore les corrections.

La dimension Réponse permet d'indiquer qu'un énoncé est une réponse à un énoncé précédent de demande d'information. Celle-ci n'a pas besoin d'avoir d'actes de dialogue plus précis car il suffit de l'utiliser en combinaison avec d'autres dimensions définies auparavant.

DAMSL introduit également des dimensions additionnelles permettant de décrire la nature ainsi que des caractéristiques particulières d'un énoncé vis-à-vis du dialogue. Ces dimensions forment ensemble la dernière grande catégorie. Cette catégorie est détaillée dans la figure 1.4.

- **Niveau d'information** :
  - Tâche
  - Gestion Des Tâches
  - Gestion Des Communications
- **Statut communicatif** :
  - Abandon
  - Non Interprétable
  - Discours Intérieur

FIGURE 1.4. – Caractéristiques de contenu et de forme des énoncés du schéma d'annotation DAMSL

Deux types de caractéristiques sont mis en avant : le niveau d'information de l'énoncé et l'état communicatif de l'énoncé. Le premier type de caractéristiques permet d'identifier la nature de certains sous-dialogues dans le dialogue. Ces étiquettes permettent, entre autres, d'indiquer si les énoncés sont produits afin de discuter du sujet motivant la présence du dialogue ou si les énoncés ont pour but de planifier le bon déroulement du dialogue. Le second type de caractéristiques permet d'indiquer si un énoncé a été interrompu ou s'il est tout simplement inintelligible.

Même si DAMSL a pour but d'être applicable pour n'importe quel type de dialogue, il a tout de même été construit en s'appuyant sur le corpus TRAINS qui est un corpus de conversations très spécifiques au domaine de la planification de trajets. Une version légèrement adaptée de DAMSL a cependant été utilisé de manière concluante afin d'annoter le corpus SWITCHBOARD [GHM92 ; JSB97] qui est un corpus de conversations téléphoniques dans lesquelles les deux participants devaient discuter d'un sujet défini à l'avance.

Bien que DAMSL permette d'obtenir un ensemble d'étiquettes d'actes de dialogue génériques prenant en compte l'aspect multidimensionnel des énoncés, le schéma d'annotation n'est pas exempt de critiques, en particulier de la part de BUNT [Bun06] qui met en avant le fait que les différents niveaux manquent de fondements théoriques. Afin de mieux prendre en compte cette problématique,

il propose un autre schéma d'annotation, DIT++ [Bun09]. Dans ce schéma d'annotation, il reprend la notion de dimensions pour désigner les classes principales d'actes de dialogue et propose qu'une dimension soit un regroupement de fonctions communicatives qui adressent toutes un aspect particulier de la participation des locuteurs au dialogue. Dans une dimension traitant un aspect donné, il propose de surcroit que les fonctions communicatives respectent les conditions suivantes :

— les participants du dialogue peuvent traiter cet aspect à l'aide de traits linguistiques ou non-verbaux ayant ce but-là, c.-à-d. qu'uniquement des aspects de communication pouvant être caractérisé selon des observations empiriques des comportements dans le dialogue ;

— cet aspect peut être traité de manière indépendante des autres aspects, c.-à-d. qu'un énoncé peut avoir une fonction communicative dans une dimension indépendamment des fonctions qu'il peut avoir dans d'autres dimensions.

Il propose donc un ensemble de 10 dimensions qui permettent de prendre en compte les conditions précédentes. Ces dimensions sont les suivantes : tâche ou activité, retour locuteur, retour interlocuteur, gestion des tours de parole, gestion des prises de contact, gestion du temps, structuration du discours, gestion des communications personnelles, gestion des communications de l'interlocuteur, gestion des obligations sociales. En plus de ces dimensions, il définit également des fonctions de communications pouvant être utilisées dans toutes les dimensions. Ces fonctions de communications restent très similaires à celles pouvant être trouvées dans les différents jeux d'étiquettes existants, et en particulier restent fortement basées sur les 5 actes de langage de SEARLE [Sea75]. Cependant, des fonctions de communication spécifiques à certaines dimensions sont également proposées. Ces dernières fonctions ne peuvent être utilisées qu'en conjonction de certaines dimensions et restent très spécifiques à certains types d'énoncé. Par exemple, pour la dimension des gestions des obligations sociales, on trouve des actes de salutations, de remerciements ou encore d'excuses.

1.3.3. Étiquetage automatique en actes de dialogue

Les différents schémas d'annotation présentés dans la section 1.3.2 permettent de modéliser la structure de surface du dialogue. Dans le cadre du TAL, la problématique se posant immédiatement est celle de la prédiction automatique de cette structure. Le but est alors de prédire automatiquement la séquence d'actes de dialogue pour une conversation donnée, c.-à-d. qu'on souhaite étiqueter chaque tour de parole avec un acte de dialogue. Ceci correspond à une tâche d'étiquetage de séquence, comme peut l'être une tâche d'étiquetage en parties du discours. Cependant, contrairement à l'étiquetage en parties du discours où chaque mot en entrée est associé à une étiquette en sortie, dans le cas de l'étiquetage en actes de

dialogue les entrées sont plus complexes car chacune constituée d'une séquence de plusieurs mots. Il est donc nécessaire de construire des modèles de prédiction capables de prendre en compte cette problématique.

Étant donné qu'une conversation est une séquence ordonnée dans le temps d'énoncés et que la tâche revient à prédire la séquence correspondante d'actes de dialogues, les modèles d'apprentissage utilisés sont généralement des modèles prenant en entrée des séquences tels que les champs aléatoires conditionnels (CRF) ou les réseaux de neurones récurrents (RNN). En effet, ces modèles permettent de prendre en compte de manière ordonnée les éléments d'une séquence. Les méthodes obtenant des résultats à l'état de l'art se basent sur des RNN hiérarchiques (c.-à-d. un premier niveau de RNN encodant chaque tour de parole, et un second niveau avec un autre RNN utilisant en entrée la séquence de tours encodés) en ajoutant potentiellement des mécanismes d'attention afin de traiter la séquence d'énoncés en entrée. De plus, l'ajout d'une couche CRF afin de traiter la sortie du réseau permet également d'améliorer la qualité des résultats.

De manière générale, deux corpus sont utilisés afin d'évaluer les performances des étiqueteurs en actes de dialogue. Le premier est SwitchBoard [GHM92; JSB97] qui est donc annoté en suivant une version modifiée du schéma d'annotation Damsl. Le second est le corpus Mrda [Shr+04] qui a été construit à partir de 75 heures de conversations orales provenant de 75 réunions différentes. Ce corpus est également annoté en utilisant une version modifiée du schéma d'annotation Damsl. Les trois systèmes suivants obtiennent des résultats proches de ou à l'état de l'art :

RNN-3-Utterances [Bot+18] Cette méthode construit dans un premier temps une représentation à partir des caractères des énoncés en utilisant un réseau Long-Short Term Memory (LSTM). Une fois les représentations des énoncés construites, un RNN est appliqué en donnant en entrée trois énoncés passés en plus de l'énoncé courant. La dernière sortie du RNN est utilisé pour la couche de décision.

BiLSTM-CRF [Kum+18] Cette méthode se base sur un réseau totalement hiérarchique avec un premier niveau construisant les représentations des énoncés à partir des mots et un second niveau utilisé pour traiter la conversation. Les deux niveaux utilisent des LSTM bidirectionnels. En sortie du second niveau, une couche CRF est utilisée pour réaliser la prédiction de la séquence d'actes de dialogue.

CRF-ASN [Che+18] Cette méthode est très similaire à celle utilisée dans le BiLSTM-CRF. À la place des LSTM, ils utilisent des Gated Recurrent Units (GRU) et ils construisent également une représentation à partir des caractères afin de prendre en compte les mots hors vocabulaire. La plus grande différence se trouve cependant juste avant la couche CRF où des mécanismes d'attention sont introduits.

La table 1.3 présente les résultats de ces trois systèmes sur les deux corpus

Modèle	Exactitude (en %) SwitchBoard	Mrda
RNN-3-Utterances	77,3	–
BiLSTM-CRF	79,2	90,9
CRF-ASN	81,3	91,7

Table 1.3. – Performances des modèles états de l'art sur la tâche de prédiction des actes de dialogue

SwitchBoard et Mrda. La métrique utilisée afin d'évaluer les performances est l'exactitude. Ces résultats nous permettent de constater que les différentes méthodes parviennent à globalement réaliser de bonnes prédictions avec des scores d'exactitude autour des 80% sur SwitchBoard et autour des 90% sur Mrda. Ceci est un bon indicateur des performances que l'on peut atteindre lors de la réalisation d'un étiqueteur en actes de dialogue, même dans le cas où les données seraient différentes.

1.3.4. Conclusion

On a pu voir qu'à travers le temps, la notion d'acte de langage puis d'acte de dialogue a bien évolué afin de se préciser dans un premier temps, pour ensuite essayer de s'intéresser davantage aux aspects de communication dans les conversations. Ceci en particulier a été réalisé afin de mieux correspondre aux besoins pratiques constatés empiriquement sur des tâches exploitant les actes de dialogues. Cependant, on peut constater que malgré cette évolution, la théorie des actes de langage introduite par Austin [Aus62] puis complétée par Searle [Sea75] est toujours présente à la base de tous les schémas d'annotation en actes de dialogue.

On peut également constater que la construction d'un schéma d'annotation en actes de dialogue n'est pas aisée, même si les schémas existants permettent d'ores et déjà de construire des modèles de dialogues pouvant être utilisés avec succès dans diverses tâches : traduction automatique [AMR95 ; Rei+96 ; Kum+08], systèmes de dialogue [LRH05 ; Kha+18], reconnaissance de la parole [Tay+98]. De plus, bien que les différents schémas d'annotation modélisent avec des précisions variant d'un schéma à un autre les différentes interactions présentes dans les dialogues, la notion d'acte de dialogue reste la même. Ceci nous permet de définir de manière simple les actes de dialogue comme étant des actes de langage utilisés dans le contexte de dialogue. De manière plus précise, on peut définir un *acte de dialogue* comme étant une fonction de communication d'un énoncé portant l'intention du locuteur dans le but de faire évoluer les connaissances d'un ou plusieurs participants de la conversation. Quel que soit le schéma

d'annotation utilisé, les actes de dialogue permettent de construire un premier niveau d'analyse du discours. Ce premier niveau est particulièrement intéressant lorsque les tâches cibles ne nécessitent pas une structure générale du dialogue mais simplement des informations très locales sur le rôle structurel d'un énoncé dans le dialogue. Il est cependant important de prendre en compte le fait que les actes de dialogue ne sont pas suffisants pour comprendre l'organisation du discours et plus particulièrement, quelles sont les interactions présentes entre les différents énoncés. Dans les sections ci-dessous, nous verrons comment ces aspects-là peuvent être mieux pris en compte.

1.4. Structure locale du dialogue

Les actes de dialogue offrent une manière simple de construire un premier niveau de structure du discours. Cependant, l'acte de dialogue d'un énoncé seul ne permet pas de retrouver les autres énoncés du dialogue avec lesquels cet acte de dialogue agit ou réagit. Une manière naïve de faire cela serait de simplement considérer qu'une conversation est une séquence d'énoncés où chaque énoncé est une réaction à l'énoncé précédent. Or, une conversation ne peut pas être simplement vue comme étant une simple séquence. Ceci est dû au phénomène d'enchevêtrement de sous-dialogues, aux références à des énoncés lointains ou au fait que les différents participants peuvent acquiescer, faire des erreurs ou corriger leur interlocuteur ce qui interrompt le flux séquentiel du dialogue.

1.4.1. Paires adjacentes

L'une des premières approches permettant de mettre en lumière des formes de structures locales dans une conversation est un concept nommé *paires adjacentes* (*adjacency pairs* en anglais) proposé par SCHEGLOFF et SACKS [SS73]. Une *paire adjacente* est une unité de la conversation constituée de deux tours de parole, chacun étant énoncé par un locuteur différent. Ces deux tours de parole ont la particularité d'être liés fonctionnellement de manière à ce que le premier tour de parole requiert la présence d'un certain type de second tour de parole. En conséquence, en produisant un premier énoncé, le locuteur espère provoquer la production d'un second énoncé réagissant au premier de la part de son interlocuteur. Dans le cas où ce second énoncé n'est pas produit, ou n'est pas celui attendu, cela produit une anomalie dans la conversation qui est probablement due à une incompréhension, à de l'impolitesse ou à une non volonté de dialoguer de la part du second locuteur. Voici quelques exemples de paires adjacentes « classiques » :

1. — Bonjour ! (Salutation)
   — Ah, salut ! (Salutation)
2. — Quel temps fait-il demain ? (Question)

— Il pleut à priori. (Réponse)

3. — Tu veux venir au cinéma ce soir ? (Offre)
   — Non, merci. (Acceptation/Rejet)
4. — Vous êtes inscrits à la conférence. (Renseignement)
   — D'accord. (Acquiescement)

Dans ces quatre exemples, on peut constater que l'absence du second tour de parole serait inattendu, en particulier dans le cas où le premier énoncé exprime explicitement une volonté du locuteur à attendre une action de la part de son interlocuteur, donner une réponse par exemple.

Les paires adjacentes peuvent être vues comme étant des groupements d'actes de dialogues dans lesquels les actes de dialogue sont explicitement liés entre eux pour former des paires d'énoncés ayant un lien fort dans le discours conversationnel.

Généralement, les deux énoncés qui constituent une paire adjacente se suivent directement. Cependant, ce n'est pas toujours le cas, par exemple lorsqu'il y a des enchevêtrements de sous-dialogues ou des moments de réflexions de la part d'un des locuteurs. Dans ces cas là, l'identification des paires est beaucoup plus difficile. Or il peut être intéressant d'identifier ces paires correctement afin de valider la prédiction d'un acte de dialogue dans le contexte local de la conversation. En effet, WRIGHT [Wri98] a montré que le fait de prendre en compte des actes passés bien sélectionnés, plutôt que de simplement prendre les quelques actes immédiatement avant, permet d'améliorer les résultats de la classification de l'acte de dialogue courant. Une heuristique qui permet d'identifier ces paires a été proposée par MIDGLEY et al. [MHM09] en utilisant des méthodes de segmentation du dialogue.

Les paires adjacentes sont une caractéristique très importante du discours qui est à la base des méthodes de désenchevêtrements de sous-dialogues dans les conversations. En effet, le but de ces méthodes peut être vu comme le fait de reconstruire une chaine de paires adjacentes cohérentes. Ces problématiques d'enchevêtrements de sous-dialogues sont particulièrement présentes dans le contexte de conversations écrites, notamment s'il y a beaucoup de participants, et de nombreux travaux ont essayé de répondre à cette problématique en reconstituant les relations entre paires d'énoncés [EC10 ; MAR12 ; Jia+18]. Les paires adjacentes, en particulier les paires « Question-Réponse » ont également de multiples applications directes que ce soit par exemple pour faire du résumé automatique [MSR07] ou directement identifier les couples questions et réponses dans le cadre de conversations d'assistance en ligne [HLA19].

1.4.2. Actes de la conversation

Les paires adjacentes sont une première approche permettant de lier ensemble deux énoncés. Toutefois, il parait clair que cette approche peut être améliorée car

se limitant à deux énoncés seulement. De plus, les paires adjacentes possibles ne sont pas définies formellement.

Dans le but de mieux modéliser les structures locales du discours conversationnel, TRAUM et HINKELMAN [TH92] proposent de compléter la théorie des actes de langage afin de mieux prendre en compte le contexte dans lequel les énoncés sont produits. En outre, ils remettent en question certaines hypothèses utilisées dans des travaux sur la théorie des actes de langage. Ces hypothèses sont les suivantes :

1. Les énoncés sont entendus et compris correctement par le destinataire, de plus, les deux locuteurs s'attendent à ce que ce soit le cas ;
2. Les actes de langage sont mis en œuvre par un agent seul qui est le locuteur. Le destinataire est uniquement présent passivement ;
3. Chaque énoncé ne peut avoir qu'un seul acte de langage.

TRAUM et HINKELMAN [TH92] considèrent donc que ces hypothèses sont trop restrictives lorsqu'elles sont confrontées à des conversations ayant réellement lieu dans la vie de tous les jours. Ils proposent donc de modéliser le discours à l'aide d'*actes de la conversation* qui sont un ensemble d'actions entre interlocuteurs et ne se contentent pas de simplement décrire les fonctions illocutoires des énoncés de manière isolée. Au contraire, ces actes ont pour but de décrire le rôle dans le discours d'un groupe d'énoncés. Les actes de la conversation ont également pour objectif de mieux évaluer à quel point est-ce que le message à faire passer dans une conversation est effectivement passé. Ils construisent donc une taxinomie qui catégorise les actes en quatre classes. Contrairement aux actes de dialogue, ces différentes classes ont des portées différentes et s'intéressent à des niveaux différents de la structure du discours, allant du simple groupe de mots dans un énoncé à un ensemble de plusieurs énoncés. Ces actes sont :

— les *actes de langage fondamentaux* qui correspondent à des actes de langage « classiques » tels que *Informer*, *Demander* ou *Promettre*. Ces actes sont au niveau de ce qu'ils appellent des unités du discours (UD). Le schéma proposé par TRAUM et HINKELMAN [TH92] ne suit pas la définition habituelle des UD. En effet, une UD est ici un premier énoncé potentiellement suivi de plusieurs énoncés de chacun des participants ayant pour but de valider que l'acte de langage soit mutuellement compris.

— les *actes d'argumentation* qui sont construits à partir de plusieurs actes de langage fondamentaux. Ces actes regroupent donc plusieurs UD et permettent d'identifier des actes de résumé, de clarification ou d'élaboration par exemple.

— les *actes de synchronisation* qui correspondent aux rôles des énoncés au sein des UD et sont donc au niveau des énoncés. Ils constituent un sous-ensemble d'actes de dialogue ayant pour objectif de vérifier que les différents interlocuteurs se comprennent. Ces actes peuvent être des actes

d'amorces, de continuation, d'acquiescement, de demande d'acquiescement, d'annulation ou de correction.

— les *actes de prise de parole* qui correspondent aux actions permettant de prendre, garder ou rendre la parole. Ces actes servent à décrire des phénomènes à un niveau inférieur des énoncés et il peut donc y avoir plusieurs de ces actes par énoncés.

Parmi les actes décrits, les actes d'argumentation permettent donc d'isoler une séquence d'énoncé dans une conversation ayant un objectif commun. Ils se rapprochent donc de la notion de paires adjacentes tout en ne se limitant pas à un couple d'énoncés. En effet, ces actes regroupent plusieurs énoncés qui ont un objectif commun dans le discours du dialogue.

1.5. Analyse profonde du discours conversationnel

Les structures vues dans la section précédente permettent d'obtenir des analyses du discours allant à un niveau au-dessus du simple énoncé. Cependant, ces modèles ne cherchent pas à obtenir une structure globale du discours. En particulier, aucun lien entre les différentes structures locales n'est exhibé et il n'est par conséquent pas possible de déterminer quels sont les différents sous-dialogues d'une conversation.

Deux approches différentes ont été étudiées dans le but de réaliser une analyse profonde du discours conversationnel. Une première approche consiste à se baser sur les théories existantes en analyse du discours dans le cadre de monologues. Ces analyses permettent d'obtenir pour chaque unité du discours ses relations discursives vis-à-vis des autres unités. Ces diverses théories sont présentées dans la section 1.5.1. Il n'est cependant pas possible de simplement appliquer ces méthodes telles quelles sur des dialogues car il existe des phénomènes n'apparaissant que dans les dialogues. Plusieurs travaux existent dans lesquels une adaptation de ces méthodes est réalisée afin d'ajouter la prise en compte de ces aspects dialogiques. Deux approches seront détaillées dans les sections 1.5.2 et 1.5.3.

La seconde approche consiste à davantage se concentrer sur le type de dialogues étudiés, en particulier le contexte dans lequel ces dialogues sont produits. L'analyse du discours est alors construite en se basant sur les différents types de sous-dialogues identifiés empiriquement qui permettent de résoudre les différents enjeux des dialogues. Cette approche sera détaillée dans la section 1.5.4.

1.5.1. L'analyse du discours dans le cadre de monologues

L'analyse du discours est un problème qui a été énormément étudié dans le cadre de monologues, et il existe une grande variété de théories permettant de modéliser les liens discursifs entre les unités du discours d'un document.

Même si les dialogues sont des documents différents des monologues et contiennent des dynamiques différentes, il reste tout de même de nombreux points communs. Par conséquent, les schémas d'annotation utilisés sur les dialogues se fondent sur des schémas existants sur les monologues. De manière générale, les schémas proposés permettent de construire des structures telles que des graphes ou des arbres. Cette sous-section me permettra donc d'introduire ces différents modèles créés dans le but d'analyser le discours dans le cadre de monologues.

Quel que soit le type de structure considéré, tous les schémas doivent définir une unité de base. Dans le cadre de l'analyse profonde du discours, celle-ci est généralement nommée unité du discours. Leur définition exacte dépend de la théorie considérée, mais de manière générale, ces unités correspondent à des phrases ou à des propositions dans le cas de phrases complexes et permettent de communiquer des informations ou à faire des liens discursifs avec d'autres UD.

Le second aspect qui est commun aux différentes théories est la présence de différentes relations discursives possibles, utilisées pour décrire la manière dont les différentes UD interagissent entre elles dans le discours. La nature exacte des relations varie d'une théorie à une autre mais on y retrouve généralement des relations permettant de modéliser au moins les phénomènes discursifs suivants :

Élaboration : Développe un propos.

Explication : Argumente et justifie un propos.

Narration : Met en lien deux propos décrivant des évènements se succédant temporellement.

Résumé : Reprend un ou plusieurs propos en les résumant.

Contraste : Nuance un propos par une alternative.

Condition : Action ou situation dont l'occurrence dépend d'une autre situation.

Contexte : Introduit du contexte à un propos, ce qui permet de simplifier la compréhension.

Cette liste est loin d'être exhaustive mais permet d'avoir une idée des relations discursives que l'on peut trouver dans l'ensemble des schémas d'annotations (parfois sous une forme différente).

L'une des théories du discours les plus connues est la théorie de la structure rhétorique (*Rhetorical Structure Theory* en anglais) (RST)[MT88]. Cette théorie construit, pour un document donné, des arbres à partir de deux éléments : des séquences continues d'unités du discours (les nœuds) et les relations discursives (les arcs). La structure obtenue est hiérarchique, la racine de l'arbre correspond à l'ensemble des UD du dialogue et chaque niveau de l'arbre correspond à des sous-séquences continues d'UD. Au niveau des feuilles de l'arbre on retrouve alors les UD seules. À chaque niveau, les sous-séquences sont mises en relation avec une autre sous-séquence, l'étiquette de la relation correspond au lien discursif qui existe entre les deux sous-séquences.

Un intérêt des structures construites par la RST provient de la hiérarchie produite par l'annotation. En effet, à différente profondeur de l'arbre, on peut ainsi observer des analyses discursives ayant des niveaux de détails variables. Les niveaux près de la racine permettent alors de mettre en évidence les éléments principaux du discours qui correspondent au squelette principal du discours. Les niveaux près des feuilles permettent eux d'avoir un niveau de détails très fin et ainsi d'obtenir le rôle précis de chaque UD dans le discours.

La théorie de la représentation du discours segmenté (*Segmented Discourse Representation Theory* en anglais) (SDRT) proposée par ASHER et LASCARIDES [AL03] est une autre théorie du discours. À la différence de la RST, la SDRT produit des graphes ce qui permet de modéliser des phénomènes de dépendances multiples. Une autre différence majeure est que les nœuds du graphe ne sont pas des séquences continues d'UD mais simplement les UD en eux-mêmes. En effet, la RST part du principe que la structure est divisée en sous-séquences continues d'unités du discours et qu'il n'est pas possible d'avoir des séquences discontinues. Sur des monologues, ceci est généralement correct, néanmoins sur le dialogue on peut immédiatement observer une limite à cette théorie lorsque l'on essaie de modéliser les phénomènes d'enchevêtrements. La SDRT permet donc d'apporter une solution à ce problème en autorisant des liens discursifs entre toutes les UD et ceci à tout moment.

Comme on peut le constater la structure produite par la SDRT semble être également adaptée à l'annotation de dialogues. Toutefois, la SDRT n'est pas utilisable telle quelle : il est nécessaire de réviser les étiquettes des relations discursives afin de les adapter aux dialogues. Nous verrons dans la section 1.5.2 un corpus sur lequel cette adaptation a été réalisée.

Un autre schéma d'annotation intéressant est celui utilisé dans le Penn Discourse Treebank (PDTB) [Pra+08]. Le Penn Treebank [MSM93] est un corpus construit à l'origine pour étudier des annotations syntaxiques pour l'anglais. Le PDTB est une extension de ce corpus qui ajoute des annotations discursives. Le schéma d'annotation est assez différent de ce qui peut être fait dans la RST ou la SDRT. Dans le PDTB, le principe est d'explicitement utiliser les connecteurs logiques comme marqueurs des relations discursives. Chaque connecteur logique a alors deux arguments qui sont les deux UD ayant un lien discursif. Dans le cas où il n'existe pas de connecteurs logiques entre deux UD à lier ensemble, un connecteur « implicite » est alors ajouté lors de l'annotation. L'ensemble des connecteurs logiques sont également associés à des étiquettes discursives possibles.

Une particularité de ce schéma d'annotation est qu'il est extrêmement lié au lexique employé dans le document. En effet, les relations discursives ne sont pas uniquement associées à une étiquette, elles sont également associées à un potentiel connecteur logique. Ce schéma d'annotation impose peu de contraintes structurelles étant donné qu'on ne cherche pas à explicitement construire un arbre ou un graphe (même si en pratique, un arbre est construit). Tout comme la SDRT, ce schéma a également été adapté pour être utilisé dans le cadre de

dialogue. Nous verrons dans la section 1.5.3 comment ce schéma d'annotation est utilisé sur un corpus de conversations SMS.

Il existe de nombreux autres schémas permettant de modéliser le discours tels que le modèle de discours linguistique (LDM) [Pol+04], le modèle Graphbank [WG05] ou encore les grammaires d'arbres adjoints lexicalisés (connu sous le nom de DLTAG) [For+03]. Toutefois, à ma connaissance, ces différents schémas ne sont pas utilisés dans le contexte de l'analyse du discours conversationnel. Tout comme la RST, ceci est probablement dû à leurs structures en arbre ayant des contraintes qui ne sont pas adaptées aux dialogues.

1.5.2. Schéma d'annotation fondé sur la SDRT : le corpus `Stac`

Les travaux présentés précédemment ne s'appuient pas directement sur les différents modèles théoriques déjà proposés dans le cadre de monologues. Or il parait légitime d'adapter ces approches aux dialogues. En effet, même s'il existe de nombreuses différences entre monologues et dialogues, certaines caractéristiques restent communes dû fait que les deux types de documents ont pour objectifs de communiquer et développer une connaissance. AFANTENOS et al. [Afa+15] ont réalisé de premiers travaux prenant en compte ces aspects-là dans le contexte de dialogues multi-parties. Ces travaux se fondent sur la SDRT [AL03] comme modèle théorique dans le but de construire un schéma d'annotation pour l'analyse du discours dans le dialogue. Le schéma d'annotation a été mis au point sur un corpus de dialogues écrits entre plusieurs participants. Ce corpus, appelé STAC [Ash+16], provient d'une version en ligne du jeu « Les Colons de Catane ».

Dans ce jeu, les différents joueurs ont pour but de collecter des ressources telles que du bois ou du minerai, afin de construire des routes et des colonies. Les joueurs peuvent s'échanger des ressources, et dans le cadre de ces travaux de recherche, les participants ont à leur disposition un tchat leur permettant de discuter de ces échanges. Le corpus est annoté à deux niveaux : une annotation en actes de dialogue et une annotation de la structure discursive des dialogues. Ces deux niveaux sont complémentaires et permettent de construire un graphe orienté dans lequel les sommets sont les actes de dialogue (et les énoncés) et les arcs correspondent aux relations discursives et dialogiques qu'il y a entre ces actes de dialogue. Les arcs construisent alors des couples gouverneur-dépendant. Le choix de la SDRT par les auteurs de STAC est dû au fait que beaucoup d'autres théories telles que la DLTAG, le LDM et la RST partent du principe que la structure du discours se présente sous forme d'arbre. Or dans les conversations multi-parties à leurs dispositions, il est très fréquent que certains énoncés dépendent de plusieurs énoncés passés, en particulier les acquiescements, remettant ainsi en cause la structure en arbre.

Étant donné que les dialogues sont très majoritairement des négociations entre joueurs, les auteurs ont choisi de se limiter à des actes de dialogue très spécialisés pour répondre à cette problématique : `offre`, `contre-offre`, `acceptation`, `refus`

et autre. Le nombre de types de relations discursives et dialogiques est quant à lui plus important et ces types de relations sont également génériques. Le schéma d'annotation et les relations utilisées sont basés sur le schéma d'annotation utilisé par MULLER et al. [Mul+12] sur le corpus ANNODIS, lui-même basé sur la SDRT. Bien qu'il existe beaucoup de similarités entre les relations discursives nécessaires afin de décrire le discours de monologues et de dialogues, la façon dont elles sont utilisées ainsi que la fréquence d'utilisation de ces relations sont très différentes. En effet, les relations Paire-Question-Réponse, Élaboration-Question, Question-Clarification, Acquiescement et Correction ne sont présentes que dans les dialogues. Au contraire, les relations Narration, Localisation-Temporelle et Contexte (*Background* en anglais) sont beaucoup plus rares dans le corpus STAC. Les différences en variété et en fréquence d'utilisation des relations confirment la nécessité de s'intéresser à l'analyse du discours de dialogues.

Contrairement à la prédiction des actes de dialogue qui est une tâche relativement facile, prédire automatiquement les relations discursives est une tâche difficile. En effet, cette tâche requiert une analyse en profondeur des conversations nécessitant de prendre en compte le contenu des énoncés, leurs contextes de productions, des connaissances extérieures aux dialogues mais aussi le fait qu'il peut y avoir plusieurs fils de discussions en même temps dans un même dialogue pouvant rendre ambigu certains énoncés. Cette difficulté est confirmée par le fait que le score d'accord inter-annotateur est plutôt faible. En effet, avec 4 annotateurs différents non experts en linguistique, un Kappa de Cohen de $0{,}72$ est obtenu sur le rattachement des énoncés au bon gouverneur et un Kappa de Cohen de $0{,}58$ est obtenu sur l'étiquetage des relations.

Lors de la création de dépendances en analyse du discours, un énoncé peut avoir des gouverneurs soit passés créant ainsi des *relations rétrospectives*, soit futurs créant des *relations prospectives*. Un aspect intéressant de l'analyse du discours dans des conversations telles que celles présentes dans STAC est que les relations entre tours de paroles de différents locuteurs ne peuvent pas être prospectives [Afa+15]. Ceci est une grande différence par rapport aux monologues ou aux textes rédigés par un seul auteur qui peuvent avoir des liens rhétoriques prospectifs, en particulier lorsque l'auteur anticipe le message principal à faire passer mais introduit ce message uniquement après avoir fait passer un point secondaire en premier. Dans les dialogues étudiés, ce phénomène n'apparait donc pas entre locuteurs différents mais peut cependant apparaitre au sein d'un tour de parole d'un seul locuteur qui peut être décomposé en plusieurs unités du discours ayant chacune des liens discursifs entre elles.

De premières tentatives de prédictions automatiques de la structure discursive ont été conduites dans le cadre de STAC. AFANTENOS et al. [Afa+15] parviennent à obtenir des résultats similaires à ce qui est généralement obtenu sur des monologues. Pour évaluer les prédictions, ils utilisent le rappel, la précision et le score F1. De plus, ils effectuent plusieurs niveaux d'évaluation afin de déterminer dans un premier temps si le rattachement réalisé est correct, puis dans un

second temps si le rattachement ainsi que l'étiquette sont corrects. En évaluant uniquement les rattachements dirigés, sans prendre en compte les étiquettes des relations, un score F1 de $67{,}1\%$ est obtenu. En prenant en compte les étiquettes, le score F1 est cette fois-ci de $51{,}6\%$. Cependant, on peut constater que les performances sont très différentes lorsqu'on ne considère que les relations intra-tours (c.-à-d. les relations discursives entre les différentes unités de discours dans le tour de parole d'un même locuteur) ou que les relations inter-tours (c.-à-d. les relations discursives entre des tours de parole de locuteurs différents). En effet, pour les relations intra-tours, ils obtiennent un score F1 de $86{,}1\%$ sur les rattachements non-étiquetés et de $52{,}1\%$ sur les rattachements étiquetés alors que pour les relations inter-tours ces scores ne sont que de $56{,}1\%$ et $44{,}8\%$ respectivement. Cette différence s'explique par le fait qu'au sein d'un même tour de parole, les locuteurs ont tendance à ne pas produire des discours complexes et donc les rattachements sont généralement entre unités de discours adjacentes. En revanche, la production des tours de parole entre locuteurs est beaucoup plus complexe du fait que les participants ne produisent pas les tours de parole en anticipation de ce qui va être dit mais plutôt en réaction de ce qui a été dit. Ce n'est donc pas les locuteurs qui contrôlent la forme de la structure du discours inter-tours à venir.

1.5.3. Schéma d'annotation fondé sur le PDTB

D'autres travaux ont été conduits par XUE et al. [XSJ16] afin de construire un schéma d'annotation du discours dans le cadre de conversations. Comme pour STAC, les travaux tournent autour de conversations écrites, cependant, cette fois-ci ce ne sont pas des conversations instantanées multi-parties mais des conversations SMS. Bien que ce format de conversations présente quelques différences par rapport aux conversations trouvées dans STAC (non nécessité de répondre immédiatement, seulement deux participants), certains phénomènes sont communs aux deux formats tels le fait qu'un message puisse facilement faire référence à un message lointain étant donné que les participants peuvent voir l'historique des messages ou le fait qu'il puisse y avoir des fils de discussion parallèles. De plus, les messages sont généralement plutôt courts du fait de la nature du médium et des contextes de communication.

Contrairement à précédemment où l'unité de discours considérée était une sous-partie du tour de parole d'un locuteur, cette fois-ci l'unité est directement le tour de parole, c.-à-d. le message dans le contexte des SMS. Les seules relations étudiées sont donc celles inter-tours et non les relations intra-tours. Ceci a pour conséquence que la structure obtenue se focalise beaucoup plus sur les interactions et la dimension dialogique. Cependant, ceci ne signifie pas qu'il n'existe plus de relations discursives usuellement rencontrées dans l'analyse du discours de monologues. En effet, il peut tout de même y avoir plusieurs messages différents, consécutifs ou non, d'un même locuteur où le but est d'élaborer, conditionner ou encore concéder. Les relations discursives utilisées afin de lier les mes-

sages d'un même locuteur sont issues du PDTB [Pra+08], cependant, certaines relations n'existent pas dans le contexte de conversations SMS, comme les relations se rapportant à la narration, alors que d'autres relations spécifiques aux dialogues n'existent tout simplement pas dans le PDTB, comme la correction d'un message précédent. Pour les relations discursives entre messages produits par des locuteurs différents, les auteurs se sont inspirés du schéma d'annotation en acte de dialogue DAMSL, en particulier les actes ayant une fonction de communication retrospectives. Cependant, contrairement au schéma d'annotation DAMSL, les étiquettes ne sont pas associées à un seul énoncé seulement mais à une paire de messages. Ils introduisent également des relations additionnelles n'entrant pas dans les deux catégories précédentes afin de prendre en compte certains messages d'introduction de sujet de discussion, d'obligation sociale ou encore utilisés afin d'attirer l'attention de l'autre locuteur. Ces relations permettent la construction d'un arbre en dépendance de la conversation.

1.5.4. Construction d'une structure hiérarchique

Les paires adjacentes et les actes de la conversation permettent d'identifier des structures très locales du discours conversationnel. Cependant, en plus de regrouper les énoncés ayant une action commune, il serait aussi intéressant d'obtenir des regroupements de plus haut niveau permettant d'identifier le rôle de chaque partie du dialogue vis-à-vis des autres parties.

Dans le contexte de conversations guidées par une tâche générale, LOCHBAUM [Loc98] essaie de reconnaitre la structure d'intention du discours. Il considère que le dialogue est construit à partir de différent sous-dialogues, chacun répondant à une sous-tâche particulière ou correspondant à une forme de correction d'un problème introduit précédemment dans le dialogue. Le dialogue est alors le résultat d'une construction incrémentale d'un plan partagé par les deux participants, dans le but de répondre à une tâche donnée. BANGALORE et al. [BDS08] se basent sur cette vision du dialogue afin d'effectuer une analyse structurelle de ces derniers. Ce plan partagé peut être représenté par un arbre où chaque niveau de l'arbre représente un aspect différent du dialogue : tâches, sous-tâches, sujets, actes de dialogue, énoncés. Cette construction permet entre autres de définir des relations de dominance et de préséance entre tâches et sous-tâches dans le dialogue. Cette structure suppose une segmentation en sous-tâches où le but est de prédire si un énoncé courant fait partie de la sous-tâche courante ou s'il faut ouvrir une nouvelle sous-tâche. Dans la figure 1.5, on peut voir un exemple type de structure obtenue. Cette structuration est donc un bon moyen de modéliser le discours lorsque l'objectif du dialogue est clairement identifié.

Le domaine de l'analyse du discours n'est pas l'unique domaine s'intéressant à établir des analyses profondes du dialogue. En effet, l'analyse de l'argumentation est un domaine de recherche ayant pour objectif de déterminer quels sont les liens entre énoncés permettant d'appuyer et de réfuter une argumentation, en

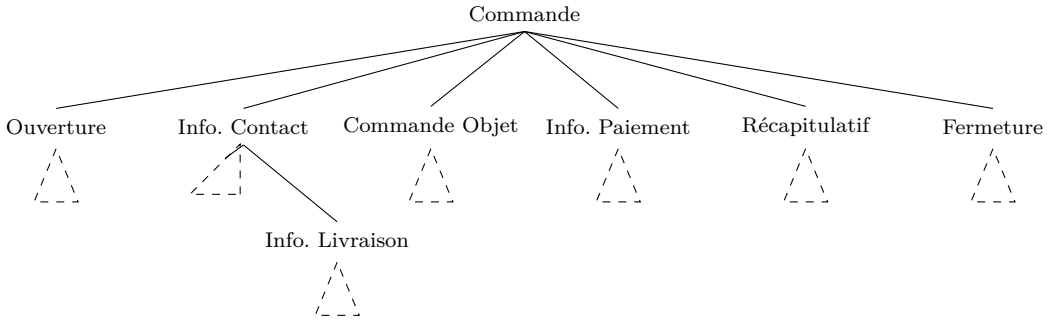


FIGURE 1.5. – Exemple d'arbre pouvant servir à représenter un dialogue guidé par une tâche

particulier dans le cadre de conversations lors de réunions. Il est primordial de bien identifier les différents arguments utilisés lors de réunions afin de réaliser certaines tâches tel que le résumé automatique de réunions [BSL16]. VERBREE et al. [VRH06] proposent de construire des modèles de diagramme argumentatifs basés sur le *Twente Argument Schema* [RHv05]. L'application de ce schéma permet la production d'un arbre dans lequel les nœuds correspondent à des tours de parole, potentiellement composés de plusieurs énoncés, et les arêtes correspondent aux relations entre les tours. Les nœuds sont catégorisés en 5 catégories : affirmation, affirmation faible, question ouverte, question fermée et question oui-non. Les relations sont divisées en 9 catégories : positif, négatif, incertain, requête, spécialisation, élaboration, option, exclusion d'option et soumis-à. On peut constater que les catégories des nœuds correspondent en partie à des sortes d'actes de dialogue très spécialisés au contexte de la réunion. Les catégories des relations ont de manière évidente été construites afin d'explicitement prendre en compte l'argumentation plutôt que le discours. Cependant, certaines relations telles que « élaboration » ou « requête » sont également liées au discours. En pratique, ce modèle a été appliqué sur le corpus de réunion AMI [Car+05] et diverses méthodes ont été étudiées afin d'analyser automatiquement la structure argumentative [Hak09 ; QWK17].

1.6. Discussion

Ce chapitre m'a permis de définir le dialogue qui sera le centre de l'attention dans ma thèse. En particulier, nous avons pu constater qu'un dialogue peut correspondre à des documents produits dans des conditions très variées, que ce soit dû aux modes de communication, au nombre de participants ou au contexte dans lequel le dialogue est produit.

Dans le cadre de ma thèse, je ne vais pas étudier l'ensemble des dialogues possibles. En effet, les dialogues étudiés sont des « tchats » entre deux scripteurs produits dans le cadre de l'assistance clientèle d'une entreprise. Nous verrons dans

le chapitre 3 la nature exacte de ces conversations ainsi que le corpus construit à partir de celles-ci.

Dans ce chapitre, nous avons également vu ce qu'est l'analyse du discours dans le contexte des dialogues. L'analyse du discours est un problème très étudié, en particulier sur les monologues. Toutefois, les modèles appliqués sur les monologues ne peuvent pas être simplement appliqués sur les dialogues et nécessitent donc des adaptations ou l'utilisation d'approches différentes.

Contrairement aux monologues, les dialogues contiennent des interactions entre plusieurs participants. Les énoncés produits par les participants vont alors avoir une influence sur les énoncés des autres participants : réponses, spécifications, ordres, affirmations, acquiescements, etc. Les théories et schémas d'annotation utilisés sur les monologues ne sont pas construits afin de prendre en compte ces spécificités des dialogues et des solutions propres aux dialogues ont donc été proposées. Sans entrer de nouveau dans les détails, il existe deux niveaux d'analyse : un premier niveau portant sur les fonctions de communication des énoncés du dialogue (actes de dialogue) et un deuxième niveau, correspondant davantage à ce qui peut se faire sur les monologues, qui cherchent à construire une structure discursive du dialogue permettant de lier ensemble les différents énoncés à l'aide de relations discursives et dialogiques.

Dans le cadre du TAL, ces deux niveaux n'ont pas reçu la même quantité d'attention du fait que la tâche de prédiction d'actes de dialogue est plus simple : les annotations sont plus rapides à produire manuellement et sont plus simple à prédire automatiquement. En outre, les schémas d'annotations utilisés sont tous relativement semblables et standardisés. Ceci est loin d'être le cas dans le cas de l'analyse profonde du discours conversationnel où il existe encore de très nombreux schémas d'annotations possibles, et ces schémas sont tous complexes à mettre en place.

La problématique centrale de ma thèse est de réussir à produire des représentations du discours conversationnel en étudiant les interactions se produisant dans le dialogue et en utilisant le moins possible de données annotées discursivement. Les différentes approches se fonderont tout de même sur des schémas d'annotations en actes de dialogue et en structures discursives adaptés à mes données, en particulier pour pouvoir évaluer les représentations obtenues.

Avant de pouvoir entrer directement dans le vif du sujet, dans le chapitre suivant je vais tout d'abord étudier comment peuvent être apprises par un algorithme des représentations d'unités linguistiques. En outre, nous verrons également comment évaluer ces représentations afin de garantir qu'elles modélisent correctement différentes propriétés linguistiques (par exemple, le sens des mots). Dans le cadre de mes travaux, cette dernière question sera primordiale car je dois être capable de déterminer si les représentations manipulées modélisent bien les interactions dans le dialogue.

Chapitre 2.

Apprentissage de représentations

Sommaire

2.1 Introduction 58
2.2 Du mot au plongement de mots 60
2.3 Amélioration des plongements de mots 64
2.3.1 Les mots hors vocabulaire 65
2.3.2 Des solutions pour la polysémie 66
2.4 Les plongements pour d'autres types d'unités linguistiques 68
2.5 Propriétés attendues et évaluation des représentations 70
2.5.1 Évaluation des plongements de mots 71
2.5.2 Évaluation des plongements de phrases 76
2.6 Discussion 78

2.1. Introduction

Le chapitre précédent m'a permis d'introduire ce qu'est un dialogue et la manière dont son discours conversationnel est modélisé. Ces modélisations sont des représentations explicites du discours, construites pour et par des humains, qui permettent de retrouver les fonctions de communication des énoncés et les relations dialogiques qui existent avec les autres énoncés.

Des représentations explicites d'unités linguistiques sont utilisées pour de nombreux autres niveaux d'analyse telles que la syntaxe (construction d'arbres syntaxiques) ou la sémantique (construction de cadres sémantiques). Un objectif de ces annotations est généralement de servir d'étape à une tâche applicative pouvant avoir besoin d'informations sur la syntaxe, la sémantique ou le discours pour être résolue. Par ailleurs, ces annotations sont généralement produites afin de réaliser des traitements automatiques, qui pourront générer automatiquement ces annotations sur de nouvelles données.

Une limite de telles approches est qu'elles requièrent la construction de guides d'annotation suivie de la réalisation d'une annotation manuelle d'un corpus — ce qui peut être très couteux en temps et en argent. Par ailleurs, il existe un autre

inconvénient à ces représentations explicites : elles peuvent oublier des phénomènes (morphologiques, syntaxiques, sémantiques ou discursifs) à prendre en compte. Ces oublis peuvent être soit dû à la nature du corpus de référence qui peut être trop spécialisé, ou simplement dû au fait que l'humain n'a pas encore une compréhension parfaite de l'objet linguistique manipulé. Par conséquent, on se retrouve alors avec une représentation qui n'est pas optimale et qui peut ne pas permettre de résoudre certaines tâches se fondant sur cette représentation.

En outre, ces représentations explicites ne sont pas toujours faciles à utiliser dans un algorithme d'apprentissage. En effet, ces représentations sont généralement construites dans le but d'être intelligibles par l'humain. Ceci lui permet d'estimer la qualité de la représentation. Cependant, une machine n'a pas naturellement les capacités de déduction d'un humain et il est donc nécessaire de produire des algorithmes essayant de reproduire le raisonnement d'un humain, tâche qui est très difficile.

Une autre approche du problème de la construction de représentations des unités linguistiques est d'essayer de produire des représentations pour et par la machine. Ces représentations doivent alors permettre, potentiellement au prix de l'intelligibilité, de modéliser sous une forme facile à manipuler par un algorithme les mêmes informations que les représentations explicites. Par ailleurs, afin d'éviter d'introduire des biais humains, il serait souhaitable que des algorithmes soient capables de déterminer eux-mêmes les caractéristiques du langage qui permettent de correctement représenter les unités linguistiques.

Toutefois, ces représentations créées par et pour la machine ne viennent pas en remplacement des représentations explicites. Ces dernières peuvent en effet aider à la construction de nouvelles représentations par la machine, que ce soit pour évaluer les représentations obtenues ou tout simplement pour guider la construction de celles-ci en indiquant explicitement des caractéristiques à prendre en compte. Et inversement, ces représentations obtenues par un ordinateur peuvent également être utiles pour concevoir des algorithmes d'apprentissage capables de reconstruire des représentations explicites, interprétables par des humains.

Le mot étant l'unité de base dans de nombreux problèmes étudiés dans le monde du TAL, je vais dans un premier temps me concentrer sur la construction de représentations pour ceux-ci. Dans la section 2.2, je vais décrire les différentes approches qui ont été développées afin d'arriver à une représentation distributionnelle des mots. La section 2.3 décrira les différentes améliorations qui ont été apportées à ces représentations. La section 2.4 donnera une vision des représentations distributionnelles qui existent pour d'autres unités linguistiques. Enfin, la section 2.5 me permettra de mettre en avant les différentes propriétés que ces représentations doivent modéliser, ainsi que la manière dont elles sont évaluées.

2.2. Du mot au plongement de mots

Dans le but de bien comprendre pourquoi il est important de se poser la question de la manière dont est construite la représentation d'une unité du langage, dans cette section je vais développer ce qui a poussé l'utilisation de représentations distributionnelles pour modéliser un certain nombre de propriétés des mots. Ceci permettra alors de constater qu'il n'est pas simple de représenter le langage, même sur des unités du langage qui paraissent très simples aux premiers abords.

Associer une représentation aux mots d'un document est indispensable afin de pouvoir appliquer divers types d'algorithmes nécessaires dans le domaine du TAL. Lorsqu'un humain produit, entend, pense à ou analyse un mot, cela nous paraît facile et naturel. Or, le cerveau passe en réalité par des processus complexes difficiles à expliquer et qui sont toujours en partie incompris par la communauté scientifique. De ce fait, lorsque l'on souhaite faire manipuler des mots par une machine, il n'est pas simplement possible de les prendre en compte de la même manière qu'un humain. Un objectif est toutefois d'essayer d'imiter dans la mesure du possible ce que l'on observe chez l'humain. Ces observations correspondent alors à des propriétés que l'on souhaite modéliser dans les représentations des mots.

La propriété la plus basique est de distinguer des mots ayant des formes différentes, c.-à-d. des séquences de symboles différentes. La méthode la plus naïve est de considérer les mots comme une suite de lettres : la chaine de caractères. Un inconvénient de cette approche est que l'unité de base n'est alors pas le mot mais la lettre. Ceci peut être souhaité dans certaines situations, mais lorsque l'on souhaite analyser une phrase ou un document, l'information importante est généralement la présence ou non d'un mot.

Pour remettre en avant les mots plutôt que les lettres, on peut attribuer un entier unique à chaque mot du vocabulaire utilisé dans les documents manipulés. On obtient alors une représentation différente par forme de mots. Cependant, il est important d'avoir conscience qu'en utilisant une telle représentation, un ordre a été construit entre les mots.

Ceci peut potentiellement être un comportement souhaité, par exemple si on a uniquement les mots « petit », « moyen », « grand », le fait de respectivement donner les identifiants 0, 1 et 2 a un certain sens. Cependant, de manière générale les vocabulaires manipulés sont beaucoup plus importants et il n'y a pas de sens à donner un ordre aux mots. Or pour certains algorithmes, en particulier les réseaux de neurones, cet ordre va être utilisé alors que ce n'est pas souhaité et cela aura des conséquences sur les résultats produits par ceux-ci. Il est donc nécessaire de trouver une représentation traitant tous les mots de la même manière, c.-à-d. sans qu'il y ait un ordre naturel implicite entre les représentations de mots.

Afin de résoudre ce problème, un encodage à un bit non nul discriminant peut être utilisé. L'*encodage à un bit non nul discriminant* (appelé *one-hot encoding* en

anglais et que je nommerai également encodage binaire) consiste à construire un vecteur de la taille du vocabulaire et où chaque composante du vecteur est à 0 sauf celle qui correspond au mot à encoder qui est mise à 1. L'encodage binaire est généralement utilisé pour représenter des données catégorielles dans le domaine de l'apprentissage automatique. Ceci correspond à ce que nous souhaitons faire ici : chaque mot est considéré comme étant une catégorie indépendante des autres catégories.

Cette approche a l'avantage d'être facile à mettre en place et de ne pas créer d'ordre implicite non souhaité entre les mots. Cependant, plusieurs problèmes surgissent :

1. Cette représentation prend beaucoup de place en mémoire étant donné que la taille du vecteur dépend de la taille du vocabulaire (et donc de la taille du corpus).
2. Conséquence du point 1, les vecteurs obtenus n'exploitent pas bien tout l'espace qui est à leur disposition. En effet, les vecteurs sont extrêmement creux, avec une seule composante qui n'est pas à 0.
3. Tous les vecteurs de mots sont équidistants entre eux. Ceci peut être un comportement souhaité dans certain cas, mais généralement, pour des mots apparaissant dans des contextes similaires, en particulier les synonymes, il serait souhaitable que les vecteurs obtenus prennent en compte cela.
4. En lien avec le point 3, on rencontre la question de l'arbitraire du signe [de 16]. Que ce soit pour la représentation binaire, ou pour les autres représentations précédemment décrites, celles-ci ne représentent pas le sens des mots mais les formes des mots.

Le point 1 est un problème plutôt technique mais qui a des conséquences très importantes en pratique. La mémoire étant une ressource limitée, ces représentations peuvent rapidement rendre des algorithmes inutilisables en pratique, en particulier lorsqu'on utilise des corpus volumineux. De plus, d'un point de vue théorique ce n'est pas une représentation satisfaisante car pour un vocabulaire infini, il serait nécessaire de construire des vecteurs de taille infini.

Le point 2, en conjonction avec le point 3, met en évidence le fait que les vecteurs binaires modélisent très peu d'informations sur les mots — une seule en réalité : l'unicité de la forme — alors qu'il serait souhaitable d'utiliser l'espace non utilisé pour représenter d'autres informations, telles que les différentes relations (synonymie, antonymie, hyperonymie, etc.) qu'il peut y avoir entre deux mots.

Par ailleurs, ces représentations ne prennent en compte que le côté arbitraire des mots en ne s'appuyant que sur la forme. Or, dans le cadre de tâche du TAL, il serait souhaitable de modéliser le sens plutôt que la forme. Par exemple, les mots « maison » et « appartement » ont un sens proche, alors que leurs formes n'ont rien en commun. De ce fait, ces deux mots sont interchangeables dans beaucoup

de situations, ce que les vecteurs binaires, ou les autres représentations étudiées auparavant, ne permettent pas du tout de modéliser.

La question de la représentation des sens des mots est relativement ancienne et WordNet [Mil95] permet de déterminer de manière explicite — sous la forme d'un graphe — les relations qui existent entre les différents mots de la langue. Cette ressource est très complète, néanmoins un inconvénient est qu'elle n'est pas utilisable telle quelle par un algorithme d'apprentissage — il est nécessaire que l'humain indique les informations intéressantes à récupérer dans le graphe.

Une autre solution ayant émergée consiste à construire des *représentations distributionnelles* des mots. Ces représentations, aussi appelées *plongements de mots* (*word embeddings* en anglais), s'appuient sur l'idée que l'information contextuelle seule est une représentation viable d'objets linguistiques [Har54 ; Fir57]. En effet, une manière de définir le sens d'un mot est de dire qu'il correspond à tous les contextes dans lequel il peut apparaitre. Par conséquent, si deux mots sont toujours utilisés avec les mêmes mots adjacents (avec une fenêtre à définir), alors on peut considérer qu'ils ont le même sens.

Les plongements de mots sont des mots plongés dans un espace vectoriel $\mathbb{R}^n$ où n est un paramètre à définir. Contrairement aux vecteurs binaires, la dimension n du vecteur ne dépend pas du vocabulaire et peut donc être bien inférieure à sa taille. L'idée des plongements de mots est donc de construire des représentations prenant en compte le contexte de production du mot (c.-à-d. avec quels autres mots ou dans quels types de documents est-ce que ce mot est généralement utilisé), plutôt que de s'appuyer uniquement sur la forme du mot. Pour cela, le but est d'apprendre automatiquement un ensemble de caractéristiques d'un mot permettant d'identifier les similarités entre deux mots.

Plusieurs méthodes ont été proposées permettant, directement ou indirectement, de construire ces plongements de mots. Dans tous les cas, les plongements sont appris dans le but de prendre en compte le contexte de production d'un mot. Toutefois, en fonction des besoins et des méthodes appliquées, le contexte considéré est différent. En faisant variés la taille du contexte, le « sens » modélisé ne sera pas le même. Lorsque le contexte est le document en entier, alors le sens se rapproche alors du thème du document. Au contraire, lorsque le contexte d'un mot ne correspond qu'aux mots adjacents, alors on se rapproche beaucoup plus du rôle syntaxique du mot dans une phrase.

Les plongements de mots construits en considérant de très grand contextes sont surtout utilisés dans le cadre du domaine de la recherche d'information, où l'intérêt est de pouvoir retrouver des documents à partir de mots-clefs par exemple. Les représentations distributionnelles sont ainsi issues de *modèles thématiques* qui sont créés dans le but de découvrir dans un ensemble de documents les différents thèmes y apparaissant. Le modèle précurseur est l'analyse sémantique latente (*Latent Semantic Analysis* en anglais) (LSA) [Dee+90] qui se base sur une matrice d'occurrences qui décrit pour chaque terme le nombre de fois où il apparait dans chaque document. Une décomposition en valeurs singulières

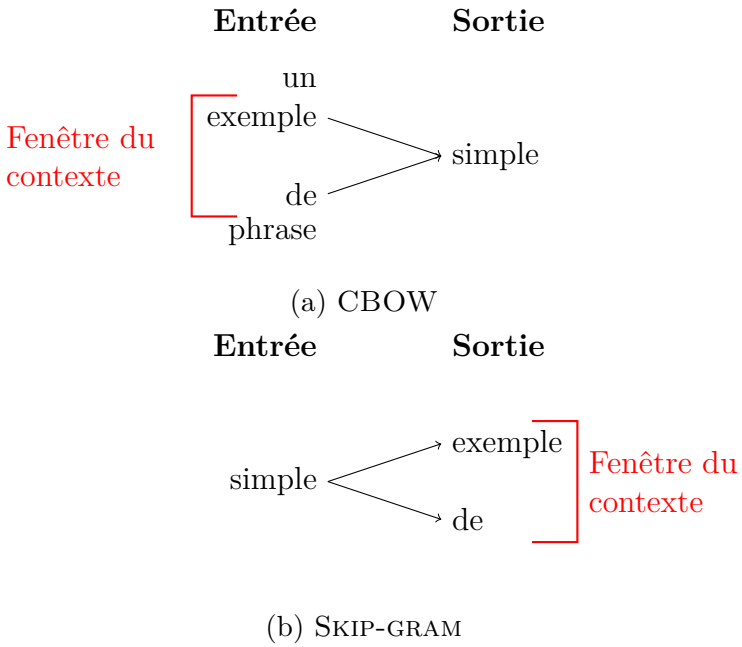
(SVD) est ensuite appliquée sur la matrice ce qui permet de réduire le nombre de dimensions tout en conservant les relations de similarités. D'autres algorithmes similaires à LSA, construisant des modèles thématiques, tel que l'allocation de Dirichlet latente (*Latent Dirichlet Allocation* en anglais (LDA) [BNJ03] ont été élaborés afin d'améliorer l'apprentissage de ces représentations.

Les plongements de mots construits en considérant un contexte plutôt court (au maximum la longueur moyenne d'une phrase) ont été développés dans le cadre du domaine du TAL, où on souhaite manipuler les sens des mots tels qu'on a l'habitude de les considérer en linguistique. Il existe deux familles d'algorithmes permettant de construire des plongements de mots avec des contextes courts, les plongements de mots issus des deux familles ayant en fin de compte les mêmes propriétés.

La première famille utilise des méthodes neuronales et s'appuie généralement sur des modèles de langages neuronaux. De manière plus générale, ces approches s'appuient sur des tâches artificielles (par exemple, la prédiction des mots adjacents) ou concrètes (par exemple, la traduction automatique) pour construire les plongements. Cette approche a été mise en avant par BENGIO et al. [Ben+03] et a pour but de construire des représentations prenant en compte le contexte immédiat autour du mot en utilisant les propriétés des modèles de langages. Plus récemment, l'outil WORD2VEC [Mik+13] a grandement popularisée les plongements de mots. Cet outil propose deux types de modèles : CBOW et SKIP-GRAM. Pour le premier modèle, le but est de prédire un mot donné uniquement à partir d'une fenêtre de mots autour de ce dernier. Pour le second modèle, le but est cette fois-ci de prédire les mots se trouvant autour d'un mot donné. La figure 2.1 présente les idées de fonctionnement de ces deux approches.

La deuxième famille s'appuie explicitement sur les statistiques de cooccurrences des mots. L'un des premiers modèle basé sur ce principe est le modèle *Hyperspace Analogue to Language* proposé par LUND et BURGESS [LB96]. Plus récemment, le modèle GLOVE [PSM14], également basé sur les cooccurrences de mots permet d'obtenir des propriétés similaires à celles obtenues par les modèles WORD2VEC. En effet, les modèles tels que SKIP-GRAM bien qu'ils n'aient pas accès à une matrice de cooccurrences explicite, reconstruisent cette matrice de manière implicite afin de pouvoir prédire les mots adjacents.

De manière générale, dans le domaine du TAL, les plongements de mots prenant en compte des contextes courts sont beaucoup plus utilisés et étudiés du fait qu'ils correspondent davantage aux propriétés attendues, c.-à-d. celles de répondre aux problèmes de similarités sémantiques entre mots. Dans mes travaux, ce sont donc ces plongements-ci qui m'intéressent pour donner des représentations aux mots des dialogues.



(a) CBOW

(b) Skip-gram

Figure 2.1. – Idées du fonctionnement des modèles CBOW et Skip-gram

2.3. Amélioration des plongements de mots

Les plongements de mots ont permis de résoudre plusieurs problèmes. Le plus important étant la prise en compte du sens des mots en se basant sur les contextes dans lesquels ceux-ci sont produits. Par ailleurs, le papier « A Unified Architecture for Natural Language Processing » de Collobert et Weston [CW08] a permis d'établir que ces plongements sont des représentations permettant d'améliorer les performances sur des tâches en aval.

Toutefois, deux problèmes restent non résolus par ces approches :

1. Les mots inconnus ne sont pas pris en compte dans ces modèles. Un *mot inconnu* est un mot qui n'est jamais rencontré dans le corpus d'entrainement. De fait, si un mot n'a jamais été rencontré lors de l'entrainement des plongements de mots alors aucune représentation du mot ne peut être construite.
2. Les plongements de mots tels qu'ils sont construits permettent de ne donner qu'une seule et unique représentation — et donc un unique sens — par mot. Or ceci est problématique, par exemple dans le cas de mots polysémiques qui sont indistinguables et auront donc la même représentation.

Des solutions ont été apportées afin de résoudre ces deux problèmes en améliorant les modèles de plongements de mots. La sous-section 2.3.1 présentera plusieurs solutions apportées au problème des mots hors vocabulaire. La sous-section 2.3.2 présentera des solutions permettant de prendre en compte le fait qu'un mot peut avoir plusieurs sens.

2.3.1. Les mots hors vocabulaire

La problématique des mots hors vocabulaire (aussi appelés inconnus) est ancienne et se posait également dans le cas des encodages à un bit non nul discriminant. De nombreuses solutions ont donc été proposées au fil du temps, et c'est également le cas pour les plongements de mots. Dans le cadre des dialogues que je manipule dans ma thèse, cette problématique est très présente. En effet, les dialogues manipulés sont des « tchats » où la production des messages est spontanée. De ce fait, il est très fréquent d'y rencontrer des fautes d'orthographe. Celles-ci vont produire de nouvelles formes de mots qui peuvent n'avoir jamais été rencontrées dans le corpus d'entrainement, produisant ainsi des mots inconnus. Par exemple, si dans le corpus d'apprentissage le mot « téléphone » est présent mais pas le mot mal orthographié « tlephone », alors on ne sera jamais capable de produire un plongement de mot pour ce dernier si on ne prend pas en compte la problématique des mots inconnus. Les solutions qui suivent ont généralement été pensées pour donner des représentations à des mots rares, mais s'appliquent tout aussi bien aux mots mal orthographiés.

Dans le contexte des plongements de mots, plusieurs approches ont été proposées afin d'obtenir des représentations pour ces mots malgré leur absence durant l'entrainement. Une approche simple est de construire une représentation commune à tous les mots inconnus en considérant certains mots comme étant inconnus durant l'entrainement (mots apparaissant très peu de fois dans le corpus d'entrainement par exemple). Ceci permet d'obtenir une représentation non aléatoire et ayant un certain sens vis-à-vis des autres plongements de mots. Cependant, cette solution n'est clairement pas satisfaisante étant donné qu'elle regroupe ensemble tous les mots « inconnus », même s'ils ont des sens et des rôles grammaticaux totalement différents. Afin d'être réellement capable de construire une représentation unique pour chaque mot inconnu, de nombreux travaux se sont donc fondés sur les sous-mots, morphèmes ou encore directement les lettres des mots.

Une première approche consiste donc à directement construire les plongements de mots en partant des lettres du mot. Plusieurs travaux se basent sur ce principe-là en utilisant généralement des réseaux de neurones convolutifs (CNN) et des LSTM afin de reconstruire les mots à partir des lettres [Kim+16 ; Lin+15].

Les *réseaux de neurones convolutifs*, historiquement développés pour l'analyse d'images, sont des réseaux appliquant des convolutions sur les entrées. Un grand intérêt des CNN est qu'ils permettent d'apprendre automatiquement des filtres supprimant les caractéristiques inintéressantes de l'entrée, et ainsi ne garder que les caractéristiques utiles pour répondre à une tâche cible. Sur du texte, cela correspond alors à appliquer une fenêtre glissante sur celui-ci et à déterminer les n-grammes pertinents dans le texte.

Les *réseaux LSTM* [HS97] sont des variantes de RNN. Ces derniers sont des réseaux qui prennent en compte des séquences en entrée en utilisant la sortie des

neurones au temps $t-1$ en entrée des neurones au temps t. Les LSTM constituent un type de RNN permettant de mieux prendre en compte des dépendances lointaines dans les séquences. Ceux-ci sont fréquemment utilisés dans le domaine du TAL car ils permettent de bien modéliser le fait que le texte est une séquence ordonnée de mots ou caractères, et modéliser le fait que certains mots sont liés à d'autres mots dans la séquence.

Ces approches neuronales peuvent également être appliquées sur les lettres d'un mot et ont l'intérêt d'être faciles à intégrer à d'autres réseaux neuronaux existants en remplacement de la couche d'entrée habituellement utilisée pour les plongements de mots. Les plongements de mots sont alors dynamiquement créés à partir des lettres par le réseau de neurones. On peut alors s'attendre à ce que les réseaux soient plus robustes aux variations morphologiques régulières, en apprenant à reconnaitre des suffixes et préfixes par exemple.

D'autres approches essaient de ne pas se limiter uniquement aux lettres mais de s'approcher de la notion de morphèmes en utilisant des n-grammes de caractères pour construire des représentations intermédiaires permettant de reconstituer les mots inconnus [Wie+16 ; Boj+17]. Un grand intérêt de ces dernières approches est qu'elles permettent de construire de manière explicites des représentations pour des sous-mots ayant du sens car apparaissant dans de multiples mots, généralement dû au fait que ce sont par exemple des morphèmes. Ceci permet entre autres de beaucoup mieux reconstruire des plongements de mots pour des mots qui sont simplement des dérivés de mots connus.

Une dernière approche possible consiste à ne pas modifier les méthodes de création de plongements de mots, mais plutôt de modifier les données en entrée des modèles. SENNRICH et al. [SHB16] proposent d'utiliser l'encodage par paires d'octets (*Byte Pair Encoding* en anglais) (BPE) afin de constituer à partir d'un corpus d'entrainement un ensemble restreint de sous-mots qui vont être utilisés pour remplacer les mots par plusieurs sous-mots. Un symbole spécial est ajouté à la fin des sous-mots qui ne sont pas en fin de mots afin de pouvoir reconstituer les mots. Par exemple, le mot « `reconnaissable` » pourrait être mis sous la forme « `re@ connai@ ss@ able` » par l'algorithme. En plus de permettre de supprimer les mots inconnus, cette approche permet également de réduire considérablement la taille du vocabulaire. Ceci peut être très utile pour réaliser des modèles de langages utilisant des modèles neuronaux élaborés se basant sur un très grand vocabulaire.

2.3.2. Des solutions pour la polysémie

Comme énoncé précédemment, un deuxième aspect problématique posé par les plongements de mots est celui des mots polysémiques. Ce point est important car avec les modèles de plongements classiques, des mots polysémiques — avec des sens pouvant aller de la simple nuance à des sens radicalement différents — auront exactement la même représentation. Même si les plongements de mots

sont construits en utilisant le contexte de production des mots, ils construisent un sens moyen pondéré par le nombre d'utilisation des différents sens existants.

Dans le cadre des dialogues, on retrouve naturellement ce problème de la polysémie, que ce soit au niveau des phrases mais également des mots. Par exemple, le mot « oui » dans un dialogue est très fréquemment utilisé mais en fonction du contexte dans lequel il est utilisé, son sens diffère. En effet, il peut :

— marquer l'accord sur une proposition ;
— sous sa forme interrogative, être utilisé pour demander ce que souhaite l'interlocuteur ;
— indiquer que l'on écoute (*backchannel*) ;
— être utilisé de manière négative pour marquer son agacement.

Généralement, on peut espérer que le contexte d'utilisation (phrase interrogative, réponse à une question, etc.) permettra aux algorithmes d'apprentissage de déduire le sens du mot sans pour autant construire des représentations différentes par sens. Néanmoins, il n'est pas garanti que cette déduction soit bien réalisée et donc plusieurs solutions ont été apportées au problème de la polysémie.

Une méthode simple consiste à exploiter d'autres caractéristiques que les seules formes des mots telles que les parties du discours [TML15] lors de la construction des plongements de mots. Une *partie du discours* est une catégorie regroupant des unités lexicales ayant des propriétés grammaticales similaires (nom commun, adjectif, préposition, etc.). Le but est de construire une représentation différente d'un mot donné par partie du discours qu'il peut avoir. Une grande limite de cette approche est qu'elle nécessite une supervision qu'il n'est pas toujours possible d'obtenir facilement. En outre, cette approche ne résout pas intégralement le problème en ne désambigüisant pas les multiples sens possibles pour une même partie de discours donnée.

Une autre approche semblable mais reposant cette fois-ci sur une approche non-supervisée consiste à déterminer les différents sens possibles d'un mot à l'aide de méthodes de groupements (*clustering* en anglais) [RM10 ; Hua+12]. Les groupes sont construits à partir de fenêtres de contexte contenant le mot à désambigüiser, c.-à-d. que toutes les occurrences du mot produites dans un contexte proche serviront à construire un sens possible sous la forme d'un regroupement (*cluster* en anglais). Ceci permet ensuite d'obtenir plusieurs « prototypes » du mot qui sont utilisés en remplacement des mots lors de l'apprentissage des plongements de mots. Cette approche a l'avantage de pouvoir d'elle-même déterminer les différents sens possibles d'un mot. Toutefois, elle nécessite que les groupements effectués soient de très bonnes qualités et de plus, elle ne permet pas de donner des représentations pour des sens jamais rencontrés durant l'apprentissage de celles-ci. Ce dernier cas est d'ailleurs un problème car pour de tels mots, la méthodologie donnera au mot une représentation d'un autre sens ce qui peut potentiellement induire en erreur les algorithmes utilisant ces représentations.

Une autre approche plus récente consiste à créer des plongements de mots dits *contextuels*. Le nom peut paraitre redondant étant donné que tous les plongements de mots sont construits de manière à prendre en compte le contexte de production du mot. Le mot « contextuel » fait ici référence au fait que ces plongements-ci sont construits directement lors de leurs utilisations en fonction des phrases — du contexte — dans lesquelles ils se trouvent. De ce fait, il n'existe plus un plongement par mot mais autant de plongements que de phrases dans lesquelles le mot peut apparaitre. De nombreux modèles ont été proposés tels que CoVe [McC+17], ELMo [Pet+18], BERT [Dev+19] ou encore XL-Net [Yan+19]. Lors de l'apprentissage, ces modèles s'appuient sur diverses tâches, CoVe s'appuyant sur de la traduction automatique alors que BERT se base sur deux tâches simultanées : la reconstruction de la phrase en entrée ayant un mot potentiellement masqué et la prédiction de la phrase suivante. Bien que ces différentes approches ont montré leurs utilités dans de nombreuses tâches, ces modèles ont l'inconvénient d'être très complexes, et nécessitent pour l'entrainement une très grande quantité de données et de très puissantes ressources de calculs. Des modèles pré-entrainés sont proposés mais ne sont pas nécessairement adaptés aux corpus et langues étudiées. Il est cependant possible, et recommandé, de réaliser des ajustements des paramètres des modèles sur les corpus et les tâches dans lesquels on souhaite utiliser les plongements contextuels.

2.4. Les plongements pour d'autres types d'unités linguistiques

Les plongements de mots ont permis de construire des représentations des mots beaucoup plus flexibles et porteuses d'informations pertinentes que ce qui était fait jusqu'à présent. Une évolution naturelle est d'étendre ce modèle de représentations à d'autres unités linguistiques tels que des phrases, des paragraphes ou des documents. La méthode la plus naïve est de simplement réaliser une somme ou une moyenne des plongements de chaque mot de la phrase ou du document. En pratique cette approche donne des résultats étonnamment raisonnablement bons sur de nombreuses tâches. On peut faire l'hypothèse que le fait de faire la somme/moyenne des vecteurs d'une phrase aura pour effet de retourner un vecteur correspondant au sens « le plus fort » dans la phrase. Néanmoins, cette approche-là pose deux problèmes majeurs :

1. Il n'est pas évident que la somme ou la moyenne permettent d'obtenir de bonnes propriétés. En particulier, tous les mots sont considérés de la même manière et ont le même poids alors qu'il est probable que certains mots soient plus importants que d'autres.
2. Le contexte de production n'est pas pris en compte alors que c'est une des propriétés principales des plongements de mots.

Afin de répondre à ces deux problèmes, il est nécessaire de construire des méthodes propres à ces unités linguistiques. La majorité des approches s'inspirent de ce qui se fait dans le cadre des mots en adaptant principalement la manière dont sont traitées les entrées et les sorties. En effet, étant donné que les unités étudiées sont désormais des séquences de mots ou de phrases, il est nécessaire de faire en sorte que les algorithmes utilisés soient capables de prendre cela en compte.

S'inspirant du fonctionnement des SKIP-GRAMS dans WORD2VEC, les *vecteurs* SKIP-THOUGHT [Kir+15] sont des plongements de phrases qui sont construits en essayant de prédire les phrases se trouvant autour des phrases considérées. Les vecteurs FASTSENT [HCK16] se basent sur le même principe mais contrairement à SKIP-THOUGHT qui essaie de reconstruire les séquences de mots des phrases adjacentes, FASTSENT se contente de prédire les sacs de mots des phrases adjacentes.

Une critique qui peut être faite à ces approches est qu'elles considèrent les phrases de manière très similaires aux mots, c.-à-d. qu'elles se contentent de capturer les phrases adjacentes, ce qui constitue un contexte plutôt pauvre. La production d'une phrase va bien entendu dépendre des phrases adjacentes, mais ceci ne prend pas en compte les liens discursifs entre les phrases — tels que la présence de potentiels enchevêtrements dans le dialogue — ni des liens argumentatifs (par exemple les liens d'inférence), ni l'environnement dans lequel est produit la phrase (dialogue, description d'une image, article scientifique, etc.). Par conséquent, il est important que les représentations modélisent correctement la place de la phrase dans le discours, les opinions et sentiments, ou encore les liens d'inférence avec d'autres phrases.

Dans le but de mieux modéliser les liens qui peuvent exister entre certaines phrases, récemment de nombreux travaux ont portés sur la construction de plongements de phrases modélisant l'inférence. La tâche d'inférence en langage naturel a pour but de prédire, à partir d'un postulat, si une hypothèse est fausse (contradiction), vraie (implication) ou neutre (indéterminé).

INFERSENT [Con+17] est un modèle se basant sur cette tâche afin de construire des plongements de phrases. L'intérêt d'une telle approche est d'aller plus loin que le simple contexte de production de la phrase en y incluant explicitement des notions de compréhension du langage naturel, étant donné qu'il est nécessaire de comprendre ce qui est dit dans la phrase afin de pouvoir déterminer la nature de l'inférence.

De la même manière qu'il existe des travaux sur les phrases, il existe également des travaux à l'échelle de documents entiers ou de paragraphes. Ceux-ci s'inspirent de ce qui se fait au niveau des mots et des phrases mais en essayant d'adapter les méthodes aux problématiques propres à ces unités linguistiques.

Il n'est pas raisonnable de mettre tous les types de documents dans une seule et même catégorie. Sur les phrases, il existe déjà une variabilité assez importante en fonction de l'environnement de production. De manière générale, plus les unités

sont grandes, plus la variabilité va augmenter et il sera d'autant plus nécessaire de la prendre en compte. Les documents peuvent être des articles (scientifiques et journalistiques), des dialogues, des *tweets* ou encore des chapitres et sections d'un livre qui auront des formes, contenus et registres de langues très différents. Il est donc nécessaire d'adapter les méthodes de construction de représentations aux types de documents considérés. Des approches neuronales ont été développées afin de modéliser différents types de documents : des paragraphes et articles complets [LM14] ou des *tweets* [VVR16] par exemple.

Contrairement aux phrases ou aux mots, il peut être difficile de modéliser le contexte de production d'un document en entier. En effet, en fonction du type de document (par exemple, un dialogue), le contexte s'appuie très majoritairement sur des facteurs extérieurs au document, c.-à-d. l'environnement de production et la raison pour laquelle le document est produit. Ces informations ne sont pas toujours disponibles et lorsqu'elles le sont, c'est généralement uniquement de manière partielle sous forme de métadonnées.

Dans le cadre plus spécifique des dialogues, on y retrouve les différents niveaux d'unités linguistiques : les mots, les phrases (sous la forme d'énoncés et de tours de parole) et le document. La construction de plongements pour ces unités va donc devoir s'adapter aux dialogues et aux tâches que l'on souhaite réaliser dessus. En particulier, il pourra être nécessaire prendre en compte le discours conversationnel lors de la construction de plongements. L'un des objectifs de ma thèse va donc être de déterminer si les représentations usuellement utilisées en TAL sont adaptées aux dialogues.

2.5. Propriétés attendues et évaluation des représentations

Jusqu'à présent, nous avons surtout vu pourquoi et comment les plongements ont été créés. En revanche, une question légitime à se poser est de savoir quelles sont les propriétés capturées par ces représentations. Les *propriétés* d'une représentation distributionnelle correspondent à des observations que l'on peut faire sur les unités lexicales que l'on souhaite également modéliser dans leurs représentations distributionnelles (par exemple, être capable de déterminer si deux mots sont des synonymes ou être capable de dire qu'une phrase est une implication d'une autre).

Un autre aspect à étudier est de savoir comment comparer la qualité des différentes approches proposées dans le but de savoir quels algorithmes permettent d'obtenir les meilleures représentations. La qualité d'un modèle de plongements va dépendre de métriques d'évaluation qui permettent de déterminer si des propriétés souhaitées sont correctement modélisées.

Dans mes travaux sur les dialogues, je vais chercher à construire des représentations du discours conversationnel pour plusieurs types d'unités lexicales. Une

question centrale sera alors de savoir comment évaluer celles-ci afin de déterminer si elles sont en effet capables de correctement modéliser le discours conversationnel. Avant de pouvoir étudier en détail cette question, il est nécessaire de savoir comment sont évaluées les différentes unités lexicales usuellement étudiées dans le domaine du TAL afin de mettre en évidence les propriétés généralement souhaitées, mais également les limites rencontrées par de telles approches.

Je vais dans un premier temps dans la sous-section 2.5.1 me concentrer sur les cadres d'évaluations utilisés pour évaluer les plongements de mots. Cette unité lexicale est probablement celle pour laquelle il est le plus facile de définir les propriétés souhaitées et donc pour laquelle les cadres d'évaluation sont relativement anciens et matures. Dans un second temps, je m'intéresserai dans la sous-section 2.5.2 à l'évaluation des plongements de phrases, unité lexicale qui est déjà beaucoup plus difficile à évaluer.

2.5.1. Évaluation des plongements de mots

L'évaluation des modèles de plongements de mots est une question qui est primordiale afin d'évaluer si ceux-ci modélisent correctement les propriétés attendues. Cependant, ces dernières années, des critiques ont été formulées sur la manière dont ces évaluations sont conduites, en particulier à propos des méthodologies utilisées [Far+16]. Par ailleurs, de nombreux travaux, en particulier en lien avec les sciences cognitives, mettent en avant de nouvelles propriétés à évaluer.

Des approches intrinsèques et extrinsèques sont communément utilisées pour évaluer la qualité de modèles de plongements de mots. Le premier type d'approche consiste à réaliser l'évaluation en cherchant à explicitement valider des propriétés que les plongements de mots sont censés satisfaire. Avec une évaluation extrinsèque, la qualité des plongements de mots est alors jugée par les performances réalisées sur une tâche en aval. Dans cette section, je m'intéresse principalement aux propriétés attendues des plongements de mots. De ce fait, dans les sous-sections qui suivent, je vais uniquement présenter les différentes évaluations intrinsèques généralement utilisées.

2.5.1.1. Association et similarité sémantique

Une propriété qui est très attendue lorsque l'on manipule des mots est que l'on s'attend à ce que des synonymes aient des représentations proches. Un avantage des plongements de mots est que ce sont des vecteurs dans $\mathbb{R}^n$ et que l'on sait donc facilement donner une existence à cette notion de « proches » à l'aide de calculs sur les vecteurs. On peut en effet utiliser diverses distances et mesures de similarités qui permettent de déterminer si deux vecteurs sont similaires ou non. Sur les plongements, la mesure qui est généralement utilisée est la *similarité cosinus* qui est le cosinus de l'angle entre deux vecteurs. On obtient alors une

valeur dans $[-1; 1]$ où 1 signifie que les deux vecteurs sont similaires, 0 que les vecteurs sont indépendants et -1 que les vecteurs sont diamétralement opposés.

Afin d'évaluer ces aspects, deux tâches ont été définies : l'association sémantique et la similarité sémantique. L'*association sémantique* cherche à évaluer si les modèles de plongements de mots sont capables de déterminer que des mots ont des liens sémantiques entre eux. Par exemple, les mots « voiture » et « route » ont un lien sémantique, contrairement aux mots « voiture » et « plante ». La *similarité sémantique* est une métrique similaire mais qui se restreint aux mots pouvant s'utiliser de la même manière et ayant un sens proche. Par exemple, les mots « voiture » et « bus » sont similaires. Deux mots sont totalement similaires sémantiquement dans le cas de synonymes.

Afin de pouvoir utiliser ces deux évaluations, de nombreux jeux de données ont été créés. Ceux-ci sont à peu près tous construits de la même manière : en demandant à des personnes de donner une note à un couple de mots afin d'indiquer si deux mots sont associés et/ou similaires. Les notes sont ensuite comparées aux distances cosinus entre les couples de mots en utilisant la corrélation de Spearman.

Les premiers jeux de données ainsi construits sont RG [RG65] et MC [MC91]. Même si ceux-ci ont permis de mettre en avant ces propriétés d'association entre mots, ils ne sont plus utilisés car les corrélations obtenues ne sont pas significatives à cause de la taille des jeux de données [Far+16]. De nombreux autres jeux de données ont par la suite été créés afin d'avoir davantage de paires de mots : WS-353 [Fin+01 ; Agi+09], MTurk-287 [Rad+11], MTurk-771 [Hal+12], MEN [Bru+12] ou encore RW [LSM13]. Ces jeux de données se concentrent principalement sur les noms, par conséquent d'autres jeux de données ont été construits afin de s'intéresser spécifiquement aux verbes : YP-130 [YP06] et Verb [BRK14].

Une limite importante des différents jeux de données présentés jusqu'à présent est qu'ils ne font pas de distinction entre association et similarité sémantique. SimLex-999 [HRK16] et SimVerb-3500 [Ger+16] se penchent spécifiquement sur la question de la similarité sémantique.

2.5.1.2. Raisonnement par analogie

Une autre propriété intéressante est que les espaces de plongements sont construits de telle manière qu'il est généralement possible d'appliquer des opérations vectorielles dessus produisant des résultats intéressants. L'exemple le plus connu, introduit par Mikolov et al. [MYZ13], est que l'opération $\text{vecteur}(\text{Roi}) - \text{vecteur}(\text{Homme}) + \text{vecteur}(\text{Femme})$ permet d'obtenir un vecteur pour lequel le plongement le plus proche est celui de « Reine ». Ce phénomène permet de construire une tâche de *raisonnement par analogie* où l'objectif est à partir d'un premier couple de mots et d'un troisième mot de déterminer un quatrième mot permettant de constituer un couple avec le troisième mot.

Il a été montré par LEVY et GOLDBERG [LG14b] que cette tâche revient à maximiser une combinaison linéaire de trois similarités cosinus entre paires de mots. Par conséquent, l'évaluation en utilisant le raisonnement par analogie peut être vu comme une extension de la tâche de similarité sémantique.

2.5.1.3. Liens avec les processus cognitifs

Un inconvénient des approches basées sur de la similarité sémantique est qu'elles se fondent sur des jugements subjectifs. En effet, les jugements de similarités sont issus de processus conscients qui peuvent introduire de nombreux biais à cause de leur subjectivité. Il peut donc paraitre souhaitable d'évaluer de manière objective les plongements de mots directement sur des processus inconscients qui modélisent le langage dans le cerveau.

Dans la communauté des sciences cognitives, de nombreux travaux ont été réalisés pour expliquer la manière dont les associations entre mots sont formées en introduisant des plongements de mots dans leurs modèles. PEREIRA et al. [Per+16] observent la manière dont des plongements de mots mis à disposition par la communauté du TAL permettent de prédire des associations libres, c.-à-d. à partir d'un mot d'amorçage indiquer le premier mot qui vous vient à l'esprit. Cependant, encore une fois, ce phénomène repose sur des processus conscients. Dans les travaux de HOLLIS et WESTBURY [HW16], les composantes principales de plongements de mots sont comparées à des temps de réactions non amorcés (c.-à-d. sans présenter un mot avant de répondre à une question ou de réaliser une action) provenant du BRITISH LEXICON PROJECT et du ENGLISH LEXICON PROJECT.

Ces deux articles s'intéressent à expliquer des comportements cognitifs à l'aide de plongements de mots. Des travaux existent également dans le domaine de la visualisation cérébrale en essayant d'étudier la corrélation des plongements de mots avec des images issues de résonances magnétiques fonctionnelles. L'idée est alors d'espérer que les plongements de mots permettent d'expliquer certaines parties de l'activité cérébrale [Søg16].

2.5.1.4. Le temps de réaction amorcé au service de l'évaluation

Afin d'utiliser les processus cognitifs pour évaluer la qualité de plongements de mots, je présente dans le papier « Evaluation of Word Embeddings against Cognitive Processes » [ARF17] une approche se fondant sur l'amorçage sémantique afin de constituer un cadre d'évaluation semblable à ceux utilisés pour l'association et la similarité sémantique. Ce papier n'a pas un lien direct avec la thématique principale de ma thèse — d'où la faible couverture du problème dans ce document — mais il permet de bien mettre en lumière les difficultés qui sont rencontrées pour correctement construire des représentations pour modéliser le langage.

Mes travaux se fondent sur le *Semantic Priming Project* (SPP) [Hut+13] qui est un jeu de données comportant les temps de réponses de 768 participants dans deux tâches chronométrées différentes : dénomination et décision lexicale. La *tâche de dénomination* (NT) consiste à demander au participant de lire à voix haute un mot montré à l'écran. La *tâche de décision lexicale* (LDT) consiste à appuyer sur un des deux boutons mis à disposition afin d'indiquer si un mot affiché à l'écran existe ou non. Que ce soit pour l'une ou l'autre des tâches, avant l'affichage du mot — appelé mot cible — à l'écran, un mot d'amorce est au préalable montré. Le temps d'attente entre l'affichage du mot d'amorce et du mot cible peut être soit de 200 ms, soit de 1200 ms. Ces tâches permettent alors de mesurer le temps de réaction des participants pour donner leur réponse (lecture du mot cible ou appui sur un bouton).

À partir de ces expériences, le SPP a construit un jeu de données constitué de 6 637 paires de mots, chacune ayant des temps de réactions dans les 4 configurations possibles, c.-à-d. LDT ou NT avec 200 ou 1200 ms d'attente. Chaque temps de réaction est une moyenne des performances obtenues par 30 participants différents.

Les temps de réaction observés dans ces expériences peuvent être expliqués par un très grand nombre de phénomènes linguistiques tels que l'association et la similarité sémantique, des traits syntaxiques et la morphologie. Contrairement à HILL et al. [HCK16] qui se concentrent sur un seul phénomène, je suppose que des représentations de mots doivent capturer l'ensemble des facteurs observés dans le comportement humain. De ce fait, j'ai évalué les plongements de mots en calculant la corrélation entre la similarité cosinus entre paires de mots et les temps de réaction.

Bien entendu, d'autres facteurs non linguistiques peuvent entrer en jeu pour expliquer les temps de réaction et il n'est pas attendu que les plongements de mots modélisent parfaitement cela.

Jeu de données	# paires	LDT-200	LDT-1200	NT-200	NT-1200
MTurk-771	26	-0.08	0.23	-0.23	-0.28
MEN-TR-3k	71	0.21	-0.20	0.04	-0.02
SimLex-999	101	-0.03	0.06	0.06	0.02

TABLE 2.1. – Corrélations de Spearman entre les temps de réaction provenant du jeux de données `SPP` et les jugements d'association/similarité provenant d'autres jeux de données.

Le cadre d'évaluation que je propose est disponible en ligne [1] et permet d'évaluer la qualité de modèles de plongements de mots quelconques. Afin de déterminer si les temps de réactions entre paires de mots reviennent au même (ou pas)

1. https://github.com/JomnTAL/spp-wordsim

que les jugements subjectifs usuellement utilisés, dans la table 2.1 je mesure la corrélation de Spearman qu'il peut y avoir entre les temps de réaction provenant de SPP et les notes subjectives provenant de trois jeux de données utilisés pour mesurer l'association/similarité sémantique. Étant donné que les paires de mots ne sont pas les mêmes dans les différents jeux de données, je n'utilise que les paires communes entre SPP et les autres jeux de données (le nombre de paires en commun est indiqué dans la colonne « # paires »).

On peut constater que les corrélations avec les jeux de données usuellement utilisés sont très faibles. En effet, quel que soit le jeu de données, les corrélations sont proches de 0. Le faible nombre de paires en commun ne permet pas d'indiquer quel jeu de données ressemble le plus à SPP mais dans tous les cas, il semblerait que les évaluations se basant uniquement sur des jugements conscients ne permettent pas d'évaluer tous les aspects du langage, et en particulier modélisent très peu les processus cognitifs inconscients. Le jeu de données SPP peut donc être utilisé pour évaluer la qualité des plongements de mots afin de déterminer si ceux-ci sont capables de capturer des phénomènes inconscients se produisant chez l'humain.

Plongements de mots	LDT-200	LDT-1200	NT-200	NT-1200
GloVe	**0,250**	**0,189**	**0,154**	**0,126**
word2cec	0,154*	0,112*	0,057*	0,089*
Multilingual	0,139*	0,112*	0,064*	0,076*
Dependency	0,055*	0,038*	−0,016*	0,047*
FastText	0,145*	0,106*	0,037*	0,083*

Table 2.2. – Évaluation sur le jeu de données SPP de plongements de mots en libre accès en utilisant la corrélation de Spearman [2]. La significativité des corrélations comparées aux meilleurs résultats obtenus (en gras) à l'aide du test Steiger est indiquée par le symbole '*' ($pval < 0{,}01$).

Étant donné les résultats précédents, il est intéressant de voir comment se comportent des modèles de plongements existants sur les données SPP. En effet, ces modèles n'ayant jamais été évalués sur des jugements objectifs, il est important de déterminer si ces plongements modélisent les processus cognitifs étudiés dans le cadre du SPP. La table 2.2 présente les corrélations de Spearman sur le jeu de données SPP de modèles de plongements de mots disponibles en libre accès. Pour rappel, les corrélations sont calculées en comparant les temps de réactions et les distances cosinus entre paires de mots. Les modèles évalués sont GloVe [PSM14], word2vec [Mik+13], Multilingual [FD14] qui sont des plongements multilingues fondés sur LSA, Dependency [LG14a] qui sont des plongements word2vec utilisant l'arbre syntaxique pour définir le contexte d'un mot et FastText [Boj+17].

On peut constater que les corrélations sont plutôt faibles — ne dépassant

pas $0{,}25$ — quelle que soit la tâche considérée. On observe tout de même de meilleures corrélations sur la tâche LDT-200 qui peuvent s'expliquer par le fait que c'est la tâche influencée par le moins de facteurs autres que celui de l'association sémantique entre mots (NT requiert de savoir prononcer le mot alors que LDT ne demande que l'appui sur un bouton). De manière générale, les modèles de plongements existants semblent mieux modéliser le mécanisme d'amorçage automatique permettant d'associer un mot à un autre de manière instantanée. On constate également que les vecteurs GLOVE sont significativement meilleurs que les autres modèles de plongements pour modéliser ces tâches. Il est toutefois difficile de donner des raisons à cela.

Les problèmes mis en avant par les évaluations se fondant sur des travaux des sciences cognitives montrent bien que même sur des tâches semblant bien établies — et qui sont de nos jours utilisée pour toutes les applications du TAL — il n'est pas évident de construire des cadres d'évaluations permettant de prendre en compte toutes les propriétés linguistiques qui devraient être modélisées. Il est donc important de garder à l'esprit que tout cadre d'évaluation, même très complet, ne permet d'évaluer que des propriétés bien spécifiques, généralement en lien avec un objectif final précis (en lien avec une tâche en aval). Ceci sera donc également valable sur mes travaux sur les dialogues et la construction de représentation du discours conversationnel. En particulier, les cadres d'évaluation et les modèles proposés seront mis en place dans le but de correspondre à des conversations de type « tchat » provenant d'assistances clientèles.

2.5.2. Évaluation des plongements de phrases

Comme pour les plongements de mots, la question de l'évaluation se pose pour les plongements de phrases. En revanche, contrairement aux premiers, les travaux sur ces derniers sont beaucoup plus récents et il n'est pas encore évident de déterminer les différentes propriétés que doivent modéliser les plongements de phrases. Comme précédemment, deux types d'évaluations existent : intrinsèques et extrinsèques.

Plusieurs travaux [Kir+15 ; HCK16] essaient d'évaluer les plongements obtenus en imitant ce qui se fait avec les plongements de mots, c.-à-d. en faisant de la similarité sémantique au niveau des phrases. Pour cela, il existe différents jeux de données tels que `SICK` et `STS 2014` qui attribuent des scores de similarité à des paires de phrases. Une limite de cette approche est qu'elle se base sur des jugements très subjectifs et qui le sont probablement davantage sur des phrases entières. Ces différents travaux s'évaluent également sur des tâches en aval dans le but de déterminer si les plongements permettent de modéliser l'opinion et les sentiments dans une phrase.

CONNEAU et al. [Con+18] proposent de s'intéresser à des phénomènes linguistiques en observant à quel point les plongements sont capables de modéliser des informations de surfaces (longueur de la phrase ou retrouver la présence

de mots précis dans la phrase), syntaxiques (vérifier que l'ordre des mots est correctement modélisé mais également modéliser les syntagmes de la phrase) et sémantiques (étudier comment le plongement évolue lorsque des mots sont remplacés).

Une limite majeure de ces approches est qu'elles considèrent les phrases indépendamment de leur contexte de production. Or, il est évident qu'une même phrase peut avoir des sens et des fonctions très différents en fonction du contexte dans lequel celle-ci est utilisée. Par exemple, la phrase « Je vais éclairer la lanterne de Bob » n'aura pas la même signification si la phrase précédente était « Il fait sombre ici ! » plutôt que « Il a rien compris... ».

Le cadre d'évaluation SENTEVAL [CK18] permet, en plus des approches précédentes, de vérifier si les plongements permettent de correctement résoudre des tâches d'inférences, de détection de paraphrases et d'identification de légendes d'images.

La tâche de détection de paraphrases permet de déterminer si les plongements capturent correctement le sens des phrases. En effet, étant donné deux phrases, le but est de déterminer si elles sont paraphrases l'une de l'autre. Cette tâche nécessite de fait une compréhension du sens des phrases. Néanmoins, la tâche ne répond pas vraiment à la problématique de la prise en compte du contexte de production, qui par ailleurs pourrait influencer le sens des phrases.

Une tâche d'inférence consiste à déterminer si une hypothèse est vraie, fausse ou neutre en fonction d'un postulat donné. Dans le contexte de l'évaluation des plongements de phrases, l'objectif est de déterminer si ceux-ci modélisent correctement cette information très fortement liée à la compréhension du langage. Les jeux de données SNLI [Bow+15] et MULTINLI [WNB18] ont été spécifiquement construits pour évaluer cette tâche. Cette tâche est une première approche permettant de vérifier que les plongements de phrases sont capables de s'adapter à un contexte de production. En effet, en fonction du postulat de départ, la classe de l'hypothèse devra être adaptée.

Une autre tâche intéressante est celle de l'identification de couples légende-image [Lin+14]. Le but est soit de classer une collection d'images en fonction de leur pertinence par rapport à une légende donnée, soit de classer des légendes données en fonction de leurs pertinences étant donné une image. Les modèles de plongements doivent donc être capables de correctement correspondre aux contextes de production donnés par les images. On peut cependant noter que cette tâche s'inscrit dans le cadre de tâches assez spécifiques et difficilement applicables à tous les domaines. En outre, généralement les phrases rencontrées ont une variabilité plutôt limitée (description d'images).

Ces deux dernières tâches d'évaluation sont intéressantes car elles imposent aux modèles de plongements de mots de prendre en compte le contexte de production. Ces approches mettent également en avant le fait qu'il ne parait pas évident de déterminer une seule méthode d'évaluation permettant de valider toutes les propriétés souhaitées. Sur le dialogue, la prise en compte du contexte

de production est d'autant plus importante que presque tous les tours de parole sont produits en réaction à d'autres tours de parole. Un moyen de modéliser le contexte de production d'un tour de parole est alors de se fonder sur des analyses du discours conversationnel, qui permettent de mettre en évidence les interactions entre locuteurs. Toutefois, il existe peu de travaux cherchant à évaluer les plongements de phrases dans le contexte dialogique. Un des objectifs de ma thèse va donc être de proposer des modèles de plongements et des cadres d'évaluation prenant en compte le discours conversationnel.

2.6. Discussion

On s'est intéressé dans ce chapitre à étudier comment sont construites des représentations d'unités linguistiques utilisables par des algorithmes d'apprentissage et qui permettent de modéliser des propriétés intéressantes.

Une question récurrente qui se pose lors de l'élaboration de représentations distributionnelles est de savoir comment faire en sorte que celles-ci prennent en compte le contexte de production dans lequel ces unités linguistiques s'inscrivent. Sur les mots, cette évolution est particulièrement visible. On a d'abord cherché à obtenir des représentations similaires pour des mots ayant certains sens proches. Puis, des plongements contextuels sont construits dans le but de pouvoir différencier les différents sens existants d'un même mot.

On peut cependant constater que ces améliorations-là s'inscrivent toujours en réponse à des besoins grandissants pour des tâches de plus en plus complexes exploitant ces représentations. Il est possible de valider des modèles de plongements de mots en les utilisant sur des tâches en aval. Toutefois, il est important de déterminer les propriétés linguistiques qui sont (ou pas) modélisées par ces modèles afin de pouvoir apporter des solutions aux limites rencontrées. Ceci est fait à l'aide de cadres d'évaluation cherchant à évaluer la qualité des plongements sur des propriétés bien précises. Il est néanmoins très difficile de faire des évaluations universelles capturant toutes les propriétés associées à une unité linguistique donnée, comme on a pu le voir avec les mots et les phrases.

Dans ma thèse, je vais m'intéresser aux dialogues entre humains. Dans le domaine des constructions de représentations de mots ou de phrases, les textes considérés sont généralement issus d'articles ou de livres qui ont des propriétés assez différentes des dialogues. Un des objectifs de ma thèse est de parvenir à construire des représentations des dialogues qui modélisent le discours conversationnel de ceux-ci. Deux types de représentations vont alors être étudiées sur des dialogues de type « tchat » : des représentations explicites du discours conversationnel telles que présentées dans le chapitre 1 et des représentations distributionnelles des phrases et des dialogues.

Deuxième partie

Représenter implicitement le discours à partir d'un corpus de tchats

Chapitre 3.

Corpus Datcha : description et annotations de départ

Sommaire

3.1 Introduction 80
3.2 DATCHA : un corpus de tchats d'assistance clientèle 81
3.2.1 Le corpus DATCHA 81
3.2.2 Descriptions des différents sous-corpus 83
3.3 Comparaison avec des conversations orales 85
3.3.1 La silhouette des dialogues 86
3.3.2 Les erreurs dans le langage 87
3.4 Annotation en actes de dialogue 88
3.4.1 Le jeu d'étiquettes utilisé 88
3.4.2 Le sous-ensemble du corpus annoté manuellement 92
3.5 Étiquetage automatique en actes de dialogue 93
3.6 Discussion 94

3.1. Introduction

Les deux chapitres précédents ont permis de présenter les dialogues, les différentes théories d'analyses du discours et comment créer des représentations distributionnelles d'objets linguistiques. Bien que la question principale de la thèse est de savoir comment obtenir des représentations du discours conversationnel à l'aide de peu de données annotées discursivement, il est indispensable de disposer d'un corpus de dialogues sur lequel s'appuyer.

Dans mes travaux, je m'intéresse spécifiquement aux dialogues de type « tchat ». Ceux-ci ont la particularité d'être des dialogues écrits, spontanés et synchrones, à la différence de conversations se trouvant dans les corpus UBUNTU [Low+15] contenant des discussions provenant d'un forum (conversations ni spontanées, ni synchrones) ou SWITCHBOARD [GHM92] contenant des conversations téléphoniques (et donc orales). Le corpus STAC [Ash+16] est un exemple de corpus de

tchats. Néanmoins, ce corpus est relativement petit (environ 1 000 dialogues) et n'est donc pas adapté à l'apprentissage de représentations distributionnelles. Par ailleurs, dans le cadre de l'ANR DATCHA dans lequel s'inscrit ma thèse, les représentations du discours doivent être utiles pour des applications en français et dans le cadre de conversations entre un téléconseiller et un client dans le domaine de l'assistance clientèle (le corpus STAC est en anglais et contient des dialogues multi-parties concernant un jeu de société).

Dans ce chapitre, je vais donc présenter le corpus DATCHA produit dans le cadre de l'ANR du même nom et qui respecte toutes les conditions énoncées précédemment [1]. Dans la section 3.2, je vais présenter le corpus, les conversations qui s'y trouvent (des « tchats ») et les annotations de base qui ont été réalisées. Les « tchats » étant des dialogues à la frontière entre le langage oral et écrit, je présente dans la section 3.3 une comparaison entre les dialogues du corpus DATCHA et ceux provenant de centres d'appels téléphoniques. Dans la section 3.4 je présente l'annotation en actes de dialogue qui a été produite sur un sous-ensemble du corpus qui a permis le développement d'un étiqueteur en actes de dialogue, présenté dans la section 3.5.

3.2. Datcha : un corpus de tchats d'assistance clientèle

Dans le but de modéliser les interactions ayant lieu entre interlocuteurs dans des conversations de type « tchats », il est utile d'avoir à disposition un corpus de conversations. Ma thèse s'inscrit dans le cadre du projet ANR DATCHA, qui me donne accès à des conversations d'assistance clientèle issus du service client de l'entreprise de télécommunication Orange. Ce corpus, que nous nommerons dans la suite de la thèse le corpus DATCHA, est constitué de conversations médiées par ordinateur de type « tchats » ayant lieu entre deux individus.

3.2.1. Le corpus Datcha

Les conversations présentes dans le corpus DATCHA sont des « tchats ». De ce fait, les communications ont lieu de manière synchrone avec accès à un historique des tours de parole précédents.

Le corpus DATCHA est un corpus de conversations d'assistance clientèle, les sujets de conversation principaux tournent autour de la résolution de problèmes rencontrés par les clients d'Orange ou de questions ayant un lien avec leurs produits. Ce sont des dialogues où les participants ont des rôles très précis et qui ne changent pas durant la conversation, ce qui a une influence sur la production

1. Les données étant confidentielles et appartenant à Orange, le corpus ne peut malheureusement pas être diffusé à la communauté.

des dialogues. En effet, les conversations sont entre deux individus qui ont des rôles asymétriques où un participant a la fonction d'expert, ici un téléconseiller, et le deuxième participant est un simple usager, ici un client. Cette relation asymétrique a pour conséquence que le client est généralement l'interlocuteur introduisant la problématique de la conversation, alors que le téléconseiller est celui menant la suite de la conversation en indiquant des procédures à suivre et en posant des questions, par exemple. De manière générale, les conversations se terminent pour les raisons suivantes :

— le téléconseiller a apporté une solution (probante ou non) au problème posé par le client ;

— le client met un terme à la conversation avant l'apport d'une solution (par exemple, si le client est insatisfait de la tournure de l'échange et met un terme à celui-ci) ;

— un problème extérieur à la conversation (par exemple, une perte de réseau) interrompt celle-ci.

Il peut également arriver que le téléconseiller n'étant pas qualifié pour résoudre le problème transfère le client vers un autre téléconseiller. Au moment du transfert, du point de vue du premier téléconseiller, la conversation est alors terminée alors qu'elle démarre pour le second. En revanche, du point de vue du client, la conversation est toujours en cours : l'historique de la conversation est toujours présent et la suite de la conversation a lieu dans la même période de temps sur le même sujet.

Les problèmes à résoudre peuvent être classés en deux grandes catégories : les problèmes techniques et les problèmes commerciaux. Les problèmes techniques sont des problèmes liés au matériel d'Orange, utilisé par le client, ou à la connexion Internet. Lors de ces conversations, le téléconseiller peut être amené à demander au client d'effectuer des actions externes à la conversation tel que la manipulation d'un appareil. Les problèmes commerciaux sont eux liés à l'offre commerciale d'Orange pouvant aller de la simple demande de renseignements sur les tarifs proposés à la résolution d'un conflit entre le client et Orange du fait de désaccords sur les services rendus. La variété des problèmes traités se traduit par la variété des dialogues générés — que ce soit au niveau du contenu ou des interactions entre locuteurs.

Une autre source de variabilité provient de la manière dont est produit le langage dans les dialogues par les participants, en particulier les clients. En effet, différents registres de langue peuvent être utilisés par les clients et ceux-ci peuvent produire un français incorrect. Ce dernier aspect a pour conséquence que les conversations peuvent être *bruitées*. Ici, le terme « bruité » est utilisé afin de caractériser plusieurs phénomènes :

— différents types de fautes d'orthographe : suppression ou ajout de lettres, absence de diacritique, utilisation d'un homophone, écriture dans un style « SMS » ;

— différentes erreurs grammaticales ;
— présence de différents symboles ou entités produits volontairement ou non tels que des URL ou des émojis et émoticônes.

Un autre phénomène qui est régulièrement rencontré dans les conversations est celui des enchevêtrements de sous-dialogues. En effet, il arrive très régulièrement que les deux participants discutent de deux sujets différents simultanément et on se retrouve alors avec des tours de parole portant sur des sujets différents qui s'enchevêtrent. La figure 3.1 illustre ce phénomène avec un premier sous-dialogue (en bleu) où le téléconseiller demande si le client est toujours présent et un deuxième sous-dialogue (en rouge) portant sur des détails afin de résoudre un problème évoqué en amont de cet extrait. La présence de ces enchevêtrements permet d'illustrer un des aspects des dialogues pour lesquels l'analyse du discours conversationnel est important afin d'aider à comprendre les conversations.

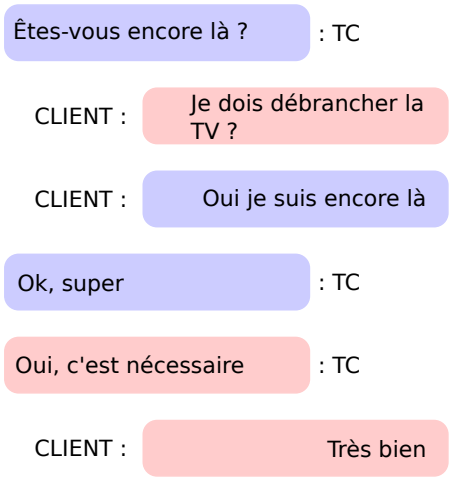


FIGURE 3.1. – Illustration du phénomène d'enchevêtrement de tours de parole

À titre d'illustration, un exemple de dialogue provenant du corpus DATCHA a été reproduit en annexe A. Les différentes annotations présentes dans cet exemple seront décrites dans les chapitres et sections à venir.

3.2.2. Descriptions des différents sous-corpus

Les contributions de la thèse utilisent en réalité deux sous-ensembles du corpus DATCHA. Ces deux sous-ensembles contiennent des conversations provenant de périodes différentes et qui ont été collectées différemment.

Le premier corpus, que nous nommerons DATCHAATH, est un ensemble de conversations provenant uniquement du service d'assistance technique traitant de problèmes liés à des offres dites « fixes [2] ». Ce corpus est constitué de 129 757

2. Ligne de téléphone et/ou d'internet fixe (en opposition à des offres pour des appareils mobiles)

conversations collectées sur plusieurs mois. Un très petit sous-corpus, nommé DATCHA91, de 91 conversations issues de DATCHAATH est également construit sur lesquelles diverses annotations manuelles ont été produites.

On y trouve en particulier des annotations en parties du discours et des analyses syntaxiques en dépendance réalisées dans le cadre des travaux de NASR et al. [Nas+16].

En complément de ces annotations, une correction manuelle de DATCHA91 a été réalisée. Lors de la correction, l'annotateur a eu pour consigne de corriger les mots mal construits sans pour autant modifier le contenu du message, c.-à-d. qu'il n'était pas autorisé à ajouter ou supprimer un mot même si la phrase ne semble avoir aucun sens. Je reviendrai plus en détail sur cette correction manuelle dans le chapitre 4.

Le second corpus, DATCHAFÉVRIER, est un ensemble de 221 228 conversations provenant de tous les services clients d'Orange, collectées durant un mois entier. La répartition entre assistance technique et commerciale est plutôt équilibrée. La distribution exacte est donnée dans la figure 3.2 où les catégories ACC, ACH et ACM correspondent à des canaux d'assistance commerciale alors que ATH et ATM correspondent à des canaux d'assistance technique. Pour les deux types d'assistance clientèle, les catégories se terminant par « H » correspondent à des canaux traitant des problèmes liés à des offres « fixes » alors que celles se terminant par « M » correspondent à des canaux traitant des problèmes liés à des offres « mobiles ». Cette distinction entre « mobile » et « fixe » est particulièrement importante pour les conversations issues de l'assistance technique car les problèmes rencontrés vont être résolus de manières différentes, et en particulier l'assistance technique sur le « fixe » va régulièrement demander des actions physiques à réaliser sur les routeurs, modems et décodeurs. En plus d'avoir une variété plus importante de sujets abordés dans les conversations que dans DATCHAATH, le corpus DATCHAFÉVRIER contient également des métadonnées associées à celles-ci comme le chemin suivi par le client avant d'arriver à la conversation ou des questionnaires de satisfaction. Nous reviendrons sur ces derniers dans le chapitre 5.

Les deux corpus, étant issus de périodes différentes, ils ne contiennent pas de conversations en commun. Les deux corpus ont également été prétraités différemment. Les conversations du corpus DATCHAFÉVRIER peuvent contenir des balises HTML, en particulier dans les tours de parole du téléconseiller. Ces balises sont généralement produites lorsque le téléconseiller utilise son interface pour inclure des phrases, ou bout de phrases, pré-écrites. Elles peuvent également être présentes lors de l'introduction de liens ou d'images. Dans le corpus DATCHAATH, les conversations ont été nettoyées par Orange et donc ces informations ne sont plus disponibles. Certains prétraitements ont été réalisés dans les deux corpus, en particulier l'anonymisation qui transforme les noms, adresses postales et électroniques et tout autre information personnelle des différents participants en nouvelles unités lexicales génériques. Par exemple, les noms des clients deviennent « _CLIENT_ » alors que ceux des téléconseillers deviennent « _TC_ ».

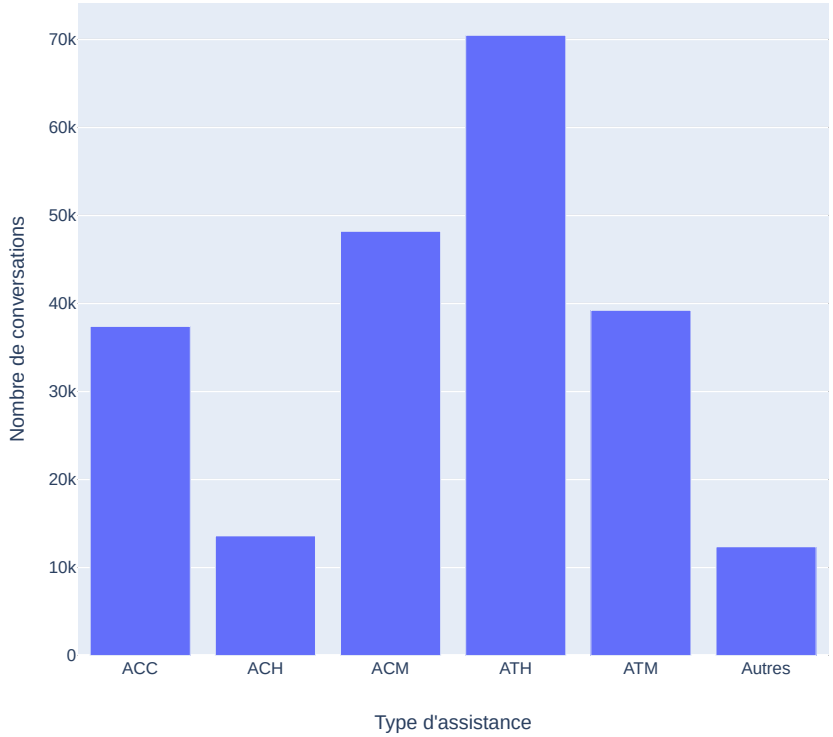


FIGURE 3.2. – Distribution des différents types de canaux d'assistance clientèle dans DATCHAFÉVRIER

La figure 3.3 récapitule les inclusions entre les différentes variantes du corpus DATCHA qui sont utilisées dans la thèse. DATCHAACT est décrit dans la section 3.4 et DATCHASAT dans le chapitre 5.

3.3. Comparaison avec des conversations orales

Les tchats produits dans le contexte d'assistance clientèle remplacent des conversations téléphoniques traditionnelles. Par ailleurs, les tchats sont probablement la forme de texte écrit reproduisant le plus de phénomènes normalement attribués à l'oral. Il serait donc intéressant de quantifier dans quelle mesure est-ce que cette similarité existe entre les conversations écrites et les conversations orales. Les travaux sur des conversations orales étant plus fréquents, quantifier cela permettrait de déterminer s'il est pertinent de s'inspirer de travaux se basant sur l'oral pour mes travaux sur l'écrit.

L'article « Web Chat Conversations from Contact Centers » de DAMNATI et al. [DGC16] propose une description intéressante du corpus DATCHA afin de pouvoir

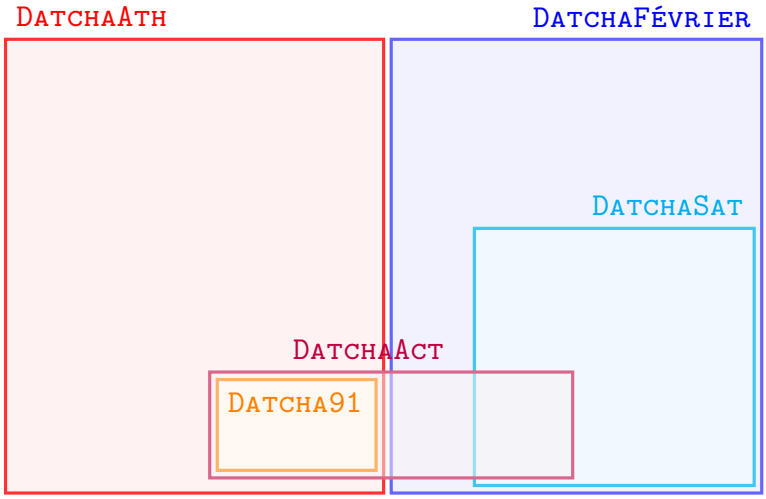


FIGURE 3.3. – Récapitulatif des différentes variantes du corpus DATCHA. Ce schéma illustre également des variantes décrites dans les chapitres suivants.

le comparer à d'autres corpus existants et de positionner les tchats vis-à-vis des conversations orales. Cette étude est importante car les « tchats » sont un format particulier ayant des ressemblances à la fois avec des textes oraux et écrits.

3.3.1. La silhouette des dialogues

La première analyse concerne la manière dont les interlocuteurs construisent temporellement les conversations en comparant les dialogues issus de DATCHA avec d'autres provenant d'un centre d'appel téléphonique traitant les mêmes thématiques. Le but est alors de voir si les « silhouettes » des dialogues oraux et écrits sont semblables. La *silhouette* d'un dialogue correspond à sa forme générale (nombre de tours de parole, durée des conversations, nombre de participants, etc.) sans tenir compte du contenu.

Dans l'étude de DAMNATI et al., ils considèrent qu'un message [3] à l'écrit correspond à un groupe de souffle à l'oral. Les tours de parole sont ici [4] plusieurs messages consécutifs du même scripteur dans le cas de l'écrit et plusieurs groupes de souffle consécutifs dans le cas de l'oral.

Les auteurs proposent alors de réaliser une comparaison des tours de parole en fonction de leurs longueurs et durées. La durée d'un message écrit est estimée à l'aide des horodatages. La table 3.1 permet d'observer les différences entre les deux types de conversations.

Il est intéressant de constater que les conversations écrites durent presque 2 fois plus longtemps que les conversations orales. En revanche, le nombre de tours

3. Un message est délimité par les retours à la ligne des scripteurs.
4. Cette définition est utilisée pour pouvoir réaliser des comparaisons avec l'oral.

	Écrit (230 dialogues)			Oral (56 dialogues)		
	Total	Client	Agent	Total	Client	Agent
Durée moyenne (sec.)	1186	549	636	595	162	222
# tours par dialogue	21,2	10,3	10,9	83,3	41,5	41,8
# messages par tour	1,41	1,27	1,54	1,47	1,33	1,62
# mots par message	11,2	8,6	13,2	11,1	10,0	12,1

TABLE 3.1. – Analyse de la forme des dialogues écrits et oraux

de parole par dialogue est 4 fois plus important à l'oral. Ces deux comportements sont probablement dus au medium de communication utilisé. En effet, l'écrit ne requiert pas d'être aussi immédiat et attentif que l'oral, les tours de parole précédents restant affichés à l'écran. Au contraire, à l'oral il est nécessaire d'être plus réactif afin d'écourter le temps d'attente, et il est nécessaire de mémoriser les tours précédents. Ce second aspect incite donc probablement les interlocuteurs à répéter des informations déjà communiquées et donc à allonger la conversation.

En dehors de ces aspects différenciant écrit et oral qui sont très liés au medium de communication utilisé, toutes les autres propriétés sont très proches. On peut en effet constater que le nombre de messages par tour ainsi que le nombre de mots par message sont similaires avec environ $1{,}5$ messages par tour et 11 mots par message. Que ce soit à l'écrit ou à l'oral, on constate également que le téléconseiller communique davantage que le client.

En se concentrant donc uniquement sur la « silhouette » des dialogues, les tchats et les conversations téléphoniques dans le contexte de l'assistance clientèle permettent la production de conversations similaires. Les différences majeures portant alors sur la vitesse de production des messages et la longueur des dialogues.

3.3.2. Les erreurs dans le langage

Une difficulté inhérente au traitement automatique du langage oral provient du fait qu'il est nécessaire d'analyser le signal audio afin de pouvoir déterminer quels mots ont été produits. Ceci est réalisé à l'aide de systèmes de reconnaissance automatique de la parole qui, bien qu'ils soient de plus en plus performants, produisent des erreurs de transcription.

Un avantage indéniable des conversations écrites par rapport aux conversations orales pour réaliser des traitements automatiques est qu'il n'y a pas d'étape de reconnaissance de la parole à réaliser, supprimant ainsi cette source de bruit. Cependant, les conversations écrites introduisent une nouvelle source d'erreurs qui n'existe pas à l'oral : les compétences à l'écrit des interlocuteurs, en particulier les compétences liées à l'orthographe et à la syntaxe.

DAMNATI et al. ont donc réalisé une analyse des erreurs orthographiques afin de pouvoir calculer un taux d'erreurs mots (*Word Error Rate* en anglais) (TEM), métrique usuellement utilisée pour évaluer des systèmes de reconnaissance de la parole. Ils ont pu constater que sur l'ensemble des conversations, le TEM est de seulement $4{,}3\%$, score qui est plutôt bas et qui est comparable avec ce que l'on peut trouver à l'oral sur le corpus SWITCHBOARD [GHM92] par exemple où les systèmes à l'état de l'art obtiennent des TEM autour de 5%. Il existe cependant une asymétrie importante entre le client et le téléconseiller. En effet, le TEM est de 10% pour le client et $1{,}6\%$ pour le téléconseiller. Cette grande différence peut s'expliquer par le fait que le téléconseiller écrit dans un contexte professionnel et a accès à certaines phrases pré-rédigées.

Comme pour l'oral, ces erreurs pourront donc potentiellement avoir des répercussions sur des tâches s'appuyant sur les tchats. Une analyse plus approfondie de l'influence que peuvent avoir ces erreurs lors de traitements automatiques reposant sur ces données est réalisée dans le chapitre 4.

3.4. Annotation en actes de dialogue

Un des objectifs du projet Datcha est de réaliser des analyses intra et inter-conversations en se fondant, entre autres, sur les structures discursives. Dans ce contexte, une annotation manuelle du discours est importante afin de pouvoir explorer différentes approches se fondant sur la structure discursive.

Une annotation manuelle en actes de dialogue a ainsi été réalisée sur un sous-ensemble du corpus DATCHA. Le choix de se limiter à une annotation en actes de dialogue — plutôt que de directement réaliser une annotation des relations discursives — est motivé par le fait que ceux-ci correspondent à un premier niveau d'analyse du discours pouvant servir comme base pour réaliser des analyses plus complexes. De plus, il est relativement simple d'annoter des énoncés avec des actes de dialogue étant donné que le contexte nécessaire pour déterminer la fonction d'un tour se cantonne généralement au tour courant et aux tours directement adjacents. Dans la suite de cette section, je décrirai donc le schéma d'annotation utilisé (3.4.1), ainsi que le sous-corpus ainsi annoté (3.4.2).

3.4.1. Le jeu d'étiquettes utilisé

Le corpus DATCHA a la particularité d'être un corpus de conversations écrites, qui tournent autour de un ou plusieurs problèmes à résoudre mais où les deux scripteurs, en particulier le client, ont une grande liberté sur les types de tours de dialogues pouvant être produits.

Afin de se concentrer sur les intentions générales portées par les énonciations, nous définissons un jeu d'étiquettes, inspiré du schéma DAMSL, construit afin de

Acte	Signification	Brève description
OPE	Ouverture	Tour d'ouverture du dialogue
PRO	Description du problème	Description du problème à résoudre
INQ	Question informative	Le scripteur demande des informations
CLQ	Question de clarification	Le scripteur demande une clarification
STA	Affirmation	Introduction de nouvelles informations
TMP	Temporisation	Introduction d'une pause dans le dialogue
PPR	Proposition de plan	Proposition d'un plan de résolution du problème à résoudre
ACK	Acquiescement	Acquiescement des propos de l'interlocuteur
CLO	Fermeture	Tour de fermeture du dialogue
OTH	Autre	Tour ne correspondant pas aux autres étiquettes

TABLE 3.2. – Jeu d'étiquettes en actes de dialogue utilisé pour DATCHA

spécifiquement prendre en compte les phénomènes majoritairement rencontrés dans les conversations de DATCHA.

Ce jeu d'étiquettes doit donc permettre de bien identifier les interactions liées au protocole imposé par le téléconseiller mais surtout sur celles liées à la résolution des problèmes du client. Le jeu d'étiquettes est brièvement décrit dans la table 3.2. Dans les paragraphes suivants, les différents actes de dialogue sont décrits de manière détaillée. La conversation en annexe A est annotée en actes de dialogue et permet d'illustrer la plupart des actes de dialogue pouvant être rencontrés.

Opening L'acte de dialogue *Opening* est utilisé pour les tours de paroles d'ouverture du dialogue. Ces tours sont à tendance protocolaire et sont généralement au début du dialogue. Ils contiennent très fréquemment des formules de salutations.

Quelques exemples hors contexte :

— Bonjour, je suis TC1

— Bonjour TC1

— Bonjour M. CLIENT, je m'appelle TC1 et je vais traiter votre demande.

Closing L'acte de dialogue *Closing* est utilisé pour les tours de fermeture du dialogue. Ces tours de parole sont également plutôt protocolaires et se situent généralement en fin de dialogue. Ils contiennent fréquemment des formules permettant de prendre congé de l'autre interlocuteur.

Quelques exemples hors contexte :

— Au revoir

— Merci, au revoir.

— Bonne journée !

— Merci pour votre confiance. Passez une bonne journée.

Statement L'acte de dialogue *Statement* permet de décrire les tours apportant de nouvelles informations au dialogue. Ces tours sont généralement au mode affirmatif mais peuvent également être au mode impératif. Ils sont souvent utilisés pour réaliser des descriptions ou pour répondre à des questions.

Exemple d'un sous-dialogue où l'affirmation est une réponse à une question :

— **TC**: Pouvez vous me confirmer le nom du titulaire de la ligne fixe NUMTEL, afin que je sois sure d' avoir le bon dossier ? (INQ)

— **CL**: C'est M. CLIENT (STA)

Un autre exemple où l'affirmation permet de donner une description et une consigne :

— **TC**: Votre décodeur a un problème. Je vous invite à le redémarrer dans 30 minutes. (STA)

— **CL**: Ok (ACK)

Acknowledgement L'acte de dialogue *Acknowledgement*, aussi appelé *Backchannel*, permet de décrire les tours utilisés pour acquiescer, ou pour simplement indiquer à l'autre interlocuteur qu'on est toujours présent et qu'on suit la discussion. Ces tours sont généralement très courts et ne sont pas indispensables à la lecture du dialogue.

Quelques exemples hors contexte :

— oui oui

— ok merci

— bien sûr

— euh non, attendez !

Temporisation L'acte de dialogue *Temporisation* est utilisé pour décrire les tours mettant en pause la conversation. Ces tours sont généralement présents lorsque l'un des deux interlocuteurs prévient qu'il va devoir réaliser une action en dehors de la discussion et qu'il va mettre en attente son interlocuteur.

Quelques exemples hors contexte :

— Merci de patienter momentanément.

— Veuillez patienter pendant que j'accède à votre dossier.

Information Question L'acte de dialogue *Information Question* permet de décrire les tours de parole où le scripteur pose une question informative à l'autre scripteur. Ces questions sont utilisées afin d'acquérir de nouvelles informations plus ou moins nécessaires à la résolution du problème.

Quelques exemples hors contexte :

— Puis-je avoir votre numéro de téléphone ?

— Est-ce que votre décodeur est allumé ?

Clarification Question L'acte de dialogue *Clarification Question* permet de décrire les tours de parole où le scripteur pose une question de clarification à l'autre scripteur. Ces questions n'apportent pas de nouvelles informations et sont généralement utilisées lorsqu'un des scripteurs n'est pas certain d'avoir compris quelque chose.

Exemple de sous-dialogue :

— **TC**: Votre décodeur a un probleme. Je vous invite le redémarrer dans 30 minutes. (STA)

— **CL**: OK je redemarre le décodeur dans 30 mins ? (CLQ)

Problem Description L'acte de dialogue *Problem Description* est un acte spécifique au type de dialogues présents dans DATCHA. Cet acte est utilisé pour caractériser les tours de parole décrivant le problème à résoudre. Ces tours sont en général uniquement produits par le client. Il arrive cependant, très rarement, que l'agent produise un tour avec cet acte. Ce cas se produit lorsque le client a réalisé des actions proposées par Orange avant l'ouverture du tchat et donc l'agent connait déjà le problème du client au démarrage du dialogue.

Exemple de sous-dialogue :

— **TC**: Que puis-je pour vous ? (INQ)

— **CL**: Mon décodeur TV est tombé en panne hier. Plus aucun signe vie. Je souhaiterais connaitre la démarche à suivre pour le remplacer. (PRO)

Plan Proposal L'acte de dialogue *Plan Proposal* est en quelque sorte la réponse à l'acte *Problem Description*. Cet acte est utilisé pour décrire les tours qui apportent une réponse au problème du client. Par conséquent, cet acte ne peut être produit que par l'agent.

Exemple de sous-dialogue contenant deux tours PPR :

— **TC**: Merci d'avoir patienter. Veuillez noter votre numéro d'échange afin de récupérer un nouveau décodeur... (PPR)

— **TC**: Vous pouvez souscrire à l'option Enregistreur lors de l'échange, pour avoir le nouveau décodeur UHD 87 SAT. (STA)

— **CL**: Option incluse dans l' offre liveboxstar non ? (INQ)

— **TC**: Je vérifie l'offre. (TMP)

— **TC**: Exact, l'option est incluse. (STA)

— **TC**: Donc à l'agence vous demandez l'enregistreur et le nouveau décodeur UHD 87 SAT. (PPR)

3.4.2. Le sous-ensemble du corpus annoté manuellement

Le corpus DATCHA étant très volumineux, il n'est pas envisageable d'annoter manuellement l'ensemble de celui-ci. Par conséquent, uniquement un sous-ensemble de ces conversations est annoté, que nous nommerons DATCHAACT dans la suite de la thèse. Afin d'avoir un corpus avec le plus d'annotations différentes possibles, une première partie de DATCHAACT correspond aux conversations provenant de DATCHA91, qui est déjà annoté syntaxiquement. L'autre partie de DATCHAACT provient d'un sous-ensemble de conversations provenant de DATCHAFÉVRIER, permettant ainsi d'obtenir une variété dans les problèmes traités par les scripteurs dans les dialogues.

Un problème qui se pose fréquemment lorsqu'une annotation en actes de dialogue est produite est celle de la segmentation en énoncés. Dans notre cas, nous considérons que les segments à annoter sont définis directement par les scripteurs eux-mêmes, c.-à-d. qu'un segment est délimité par les retours à la ligne produits par les utilisateurs et aucune segmentation additionnelle n'est réalisée. Parfois, plusieurs actes de dialogue peuvent cependant être pertinents pour un même segment. Dans ces cas-là, la liste des actes de dialogue correspondants est annoté tout en indiquant l'acte de dialogue dominant du segment. Le choix de segmentation réalisé a l'inconvénient d'avoir un manque de précision, mais a plusieurs avantages :

1. L'annotation manuelle est plus facile et rapide car il n'y a pas à se poser la question de la segmentation.
2. On laisse les scripteurs définir les segments et donc cela donne davantage d'importance aux retours à la ligne qu'ils ont choisi de faire volontairement.

L'annotation a ainsi été réalisée sur $2\,988$ dialogues par un unique annotateur. Un second annotateur a été utilisé sur $3\,027$ tours de paroles permettant de calculer un Kappa de Cohen de $\kappa = 0{,}67$. Celui-ci est plutôt faible mais la moitié des divergences correspondent à des omissions dans le guide d'annotation utilisé qui était imprécis dans certaines situations fréquentes dans le corpus DATCHAACT. Des ambigüités peuvent cependant également être en cause par exemple entre certaines questions (INQ et CLQ) ou certaines affirmations de type STA qui peuvent parfois également être associés aux étiquettes PRO ou PPR.

Le corpus DATCHAACT est partitionné en trois pour obtenir un corpus d'apprentissage ($2\,390$ dialogues), de développement (299 dialogues) et d'évaluation (299 dialogues).

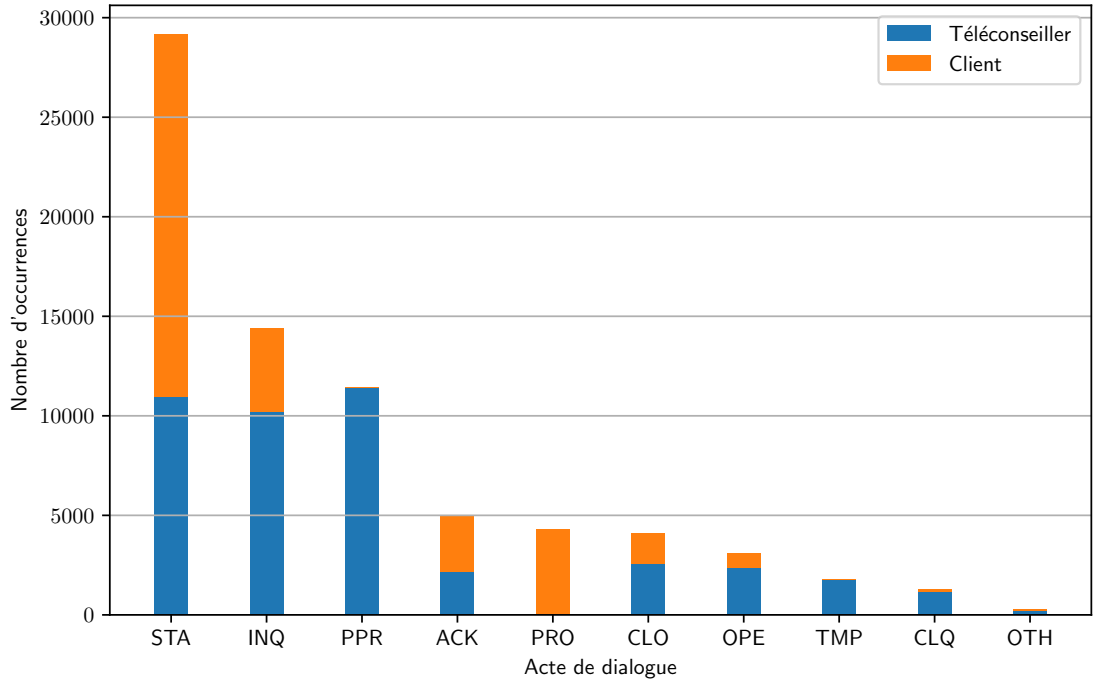


FIGURE 3.4. – Distribution des actes de dialogue dans le sous-ensemble d'apprentissage du corpus DATCHAACT

La figure 3.4 présente la distribution des différents actes de dialogue au sein du corpus d'apprentissage DATCHAACT ainsi annoté manuellement. On peut également y trouver la distribution des actes par type de scripteur. À partir de cette figure, on peut très clairement voir que les actes de dialogue les plus fréquents sont STA, INQ et PPR. Ceci n'est guère surprenant étant donné que ce sont les actes de dialogue directement liés au processus de résolution du problème du client par le téléconseiller.

En observant la distribution par type de scripteur, on peut constater que le client produit en majorité des tours de parole ayant pour acte de dialogue STA. Le client va donc généralement produire des affirmations, qui sont sans doute très souvent des réponses aux questions du téléconseiller. Au contraire, le téléconseiller produit des tours de parole beaucoup plus variés. Ceci semble donc nous confirmer que c'est le téléconseiller qui dirige la conversation et non le client qui va lui se contenter dans un premier temps d'énoncer son problème puis il va généralement répondre aux questions du téléconseiller.

3.5. Étiquetage automatique en actes de dialogue

L'annotation porte sur un sous-ensemble du corpus DATCHA qui est relativement petit comparativement au nombre de conversations disponibles dans le

corpus. De ce fait, il est intéressant de construire des étiqueteurs permettant d'annoter automatiquement l'ensemble du corpus DATCHA. Étant donné que la quantité de conversations annotées est assez importante, il est raisonnable de construire ces étiqueteurs à l'aide d'algorithmes d'apprentissage supervisés.

L'étiqueteur qui a été construit à partir de ces données annotées manuellement utilise des CRF. J'ai également construit un modèle hiérarchique à base de LSTM mais celui-ci étant plus complexe à utiliser par la suite pour propager l'annotation sur l'ensemble du corpus, et le modèle CRF obtenant des performances très similaires [PNA18], le choix a été fait d'utiliser ce dernier pour annoter automatiquement l'ensemble du corpus DATCHA.

Un modèle fondé sur un CRF nécessite que les caractéristiques en entrée soient sélectionnées (par un humain) avec pour objectif de retenir celles permettant de résoudre au mieux la tâche d'étiquetage. Les caractéristiques utilisées pour décrire un tour de parole dans le modèle CRF sont les suivantes :

— la longueur du tour ;

— la position relative du tour dans le dialogue ;

— le scripteur du tour ;

— les mots apparaissant dans le tour.

Le modèle CRF appris sur le corpus d'apprentissage de DATCHAACT obtient un score d'exactitude de 86% sur le corpus de test de DATCHAACT.

Le score obtenu correspond aux ordres de grandeur obtenus sur d'autres corpus à l'aide des modèles états de l'art décrit dans la section 1.3.3 du chapitre 1. Étant donné que le but de ma thèse est de construire des représentations du discours conversationnel dans son ensemble et au vu des résultats obtenus par le CRF, il ne parait pas indispensable de porter mon attention sur la tâche de prédiction d'actes de dialogue et je me tiens à ce qui a déjà été fait.

En utilisant ce modèle, le corpus DATCHAFÉVRIER a été automatiquement annoté en actes de dialogue, permettant ainsi une utilisation de ceux-ci dans le cadre de tâches nécessitant un ensemble très important de données mais où il n'est pas primordial que l'annotation en actes de dialogue soit parfaite.

3.6. Discussion

Dans ce chapitre, j'ai décrit le corpus sur lequel l'ensemble de mes travaux seront réalisés. Le corpus DATCHA permet d'étudier une catégorie de conversations qu'il est difficile d'obtenir car contenant des informations sensibles pour les entreprises. Ces tchats ont l'intérêt d'être produits dans le cadre bien défini de l'assistance clientèle. Cela permet de se concentrer sur un ensemble contrôlé d'interactions entre locuteurs qui ont toujours pour objectif de réussir à résoudre un problème.

Par ailleurs, ces dialogues ont la particularité d'être produits dans un contexte spontané étant donné que ces conversations ne sont pas produites dans le but de constituer un corpus — les interlocuteurs ayant pour seul objectif de résoudre un problème bien réel. Cependant, l'aspect spontané des conversations fait que celles-ci peuvent prendre des formes très variées, non contrôlées, ce qui a pour conséquence de complexifier l'analyse automatique.

Enfin, le medium de communication et le cadre dans lequel il est utilisé permet d'acquérir un nombre de conversations très volumineux directement utilisables pour faire du TAL. En complément des quelques annotations (morpho-syntaxiques, correction manuelle, actes de dialogue, questionnaire de satisfaction client) incluses dans le corpus, cette quantité importante de conversations offre la possibilité d'explorer un nombre important d'approches, en particulier neuronales, afin d'apprendre diverses représentations distributionnelles ou pour réaliser des analyses du discours.

Nous verrons dans les chapitres suivants différentes approches permettant d'exploiter les propriétés de ce corpus dans le but de modéliser le discours conversationnel. Dans la section 3.3.2, on a cependant pu constater que les dialogues disponibles sont bruités, en particulier les tours de parole du client. Ce bruit peut potentiellement avoir une influence sur des tâches de TAL. Dans le chapitre suivant, je vais donc mesurer l'influence que peut avoir le bruit — en particulier les erreurs orthographiques — sur une tâche reposant fortement sur la forme des mots : l'étiquetage en partie du discours.

Chapitre 4.

Influence du langage tchat sur des tâches de TAL

Sommaire

4.1 Introduction 96
4.2 Corrections manuelles du corpus 98
4.3 Correction automatique du corpus 99
4.3.1 Définir le lexique de mots corrects 101
4.3.2 Correction automatique basées sur des distances d'édition . 101
4.3.3 Utilisation de plongements de mots pour réordonner les corrections 102
4.4 Étiqueteur en parties du discours sur les données DATCHA 103
4.5 Évaluation de l'influence des erreurs 104
4.6 Discussion 109

4.1. Introduction

Généralement lorsqu'on travaille sur des textes écrits, ceux-ci ne sont pas des productions spontanées. En effet, lors de la rédaction de ces textes, les auteurs ont le temps de réfléchir à chaque phrase permettant ainsi de limiter le nombre de fautes orthographiques et grammaticales. Les textes spontanés sont généralement liés à l'oral : conversations téléphoniques, réunions ou discussions entre connaissances en face à face.

Une particularité des communications médiées par ordinateur de type « tchats » est que ce sont des textes écrits spontanés. Les participants s'attendent à ce que leurs interlocuteurs réalisent des échanges relativement rapides, ne laissant ainsi pas un temps important à la réflexion. De surcroit, dans les données DATCHA, les clients sont généralement dans un contexte non professionnel, peuvent avoir des compétences très variables en français et n'ont pas réellement de contrainte sur la forme que doivent avoir les discussions. Ces différents facteurs ont pour effet que beaucoup d'erreurs orthographiques et grammaticales — que je nommerai *bruits* — peuvent être présentes dans les conversations.

Les tâches de TAL reposent sur les mots. On peut donc se demander quelle est l'influence des erreurs au niveau des mots sur celles-ci. La réponse à ce problème est importante car des mots mal orthographiés peuvent provoquer des erreurs à divers niveaux d'analyses : morphologique, syntaxique ou encore discursif. Dans mon cas, j'aimerai être capable de dire explicitement si les fautes d'orthographe influencent la compréhension du dialogue. Ceci est difficile à directement évaluer car cela nécessiterait d'avoir des données annotées discursivement mais aussi corrigées manuellement. Or, à ce moment là des travaux, nous n'avions pas d'annotations discursives à disposition, rendant impossible une telle analyse.

Par conséquent, nous avons fait le choix de passer par une approximation avec une tâche morphosyntaxique bien établie : la prédiction des parties du discours. Un intérêt de cette tâche est que si on constate que le bruit au niveau des mots influence peu les prédictions, alors il est probable que ce soit également le cas sur des tâches d'analyse du discours. Au contraire, si une influence relativement importante est constatée, alors il serait judicieux d'appliquer des prétraitements en amont d'une analyse du discours afin de limiter l'influence du bruit initial.

La tâche de prédiction des parties du discours est bien maitrisée. Sur le français avec le corpus French TreeBank (FTB) [ACT03], les résultats obtenus sont proches des 98% d'exactitude sur l'ensemble des mots et proches des 92% sur les mots inconnus [DS09 ; Con+11]. Cette tâche est donc intéressante à étudier dans notre cas car le nombre de mots inconnus est potentiellement beaucoup plus important que d'ordinaire. En effet, chaque faute d'orthographe va en quelque sorte pouvoir créer un nouveau mot. Il est aussi possible qu'à cause d'une faute d'orthographe — par exemple une absence de diacritique — certains mots se retrouvent avec la même orthographe que d'autres mots existants ayant des parties de discours très différentes (par exemple, le verbe « a » et la préposition « à »). En outre, les étiqueteurs en parties du discours sont en général étudiés sur des corpus journalistiques — le FTB étant issu du journal « Le Monde » par exemple — et donc le simple fait de travailler sur des tchats — avec des constructions de phrases différentes — peut avoir une influence sur les performances.

Dans le but d'étudier l'influence de ces différents phénomènes, l'étude est conduite en plusieurs parties :

1. Les conversations du corpus Datcha91 sont corrigées manuellement (section 4.2).
2. À l'aide d'un correcteur orthographique, on réalise une correction automatique du corpus Datcha91 (section 4.3).
3. Des étiqueteurs en parties du discours sont entrainés sur les données Datcha, s'appuyant sur des plongements de mots appris dans des conditions différentes (section 4.4).
4. Les performances des étiqueteurs sont comparées en utilisant plusieurs types d'entrées : corrigée manuellement, corrigée automatiquement et brute (section 4.5).

Ce dernier point est le cœur du chapitre et permet d'évaluer dans quelle mesure est-ce que le format des tchats influencent les prédictions dans une tâche morphosyntaxique. Par extension, ces travaux permettent de déterminer s'il est nécessaire de s'adapter à ce type de textes lors de leurs utilisations dans diverses tâches du TAL, dont l'analyse du discours.

4.2. Corrections manuelles du corpus

Comme indiqué dans le chapitre précédent, le corpus DATCHA91 a été construit à partir de 91 conversations, en y incluant des annotations morphosyntaxiques. L'une d'entre elles est l'annotation en parties du discours. Ce corpus est divisé en deux avec un sous-ensemble d'apprentissage (46 dialogues) et un sous-ensemble de test (45 dialogues). Chaque sous-ensemble contient approximativement 17 000 unités lexicales, dont 5 400 provenant des clients et 11 600 provenant des téléconseillers.

Un second intérêt du corpus DATCHA91 est qu'il a été manuellement corrigée permettant ainsi d'évaluer la proportion d'erreurs présentes dans les conversations. Dans les travaux de NASR et al. [Nas+16], deux métriques sont utilisées : le taux d'erreurs mots (*Word Error Rate* en anglais) et le taux de messages avec substitution (TMS). Le TEM est le nombre de mots erronés en proportion du nombre total de mots et le TMS est le nombre de messages contenant au moins une erreur en proportion du nombre total de messages.

On peut constater dans le corpus que 4,5% des mots sont mal orthographiés. Ceci reste relativement faible mais ce taux est beaucoup plus élevé chez le clients (10,5%) que chez le téléconseiller (1,5%). De plus, ils ont pu constater que 27,2% des messages contiennent au moins un mot mal orthographié, ce taux étant de nouveau beaucoup plus important lorsqu'on considère uniquement les messages du clients (41,3%). Une autre information intéressante est que les écarts-types du TEM et du TMS chez le client sont très élevés, ce qui montre bien que d'un client à un autre, les compétences linguistiques et l'attention apporté à la conversation sont très variables. Cette étude motive donc bien le fait d'analyser l'influence que peuvent avoir ces erreurs très présentes sur une tâche morphosyntaxique et si c'est le cas de voir s'il est possible de diminuer l'influence de ces erreurs.

Pour chaque correction réalisée, le type de l'erreur est également indiqué. La typologie utilisée, également issue des travaux de NASR et al. [Nas+16] mais légèrement simplifiée pour mes travaux, est présentée dans la table 4.1.

Concernant l'annotation en parties du discours, les étiquettes utilisées ainsi que la proportion de chaque étiquette dans le corpus sont présentées dans la table 4.2. Cette table présente également le TEM par partie du discours. Il est intéressant de constater que les taux les plus élevés concernent principalement les parties du discours correspondant à des mots porteurs de sens (classes ouvertes). Cependant, même certains mots fonctionnels tels que les prépositions ou les pro-

Type d'erreur	Description
DIACR	Absence ou ajout erronés de signes *diacritiques*
APOST	*Apostrophe* manquante ou mal positionnée
AGGLU	*Agglutination* de plusieurs mots en un seul
SPLIT	Mot *scindé* en plusieurs mots
INFPP	Confusion entre participe passé et infinitif
INFL	Autre erreur de *flexion*, c-à-d. une erreur de conjugaison, de genre ou de nombre
MOD1C	Substitution, deletion ou insertion d'une lettre ou échange de deux lettres
OTHER	Autre type d'erreur

TABLE 4.1. – Typologie des différentes erreurs possibles sur les mots

noms ont des TEM assez élevés avec respectivement des valeurs de $3{,}5\%$ et $5{,}2\%$, qui en conjonction avec le nombre élevé d'occurrence de ces parties du discours dans le corpus font que ces erreurs sont très présentes dans le corpus. Ceci est également vrai pour les noms, verbes, adverbes et adjectifs.

4.3. Correction automatique du corpus

Maintenant que DATCHA91 est corrigée manuellement, j'ai à ma disposition deux versions du corpus : l'une corrigée, l'autre brute (non modifiée). Ces deux variantes me permettraient déjà de réaliser des expérimentations afin d'évaluer l'influence que peuvent avoir les fautes d'orthographe sur la prédiction des parties du discours.

Une hypothèse que je fais est que les fautes d'orthographe auront une influence négative sur les prédictions. En effet, il serait surprenant que des erreurs transformant une préposition en verbe (« à » en « a ») n'aient pas d'influence négative. Autre exemple, si je présente à la machine la phrase « Un bèl home competent » (au lieu de « Un bel homme compétent »), il lui sera difficile de déterminer les mots qui sont des noms et ceux qui sont des adjectifs — les mots devenant inconnus à cause des erreurs. Si cette hypothèse est vraie, il serait alors intéressant de déterminer s'il est possible d'appliquer des prétraitements aux données permettant de limiter les erreurs.

Pour répondre à cette question, une première approche retenue est de corriger automatiquement, à l'aide d'un correcteur orthographique, le corpus DATCHA91 avant de le donner en entrée d'un étiqueteur en parties du discours. Afin d'éviter d'insérer de nouvelles erreurs, nous ne corrigeons que les mots hors-vocabulaire, c.-à-d. les mots pour lesquels les fautes d'orthographe créent des mots inexistants dans la langue française. Bien entendu, ce choix aura pour conséquence que cer-

Partie du discours	Étiquette	Proportion (en %)	TEM (en %)
Verbe au participe présent	VER :ppre	0,3	0,0
Déterminant	DET	13,2	1,3
Nom propre	NAM	1,7	1,5
Interjection	INT	2,1	1,5
Pronom relatif	PRO :REL	0,8	1,6
Conjonction	KON	4,6	1,8
Nombre	NUM	2,0	2,4
Verbe à l'impératif	VER :imp	0,9	3,1
Préposition	PRP	11,9	3,5
Verbe à l'infinitif	VER :inf	5,1	4,6
Pronom	PRO	13,7	5,2
Adverbe	ADV	6,9	5,6
Verbe	VER	10,9	5,8
Adjectif	ADJ	3,9	6,7
Nom	NOM	19,6	6,7
Abréviation	ABR	0,2	10,0
Verbe au participe passé	VER :pper	2,2	16,9

TABLE 4.2. – Le jeu d'étiquettes utilisé pour l'annotation en parties du discours ainsi que la proportion de chaque étiquette dans le corpus

taines erreurs graves pouvant faire passer un mot d'une catégorie de partie du discours à une autre resteront dans le corpus corrigé automatiquement. Mais ce choix permet de ne pas insérer de nouvelles erreurs et me permet d'éviter d'ajouter des facteurs de variabilités supplémentaires qui pourraient influencer les résultats des expérimentations. Il est d'abord nécessaire de déterminer si l'utilisation de prétraitements sur les données est en effet utile sur des tâches morphosyntaxiques avant d'utiliser des correcteurs orthographiques plus ambitieux.

La correction automatique d'un mot hors-vocabulaire se fait en trois étapes :

1. À l'aide d'une mesure de similarité entre chaine de caractères, on génère une liste ordonnée de mots se trouvant dans le lexique de mots corrects pouvant correspondre au mot erroné (section 4.3.2).
2. Les mots de la liste sont ensuite réordonnés à l'aide d'une distance distributionnelle. Ceci suppose l'apprentissage de plongements de mots pour les mots bien orthographiés et mal orthographiés (section 4.3.3).
3. Le premier mot de la liste remplace le mot erroné.

Pour la première étape, il est nécessaire de construire un lexique des mots corrects. La section 4.3.1 décrit le processus de création de ce lexique.

4.3.1. Définir le lexique de mots corrects

Afin de pouvoir définir quels sont les mots mal orthographiés, il est nécessaire de construire un lexique des mots corrects. Il n'est pas possible de définir un lexique exhaustif, il est donc important de construire un lexique permettant à la fois de prendre en compte les termes du langage généraliste mais également les termes utilisés principalement dans le contexte de l'entreprise, comme par exemple « livebox » ou « décodeur » dans le cadre d'Orange. Ceci est important car le correcteur doit éviter de proposer des termes trop éloigné du domaine de l'assistance clientèle, tout en ayant la faculté de proposer une alternative proche pour tous les mots erronés. Afin de prendre en compte ces deux besoins, le lexique est construit à partir des mots apparaissant au moins 500 fois dans la version française de Wikipédia, permettant d'obtenir un lexique de $36\,420$ mots. Pour prendre en compte les spécificités du domaine de l'assistance clientèle, 388 mots sélectionnés manuellement sont également ajoutés au lexique.

4.3.2. Correction automatique basées sur des distances d'édition

Le correcteur s'appuie sur la distance de Damerau-Levenshtein (DDL) [Dam64]. L'algorithme détermine la distance entre deux chaines de caractères en calculant le nombre minimal de transformation nécessaire pour passer d'une chaine à l'autre. Les opérations de transformations sont l'insertion, la suppression et la substitution d'un caractère mais aussi la transposition de deux caractères adjacents. À la différence de la version standard de la DDL, des poids sont attribués aux types des erreurs :

— l'absence ou l'ajout superflu de signes diacritiques ajoutent seulement $0{,}3$ à la distance d'édition ;

— les lettres adjacentes sur le clavier ajoutent $0{,}9$ à la distance d'édition ;

— la transposition de lettres adjacentes dans le mot n'ajoutent que $0{,}9$ à la distance d'édition ;

— toutes les autres différences ajoutent 1 à la distance.

Le correcteur produit une liste de candidats possibles pour la correction, la liste étant limitée aux mots ayant une distance d'édition inférieur à un seuil. Ce seuil est dynamique en fonction du nombre de lettres dans le mot. Le code source du correcteur est disponible en ligne [1].

1. https://github.com/Orange-OpenSource/lexical-corrector

4.3.3. Utilisation de plongements de mots pour réordonner les corrections

Une limite de la DDL est qu'elle ne se fonde que sur les caractères. On a pu voir dans le chapitre 2 que les signes d'un mot sont arbitraires et de ce fait, il n'y a pas toujours un lien entre deux mots ayant des signes proches. Les plongements de mots permettent de donner une réponse à ce problème en construisant une représentation du sens (ou tout du moins des contextes de production du mot).

Le but est donc d'utiliser les plongements de mots pour réordonner les mots corrects de la liste générée par le correcteur en prenant en compte le sens des mots. Pour chaque proposition de correction de la liste, on calcule la similarité cosinus entre le plongement du mot à corriger et le plongement de la proposition de correction. Ceci permet d'obtenir des valeurs entre -1 (sens très différents) et 1 (sens très similaires) pour chaque proposition de la liste.

Pour pouvoir calculer ces distances, il est nécessaire d'avoir des plongements de mots aussi bien pour les mots corrects que pour les mots erronés. Étant donné que nous avons à notre disposition le très grand corpus de tchats DATCHAATH, on peut s'attendre à y rencontrer la plupart des mots mal orthographiés usuellement rencontrés. En outre, ces plongements de mots auront l'intérêt d'être adaptés à nos données d'assistance clientèle, ce qui n'aurait pas été le cas si nous avions utilisé un autre corpus. Pour ces expérimentations, nous utilisons WORD2VEC [Mik+13] avec une fenêtre de taille 4. De plus, afin de prendre en compte un maximum de mots possibles, tous les mots apparaissant au moins 2 fois sont conservés lors de l'entrainement. Ainsi, le lexique produit contient $43\,300$ formes de mots différentes.

À partir des plongements de mots (et des mesures de similarités calculées), il est possible de redéfinir la distance d'édition entre chaines de caractères afin de prendre en compte le sens des mots. L'idée est de définir une distance d_{emb} à partir de la mesure de similarité entre plongements qui pourra être utilisée pour pondérer la distance d'édition retournée par la DDL. Plus formellement, d_{emb} est calculé comme suit :

$$d_{emb}(w_e,w_p) = \begin{cases} 1 - \cos(w_e,w_p), & \text{si } w_e \in V_{emb} \text{ et } w_p \in V_{emb} \\ 1, & \text{sinon} \end{cases}$$

où w_e est le mot erroné à corriger, w_p une proposition de correction et V_{emb} le lexique des plongements de mots.

Soit $C(w_e,w_p)$ la distance d'édition fourni par le correcteur lexical entre w_e et w_p, la nouvelle fonction de cout $C_{emb}(w_e,w_p)$ fondée sur les plongements de mots est définie comme suit :

$$C_{emb}(w_e,w_p) = d_{emb}(w_e,w_p) \cdot C(w_e,w_p)$$

À partir de là, il suffit de recalculer le nouveau classement des propositions de corrections en réordonnant la liste de mots données par le correcteur lexical précédent à l'aide de la fonction de cout C_{emb}.

Une fois le nouveau classement obtenu, il suffit alors de choisir le mot le mieux classé pour remplacer le mot erroné dans le corpus.

4.4. Étiqueteur en parties du discours sur les données Datcha

L'étiqueteur en parties du discours utilisé lors de mes expérimentations est fondé sur des GRU [Cho+14]. Les GRU sont un type de RNN qui ont un mode de fonctionnement proche de celui des LSTM (introduits dans le chapitre 2). La différence est que les GRU réduisent le nombre de paramètres à apprendre, et donc réduisent le temps de calcul nécessaire. Pour de nombreuses tâches, les performances entre les LSTM et les GRU sont comparables.

L'utilisation d'un GRU plutôt qu'un type de réseau non récurrent est justifiée par le fait que les RNN considèrent les entrées comme des séquences ordonnées et utilisent les éléments passés de la séquence pour les prédictions de l'élément courant. Ceci est donc particulièrement intéressant pour des tâches d'étiquetage de séquences telles que l'étiquetage en parties du discours. Mon étiqueteur utilise des GRU bidirectionnels permettant la prise en compte du contexte gauche et du contexte droit pour chaque mot d'une phrase donnée.

L'entrée du réseau est la séquence de mots associés à divers traits simples. Les mots sont encodés à l'aide d'une table de recherche qui associe chaque mot avec sa représentation en plongement de mots. Les plongements de mot peuvent être initialisés avec des plongements pré-entrainés ou alors être directement appris lors de l'apprentissage du modèle. Pour les traits, une valeur booléenne est utilisée afin d'indiquer la présence ou non d'une lettre majuscule dans le mot et des vecteurs de type « one-hot » sont également utilisés pour indiquer quels sont les suffixes de tailles 3 et 4 du mot. Enfin, un lexique extérieur est également exploité indiquant quelles sont les parties de discours possibles pour le mot donné. Cette information est représentée sous la forme d'un vecteur binaire. Le lexique utilisé est le *Lefff* [Sag10]. Étant donné que je réalise une classification non-binaire, j'utilise en couche de décision la fonction d'activation *softmax* et durant l'entrainement la fonction de perte est l'entropie-croisée. La descente du gradient se fait grâce à l'algorithme Adam [KB15].

Le corpus d'apprentissage de l'étiqueteur est le sous-ensemble d'apprentissage du corpus DATCHA91 corrigé manuellement.

Pour rappel, un des objectifs de ce chapitre est de déterminer si des prétraitements peuvent être utiles pour limiter l'influence des erreurs orthographiques lors de la prédiction de parties du discours. La première approche, introduite précédemment, est de corriger automatiquement les entrées de l'étiqueteur.

Une autre possibilité est de faire en sorte que les mots erronés soient dans un même espace de représentation que les mots corrects. Le but étant que les représentations des mots erronés et corrects utilisés dans les mêmes contextes soient proches et que l'étiqueteur ne puisse pas faire la différence entre les deux. Cette approche peut être vue comme une forme allégée de la correction automatique où on ne construit plus explicitement un classement des meilleures corrections possibles, on laisse l'étiqueteur le faire de manière implicite. Afin de pouvoir déterminer si cette approche est intéressante, trois étiqueteurs utilisant des plongements différents sont entrainés :

— un étiqueteur n'utilisant pas de plongements de mots pré-entrainés, les plongements sont appris durant l'apprentissage de l'étiqueteur ;

— un étiqueteur utilisant des plongements WORD2VEC appris sur DATCHAATH brut (c.-à-d. les données du corpus ne sont pas modifiées) ;

— un étiqueteur utilisant des plongements WORD2VEC appris sur DATCHAATH corrigé automatiquement en utilisant le correcteur présenté dans la section précédente.

Je ne m'attends pas à ce que les plongements appris directement par l'étiqueteur permettent d'obtenir les meilleures prédictions, le corpus d'apprentissage étant trop petit en plus d'être corrigé manuellement (c.-à-d. qu'il n'y aura pas de plongements pour les mots erronés). Cependant, cette configuration permet d'avoir un point de référence à battre pour les deux autres types de plongements. Les deux autres configurations permettent dans un premier temps de déterminer s'il est en effet utile d'avoir des plongements pour les mots erronés. Dans un second temps, je souhaite déterminer s'il y a un intérêt à combiner les deux types de prétraitements possibles.

4.5. Évaluation de l'influence des erreurs

Afin d'évaluer l'influence des erreurs sur la prédiction des parties du discours, je souhaite comparer les performances de l'étiqueteur en partie du discours en expérimentant avec les deux types de prétraitements présentés : la correction automatique et la construction de plongements de mots pour les mots erronés.

Le premier objectif des expérimentations est d'évaluer si le fait d'avoir des données bruitées a une influence sur les performances d'un étiqueteur en partie du discours. Dans l'hypothèse où l'étiqueteur ne serait pas capable seul de compenser les erreurs orthographiques, il sera donc nécessaire de réaliser des prétraitements pour limiter les erreurs. Les autres objectifs des expérimentations sont alors les suivants :

1. Déterminer si la correction automatique du corpus de test permet de limiter les erreurs de prédiction de l'étiqueteur. L'idée est que le correcteur automatique permettra aux données de test — initialement très bruitées — de

se rapprocher de la qualité des données d'apprentissage qui ont été corrigées manuellement. Étant donné qu'on ne corrige pas toutes les erreurs possibles, on ne peut pas s'attendre à totalement résoudre le problème avec cette seule approche.

2. Déterminer si le fait d'apprendre des plongements pour les mots corrects et erronés permet de masquer les erreurs en construisant les mêmes représentations pour les mots corrects et erronés qui apparaissent dans les mêmes contextes.
3. Dans le cas où les deux prétraitements ne suffisent pas individuellement à limiter l'influence des erreurs orthographiques, déterminer s'il est intéressant de combiner les deux approches.

Il est peu probable que les deux approches permettent de totalement résoudre le problème de l'influence des mots erronés sur une tâche morphosyntaxique. En effet, les deux approches n'apportent pas de solutions pour les erreurs qui transforment un mot en un autre mot existant. On peut cependant espérer qu'en corrigeant une partie des erreurs, l'étiqueteur puisse plus facilement utiliser le contexte gauche et droit d'un mot pour déterminer sa partie du discours.

Pour l'évaluation, trois versions du sous-ensemble de test de DATCHA91 sont utilisées :

1. MANUEL : corrigée manuellement, correspond à la version de référence. Ce sous-ensemble nous permet d'évaluer les performances maximales qu'on peut espérer obtenir avec l'étiqueteur utilisé.
2. BRUT : sans correction. Correspond aux conditions que l'on a à la base si on ne fait aucun prétraitement sur les données. Ce sont à priori les conditions les plus difficiles pour l'étiqueteur.
3. AUTOMATIQUE : corrigée automatiquement par le correcteur orthographique. Ce sous-ensemble doit nous permettre de déterminer si le fait de corriger automatiquement les données en entrée permet de réduire les erreurs de prédictions de l'étiqueteur.

Les trois types d'étiqueteurs (utilisant des plongements différents) sont utilisés sur chacun de ces sous-ensembles.

Afin de comparer les performances des différents modèles sous différentes configurations, la qualité des prédictions en parties de discours est évaluée grâce à l'exactitude, définie par :

$$\text{exactitude} = \frac{\text{Nombre de parties du discours correctement prédites}}{\text{Nombre de mots dans le corpus de test}}$$

La table 4.3 présente les résultats obtenus par les différents modèles. Outre le calcul de l'exactitude sur l'ensemble du corpus de test, les calculs de l'exactitude en ne prenant en compte que les tours de parole du téléconseiller ou du client sont également donnés.

Corpus	Tous	Téléconseiller	Client
	Étiqueteur avec plongements non préentrainés		
MANUEL	95,37	96,39	93,39
AUTOMATIQUE	93,83	95,52	90,54
BRUT	93,07	95,31	88,70
	Étiqueteur avec plongements appris sur DATCHAATH brut		
MANUEL	95,36	96,37	93,40
AUTOMATIQUE	94,25	95,78	91,25
BRUT	94,01	95,77	90,60
	Étiqueteur avec plongements appris sur DATCHAATH corrigé		
MANUEL	95,35	96,35	93,42
AUTOMATIQUE	94,13	95,62	91,24
BRUT	93,43	95,52	89,37

TABLE 4.3. – Exactitudes (en %) des prédictions des parties du discours des différents étiqueteurs

En observant les résultats obtenus sur le corpus MANUEL, on constate que quels que soient les plongements de mots utilisés, les résultats sont les mêmes avec une exactitude de $95{,}4$% sur l'ensemble des tours, $96{,}4$% sur ceux du téléconseiller et $93{,}4$% sur ceux du client. Ces scores correspondent aux exactitudes maximales que nous pouvons espérer atteindre avec le type d'étiqueteur utilisé. Ce sont les tours du client qui provoquent le plus d'erreurs et ceci même avec les corrections orthographiques. Les autres types d'erreurs (par exemple grammaticales) n'étant pas corrigées, ce résultat n'est pas étonnant.

À contrario, en observant les résultats sur la configuration la plus difficile (c.-à-d. sur le corpus BRUT et en n'utilisant pas de plongements pré-entrainés), les scores d'exactitude sont de $93{,}1$% sur l'ensemble des tours, $95{,}3$% sur ceux du téléconseiller et $88{,}7$% sur ceux du client. Ces scores permettent de voir l'influence immédiate des erreurs orthographiques sur la tâche. On remarque que l'étiqueteur fait beaucoup d'erreurs de prédiction sur les tours du client à cause des erreurs orthographiques ($-4{,}7$ points) alors que ce phénomène est beaucoup moins important sur ceux du téléconseiller ($-1{,}1$ points). Encore une fois, ce résultat n'est pas surprenant — le client faisant beaucoup plus d'erreurs d'orthographe — mais permet de mettre en lumière le fait qu'il n'est pas possible de négliger la problématique du bruit dans les corpus, celui-ci ayant une influence visible sur les performances des modèles. Étant donné que les améliorations qui peuvent être apportées concernent principalement les tours du client, dans la suite de cette analyse des résultats, je vais me concentrer sur ces tours-ci.

En s'intéressant maintenant aux résultats obtenus en utilisant des plongements

de mots pré-entrainés sur le corpus DatchaAth brut et en s'évaluant toujours sur le corpus Brut, on observe que les plongements appris sur le corpus brut permettent d'obtenir un amélioration de $1{,}9$ points par rapport à la configuration de base ($90{,}6$% contre $88{,}7$%). Ceci permet de montrer que le fait d'entrainer des plongements de mots pour les mots erronés permet de limiter les erreurs de prédiction de l'étiqueteur.

Lorsqu'on observe les résultats obtenus en s'évaluant cette fois-ci sur le corpus Automatique et en n'utilisant pas de plongements pré-entrainés, on constate également une amélioration ($+1{,}8$ points) par rapport à la configuration de base. Les deux méthodes de prétraitement permettent donc, avec des résultats comparables, d'aider l'étiqueteur à limiter les erreurs de prédictions.

Enfin, si on observe les configurations qui combinent les deux approches (corpus Automatique avec des plongements appris sur DatchaAth brut ou corrigé automatiquement), on constate une nouvelle légère amélioration ($+0{,}7$ points) pour arriver à un score d'exactitude de $91{,}25$%. On peut noter que le fait d'apprendre les plongements de mots sur une version corrigée automatiquement du corpus DatchaAth n'a aucun intérêt par rapport à un apprentissage sur Brut. Ceci peut s'expliquer par le fait que le corpus non corrigé contient probablement à la fois les versions erronées et correctes des mots et en quantité suffisante. Le corpus DatchaAth corrigé automatiquement ayant moins de mots erronés produira un lexique moins complet au final, tout en ayant le risque de remplacer une erreur par une autre erreur.

Ces résultats montrent que les deux méthodes de prétraitements peuvent être combinés pour améliorer très légèrement leur efficacité. Au final, celles-ci permettent (individuellement et combinées) d'obtenir des scores, certes toujours plus bas que si les phrases étaient corrigées manuellement, mais qui permettent de bien limiter les erreurs dues aux fautes d'orthographe.

Les résultats précédents nous permettent de voir les scores sur l'ensemble des prédictions mais ne permettent pas de voir si certaines erreurs orthographiques sont plus à même à provoquer des erreurs de prédiction des parties du discours. La table 4.4 permet d'analyser l'influence des types d'erreur orthographique sur la tâche d'étiquetage en parties du discours. Chaque ligne de la table correspond à un type d'erreurs orthographiques. La partie gauche de la table présente les résultats obtenus en utilisant en entrée de l'étiqueteur le corpus Brut alors que la partie droite présente les résultats obtenus sur le corpus Automatique. L'étiqueteur utilisé est celui utilisant les plongements de mots appris sur le corpus DatchaAth brut.

Si on s'intéresse dans un premier temps aux résultats sur les entrées brutes (c.-à-d. en utilisant que le prétraitement sur les plongements de mots), la table 4.4 montre que le type d'erreurs sur lequel l'étiqueteur se trompe presque toujours est INFPP, c.-à-d. les erreurs de flexions entre participe passé et infinitif. En effet, l'étiqueteur a un taux de réussite de seulement 10% sur les mots ayant ce type d'erreur là. Ceci semble indiquer que l'étiqueteur se base énormément sur le

Type d'erreur orthographique	Brut		Automatique	
	# err. ortho.	# err. POS	# err. ortho.	# err. POS
DIACR	250	96	81	65
APOST	47	5	11	3
MOD1C	135	44	77	26
AGGLU	57	47	54	46
SPLIT	31	24	31	24
INFPP	29	26	29	26
INFL	84	9	77	8
OTHER	50	20	40	22

Table 4.4. – Nombre d'erreurs de prédictions des parties du discours (POS) comparées au nombre de mots erronés pour un type d'erreur orthographique donné sur des entrées brutes et des entrées corrigées automatiquement

suffixe du mot pour réaliser cette prédiction et non sur le contexte de production du mot. De fait, une erreur orthographique sur ces mots là entraine presque inévitablement une erreur de prédiction de la partie du discours.

Les erreurs orthographiques de type AGGLU et SPLIT sont également une grande source d'erreurs pour l'étiqueteur avec des scores d'exactitudes de seulement 17,54% et 22,58% respectivement. Ici, la raison est probablement que ces deux erreurs suppriment ou ajoutent des unités lexicales à la phrase. Ceci va à la fois empêcher l'apprentissage de plongements de mots pertinents mais également nuire à l'étiqueteur dans la prise en compte du contexte de production.

À l'inverse, les erreurs de type APOST et INFL n'ont que très peu d'influence sur les performances de l'étiqueteur qui obtient des exactitudes de 89,36% et 89,29% respectivement. Dans le cas des erreurs de type APOST, les mots concernés sont très probablement présents dans le corpus DatchaAth sous leur formes correctes et incorrectes étant donné que ce sont généralement des mots-outils. Pour les erreurs de flexions, ce sont généralement des fautes d'accord et donc cela n'impacte pas la partie du discours dans ces cas-là.

Si on compare ces différents résultats avec ceux obtenus sur des entrées cette fois-ci corrigées automatiquement (c.-à-d. en combinant les deux approches de prétraitement), on constate que seules les erreurs DIACR, APOST et MOD1C qui sont véritablement corrigées. Ceci n'est pas surprenant car ce sont ces erreurs qui vont généralement créer des mots hors-vocabulaire — seules erreurs prises en charge par le correcteur.

Pour les erreurs de diacritiques, on peut constater que le processus de correction corrige 67% des erreurs orthographiques (de 250 erreurs à 81) mais ne permet de diminuer que de 32% les erreurs d'étiquetages en parties du discours (de 96 erreurs à 65). La correction automatique a donc permis de corriger quelques

erreurs mais ce n'est pas le cas pour la grande majorité d'entre elles. Cette non amélioration est due au fait que la majorité des erreurs de diacritiques restantes sont des erreurs graves puisqu'elles modifient les mots en des mots existants mais ayant des parties de discours très différentes, par exemple « a » et « à » ou encore « ou » et « où ».

Pour MOD1C, on constate une réduction des erreurs orthographiques de 57% et une réduction des erreurs d'étiquetage de 59%. Pour cette erreur, il est donc intéressant de réaliser une correction automatique du corpus. Ceci peut simplement s'expliquer par le fait que ce type d'erreur va presque systématiquement créer des mots hors-vocabulaire et que l'utilisation de la DDL est particulièrement adaptée pour ces erreurs.

Pour les autres erreurs, même lorsqu'il y a une réduction du nombre d'erreurs orthographiques, la réduction du nombre d'erreurs d'étiquetage est très faible ou nulle. Pour observer une réduction, il faudrait avoir un correcteur orthographique beaucoup plus performant qui pourrait prendre en compte tous les mots erronés lors de la correction.

4.6. Discussion

Dans ce chapitre, j'ai analysé un sous-ensemble du corpus DATCHA afin de déterminer s'il est nécessaire de traiter en amont de toutes tâches le problème des erreurs introduites par les scripteurs dans les mots produits. Ici, je me suis concentré sur une tâche simple d'étiquetage en parties du discours. Les résultats obtenus nous permettent de conclure qu'il semble être souhaitable d'avoir un corpus entièrement corrigé manuellement afin d'obtenir les meilleures performances sur cette tâche. Cependant, une correction manuelle de l'ensemble du corpus n'est pas réaliste et donc la question se pose de savoir s'il est utile ou non d'appliquer des prétraitements aux entrées de l'étiqueteur avant de les utiliser.

Sur la tâche étudiée ici, il ne semble pas nécessaire de réaliser une correction automatique des entrées. En effet, même si on peut constater un très léger gain, celui-ci ne justifie probablement pas son utilisation systématique, qui reste relativement lourde à mettre en place. De plus, cela a l'inconvénient de totalement supprimer les fautes d'orthographe qui peuvent potentiellement être une information utile pour certaines tâches.

En revanche, l'utilisation de plongements de mots appris sur un ensemble important de conversations est un prétraitement intéressant. En effet, ceux-ci permettent d'obtenir des gains légèrement plus intéressants que ceux obtenus par un système ne reposant que sur la correction automatique, tout en ne constituant pas une véritable étape de prétraitement additionnelle. En effet, les plongements de mots étant désormais systématiquement utilisés dans des modèles neuronaux, l'apprentissage de plongements de mots est de toute manière effectuée que ce soit en les préentrainant en amont, soit en les apprenant directement sur les

données d'apprentissage lorsque celles-ci sont suffisamment importantes.

Dans les chapitres suivants, je m'intéresserai à explorer différentes approches permettant de réaliser des analyses du discours. Même si je me suis fondé sur une tâche morphosyntaxique pour analyser l'influence des fautes d'orthographe sur les prédictions, on peut tout de même utiliser ces résultats pour décider des prétraitements à appliquer ou pas sur les données. En effet, dans le discours c'est plutôt les énoncés qui constituent les unités de base et il est donc probable que les fautes d'orthographe n'influencent pas autant les prédictions que sur une tâche morphosyntaxique. Au vu de l'étude réalisée dans ce chapitre, il ne semble pas être nécessaire de réaliser une correction automatique des fautes mais par contre d'au moins utiliser des plongements de mots adaptés à mon corpus.

Concernant les plongements de mots, ceux utilisés dans les expérimentations précédentes sont relativement simples du fait qu'ils considèrent chaque forme d'un mot comme un seul bloc, ne permettant donc pas de déterminer des représentations pour des mots totalement inconnus. Le fait d'avoir un corpus très volumineux comme DATCHA permet déjà de limiter ce phénomène étant donné qu'il est très probable de rencontrer la plupart des fautes généralement commises par les scripteurs. Il est également possible d'utiliser des modèles de plongements de mots permettant de construire des représentations à partir de sous-mots en particulier pour les mots jamais rencontrés. Il existe plusieurs approches telles que FASTTEXT [Boj+17] ou l'encodage par paires d'octets [SHB16] permettant ce découpage en sous-mots. Celles-ci seront utilisées dans les différentes contributions de la thèse.

Chapitre 5.

Prédire la qualité des interactions avec une méthodologie bout en bout

Sommaire

5.1 Introduction . . . 111
5.2 Des questionnaires de satisfaction utilisés comme supervision . . . 113
5.2.1 Liens entre les différentes questions . . . 114
5.2.2 Le *Net Promoter Score* . . . 115
5.3 La satisfaction client comme support pour des modèles bout en bout 116
5.3.1 Estimation de la difficulté de la tâche . . . 118
5.3.2 Étude de la nature des erreurs de prédiction . . . 125
5.3.3 Limiter les confusions entre classes extrêmes . . . 127
5.3.4 Conclusion . . . 134
5.4 Influence du contenu du dialogue sur les prédictions . . . 135
5.4.1 Influence des scripteurs . . . 135
5.4.2 Étude de l'importance du lexique . . . 137
5.5 Conclusion . . . 139

5.1. Introduction

La construction de structures explicites du discours posent des problèmes complexes sur le choix des schémas d'annotation et de la production de cette annotation sur un ensemble volumineux de données dans le but d'apprendre un modèle d'analyse du discours. Une solution à cette problématique est de se reposer sur une tâche support, indirectement liée à l'analyse du discours, afin de construire un modèle de type bout en bout. Le modèle construit peut ensuite être utilisé pour extraire des représentations distributionnelles, à la manière des plongements de mots ou de phrases, qui modéliseront en partie le discours conversationnel à l'échelle du dialogue ou du tour de parole.

Un grand intérêt du corpus DATCHA est qu'il contient un nombre très conséquent de conversations — ce qui rend l'apprentissage de représentations distributionnelles réaliste. On a également constaté dans le chapitre précédent que

la nature bruitée des données ne pose pas de problèmes dans l'utilisation de celles-ci sur des tâches du TAL — l'utilisation de plongements de mots adaptés permettant de limiter l'influence des erreurs orthographiques.

Néanmoins, afin d'apprendre des représentations à l'aide d'un modèle bout en bout, il est nécessaire d'avoir à disposition une tâche support sur laquelle réaliser l'apprentissage du modèle. Or, nous avons à notre disposition peu de données annotées manuellement :

— 91 conversations annotées dans DATCHA91 : parties du discours, analyse syntaxique et corrections manuelles ;

— environ 3 000 conversations annotées en actes de dialogue dans DATCHAACT.

DATCHA91 est beaucoup trop petit pour apprendre des représentations distributionnelles pertinentes et les annotations disponibles n'ont pas de liens avec le discours conversationnel. Quant à DATCHAACT, celui-ci n'est annoté qu'en actes de dialogue. Bien que ceux-ci permettent de modéliser le discours de surface, ils ne tiennent compte que des fonctions de communication des énoncés sans chercher à identifier les relations entre énoncés — le contexte nécessaire pour les prédire est extrêmement local et tient principalement compte du contenu de l'énoncé correspondant.

Une particularité du corpus DATCHA est qu'en plus des dialogues, il contient également diverses métadonnées associées à chacune des conversations. Parmi ces métadonnées, on y retrouve les réponses à un questionnaire de satisfaction auquel chaque client a la possibilité de répondre à l'issue de sa conversation avec le téléconseiller.

Prédire la satisfaction client à l'issue d'une conversation n'est pas un problème nouveau [Boc+17 ; Luq+17]. Néanmoins, les travaux existants portent sur des conversations téléphoniques ce qui a pour effet de limiter la quantité de données disponibles. En outre, les caractéristiques utiles à la prédiction de la satisfaction client ne sont pas nécessairement les mêmes à l'orale et à l'écrit (par exemple, la prosodie ne peut pas être utilisée à l'écrit). De ce fait, le corpus DATCHA permet de développer des approches spécifiques au langage tchat. La question centrale est alors de savoir si le contenu seul (c.-à-d. sans prendre en compte des facteurs extérieurs comme la fidélité du client ou le temps d'attente) est suffisant pour juger de la qualité des interactions dans un dialogue.

Par ailleurs, un second enjeu est de savoir s'il existe un lien entre la satisfaction du client et le discours conversationnel, c.-à-d. savoir si la satisfaction se traduit dans la conversation par la production d'enjeux dialogiques précis et de schémas d'interactions particuliers. Si ce lien existe, il serait alors envisageable d'utiliser cette tâche comme d'un proxy [1] pour construire des représentations du discours conversationnel à l'aide d'une approche bout en bout.

Dans ce chapitre, je vais développer et améliorer des modèles de prédiction de la satisfaction qui pourront ensuite être utilisés afin de déterminer si un lien

1. Tâche qu'il est possible d'utiliser pour résoudre une autre tâche de manière indirecte.

entre satisfaction client et discours conversationnel existe. En complément, on étudiera ce que la satisfaction client permet de révéler de la structure du dialogue. Je vais dans un premier temps décrire dans la section 5.2 le questionnaire de satisfaction proposé aux clients. Dans la section 5.3 je présenterai divers modèles bout en bout pour prédire la satisfaction client. En particulier, j'identifierai des problèmes rencontrés par les modèles dans les prédictions qu'ils produisent et je proposerai diverses solutions pour les résoudre. Enfin, la section 5.4 permettra de déterminer si les interactions entre scripteurs peuvent être modélisés par un tel modèle construit pour prédire la satisfaction client.

5.2. Des questionnaires de satisfaction utilisés comme supervision

Le corpus de tchats DATCHA a l'avantage d'être très volumineux. En outre, le fait qu'il soit construit à partir de tchats permet en théorie d'y ajouter facilement de nouvelles données (en effet, il n'est pas nécessaire de réaliser de prétraitements, comme ça peut être le cas à l'oral par exemple). Cependant, se pose toujours le problème de l'annotation. En effet, il n'est pas raisonnable de faire annoter un sous-ensemble volumineux du corpus par un humain, les annotations voulues étant généralement non triviales et la masse de données rend de toute manière cette tâche irréaliste car trop couteuse.

Il existe cependant un moyen d'obtenir une annotation d'une très grande partie du corpus en faisant en sorte qu'elle soit réalisée en même temps que la conversation par les scripteurs eux-mêmes. Bien entendu, dans le cadre d'une assistance clientèle, il ne va pas être demandé aux clients de réaliser des annotations permettant de prendre en compte le discours conversationnel directement. En revanche, Orange souhaitant savoir si les services rendus à ses clients sont de bonne qualité et si les clients sont satisfaits de ces prestations, un questionnaire de satisfaction est proposé à l'issue de toutes les conversations. Les questions proposées aux clients sont répertoriées dans la table 5.1. Dans cette table, on peut également trouver des alias identifiant les questions qui seront utilisés dans la suite du document pour faire référence à celles-ci.

Un intérêt pour moi est que ces questions portent sur différents aspects des interactions entre les deux scripteurs dans les conversations. Certaines portent explicitement sur les interactions qui ont eu lieues dans la conversation (Accompagnement, Écoute, Conseil), d'autres sont plus indirectement liées. L'expérience du client à l'issue de la conversation peut être liée aux questions Solution et Recontacter alors que la question Recommander permet d'obtenir une appréciation plus générale pour laquelle les clients peuvent exprimer un ressenti portant sur davantage que la simple conversation. Cette dernière question est alors une sorte de synthèse des autres questions. Sur celle-ci, les clients doivent donner une note entre 0 et 10. Sur toutes les autres questions, les clients doivent

Question	Alias
J'ai été accompagné(e) et j'ai eu les explications pour faire par moi-même	Accompagnement
J'ai été écouté(e) et ma demande a été prise en charge	Écoute
J'ai été bien conseillé(e)	Conseil
Le temps d'attente avant votre mise en relation avec un conseiller, vous a-t-il paru ?	Attente
Suite à cet eChat, pensez-vous avoir besoin de recontacter votre Service Clients Orange ?	Recontacter
La solution proposée par Orange me convient	Solution
Suite à votre contact avec le Service Clients, recommanderiez-vous Orange à vos proches ?	Recommander

TABLE 5.1. – Les différentes questions du questionnaire de satisfaction

répondre sur une échelle de 5 niveaux allant de « Très court » à « Beaucoup trop long » pour la question Attente et allant de « Pas du tout satisfait(e) » à « Très satisfait(e) » pour les questions restantes.

En prenant en compte ces annotations « gratuites », le corpus DATCHASAT est ainsi construit à partir du corpus DATCHAFÉVRIER, décrit dans le chapitre 3 dans la section 3.2.2, car c'est uniquement sur ces conversations-ci que les questionnaires ont pu être récupérés. Le corpus DATCHASAT est uniquement composé des conversations pour lesquelles une réponse est apportée à chacune des questions considérées. Ce corpus est constitué de 80 381 conversations, correspondant à 36% des dialogues présents dans DATCHAFÉVRIER. Ceci correspond à 48 229 conversations dans le corpus d'entrainement, 16 076 dans le corpus de développement et 16 076 dans le corpus de test.

5.2.1. Liens entre les différentes questions

Avant de faire quoi que ce soit avec les réponses aux différentes questions, il est important de bien cerner la nature exacte des réponses apportées par les clients.

L'aspect qui nous intéresse le plus est de savoir si les réponses apportées par les clients aux questions sont véritablement indépendantes. En effet, on peut s'attendre à ce que les clients aient tendance à répondre globalement de la même manière à toutes les questions, c.-à-d. en donnant un avis plutôt positif à toutes les questions s'ils sont contents de manière générale, et inversement s'ils sont mécontents. Dans la figure 5.1, les corrélations de Spearman entre les différentes questions pour lesquelles les réponses possibles ont un ordre sont présentées.

On peut constater que toutes les questions, sauf Attente, sont fortement corrélées. La question Recommander a une corrélation d'environ 0,7 avec les autres

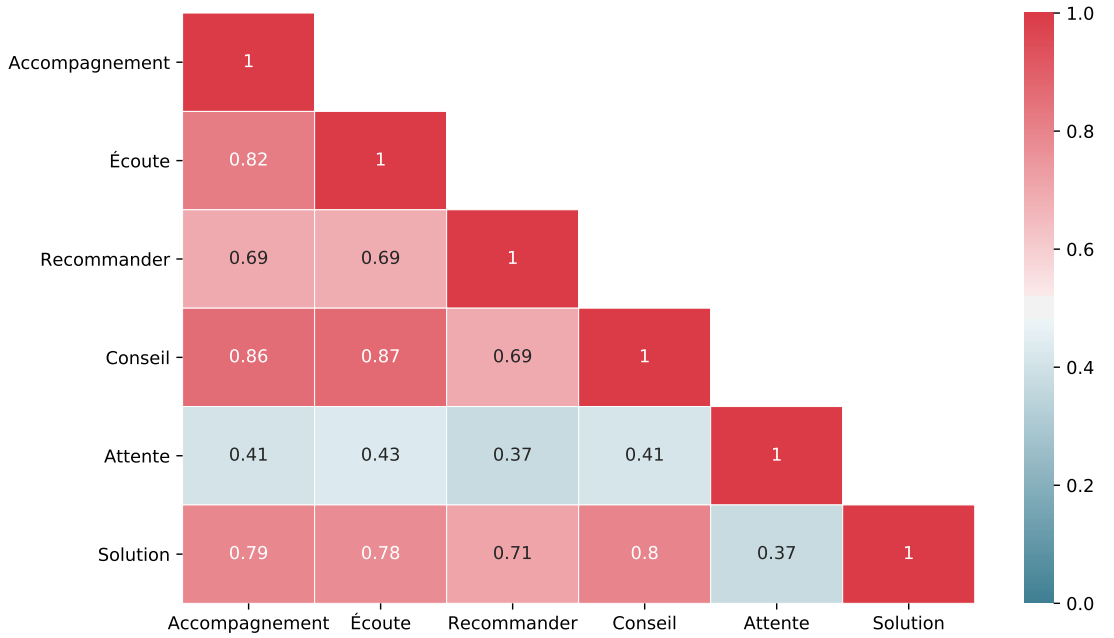


FIGURE 5.1. – Corrélation de Spearman entre les réponses données aux différentes questions

questions hors Attente. Cette corrélation un peu plus faible est probablement due au fait que pour la question Recommander certains clients vont potentiellement juger l'entreprise Orange sur des facteurs extérieurs au dialogue (par exemple, leurs retours d'expérience avec l'entreprise lors d'interventions chez eux). La question Attente est elle beaucoup moins corrélée avec les autres questions en ayant des corrélations de Spearman autour des $0{,}4$.

Ces corrélations semblent donc indiquer que les clients ont tendance à donner une réponse similaire à toutes les questions. Lors de la prédiction de ces questions par un algorithme d'apprentissage, il est donc probable que les prédictions soient semblables pour une même conversation. Il n'y a que le temps d'attente qui est véritablement considéré différemment par les clients, mais ceci est probablement lié au fait que cette question n'est pas directement liée aux interactions ayant lieu durant la conversation, mais plutôt avant celle-ci. Par conséquent, il est probable que la prédiction des réponses à cette question en se fondant uniquement sur le contenu des dialogues sera difficile.

5.2.2. Le *Net Promoter Score*

Parmi les différentes questions du questionnaire, Recommander a une importance particulière pour Orange. En effet, c'est la question utilisée par Orange pour calculer le Net Promoter Score (NPS). Le NPS, introduit par REICHHELD [Rei03], est une métrique utilisée dans le domaine d'analyse de la relation client

permettant de mesurer la loyauté des clients envers une entreprise. Sa valeur peut aller de -100 à 100 et une valeur positive est considérée comme étant un bon score. Afin de calculer ce score, les personnes interrogées sont divisées en trois catégories : les détracteurs, les passifs et les promoteurs. Un *détracteur* est un client ayant donné une réponse entre 0 et 6 compris, un *passif* a donné la note de 7 ou 8 et un *promoteur* 9 ou 10. Le NPS est ensuite calculée à l'aide la formule suivante :

$$\text{NPS} = (\frac{\text{\# promoteurs}}{\text{\# clients}} - \frac{\text{\# détracteurs}}{\text{\# clients}}) \times 100$$

Le score en lui-même n'est pas tellement intéressant pour mes travaux car il est en réalité difficile à interpréter lorsque le score est proche de 0. En effet, lorsqu'il est proche de 0, il n'est pas possible de savoir si cela est dû au fait qu'il y a uniquement des clients passifs, le même nombre de clients promoteurs et détracteurs, ou un mixte des deux.

Toutefois, ce score définit trois classes qui permettent de modéliser la satisfaction des clients et de prendre en compte les différences de jugement entre clients. Par ailleurs, ces trois classes permettent d'évaluer les conversations en s'appuyant sur ce qui se fait généralement dans le domaine des relations clients. Cette catégorisation permet également de simplifier les modèles qui vont manipuler ces données (en évitant d'avoir une échelle de 10 introduisant beaucoup de subjectivité due à l'absence de guide d'annotation).

5.3. La satisfaction client comme support pour des modèles bout en bout

Dans le but d'obtenir des représentations distributionnelles du discours conversationnel, j'envisage de passer par une tâche support afin de construire un modèle bout en bout. Dans mon cas, l'idée serait d'utiliser la prédiction de la satisfaction client comme tâche support.

Il est évident que la satisfaction client n'est pas l'unique aspect à prendre en compte pour prédire le discours conversationnel. En effet, la simple présence de certains mots dans le dialogue est très probablement un indicateur fort de la satisfaction du client. On peut par exemple s'attendre à ce que si certaines formulations de politesse sont manquantes dans le dialogue, alors il y a des chances pour que la conversation se soit mal déroulée et que le client est insatisfait. De même, il est très probable que les clients prennent en compte des facteurs extérieurs aux déroulements des conversations pour répondre aux questions, tels que des retours d'expériences passées avec Orange.

Néanmoins, je fais l'hypothèse que la manière dont est structuré discursivement un dialogue peut être un indicateur intéressant pour identifier le ressenti des clients. En effet, on peut supposer que si une conversation introduit des enjeux dialogiques usuellement rencontrés dans des conversations se déroulant

correctement, alors il y a des chances que celle-ci aussi se déroule bien. Et au contraire, on peut espérer retrouver des patrons de structures discursives communs aux conversations après lesquelles les clients se retrouvent insatisfaits.

Avant de pouvoir considérer la tâche de prédiction de la satisfaction client comme un support pour construire des représentations du discours, on doit tout d'abord déterminer s'il est possible de prédire la satisfaction client en se basant uniquement sur les interactions se trouvant dans une conversation.

La tâche de prédiction de la satisfaction client est un problème qui intéresse beaucoup les entreprises qui ont des services d'assistance en ligne ou des centres d'appels. Deux types de supervisions peuvent être utilisés : de la supervision directe à partir de questionnaires de satisfaction renseignés par les clients à l'issue de leur conversation [Boc+17 ; Luq+17] ou de la supervision indirecte produite par des experts [And+17 ; CSR16 ; Roy+16]. Dans notre cas, les données à disposition proviennent d'une supervision directe, une supervision indirecte nécessiterai une annotation manuelle ce qui n'est pas envisageable dans notre cas.

Néanmoins, BOCKHORST et al. [Boc+17] soulèvent trois problèmes de la supervision directe :

1. Uniquement un sous-ensemble des clients répondent aux questionnaires, pouvant mener à un corpus beaucoup trop petit.
2. Les clients doivent répondre sur une échelle de 5 ou 10 niveaux. La question se pose alors de si on doit considérer chaque niveau comme étant une classe, s'il faut utiliser de la régression ou s'il faut regrouper ensemble plusieurs niveaux afin d'obtenir de nouvelles étiquettes.
3. Le dernier problème concerne la faisabilité de la tâche en elle-même. En effet, les réponses aux questions sont subjectives et sont en lien avec une satisfaction globale allant plus loin que la conversation courante. De ce fait, on peut se demander si les conversations contiennent suffisamment d'indices objectifs pour répondre à cette question subjective.

Le premier point n'est pas un problème dans notre cas. Contrairement aux centres d'appels téléphoniques pour lesquels il est difficile d'accumuler une quantité importante de conversations, dans le cas des tchats on a pu voir dans la section précédente qu'il n'y a pas un manque de données.

Le deuxième point est important mais dépend tout d'abord de la faisabilité de la tâche. Dans un premier temps, je vais considérer le problème comme étant un problème de classification. Je vais toutefois me fonder sur les trois catégories utilisées pour calculer le NPS telles que définies par le domaine de la gestion de la relation client. Une étude plus poussée sur ce point sera développée dans la sous-section 5.3.3 afin de comparer différents schémas de prédiction.

Le dernier point est la motivation principale de cette section : est-il possible de déduire directement à partir du simple contenu des conversations des opinions aussi subjectives que la satisfaction client ? Ce point-là sera développé tout au long de la section, en commençant dans la sous-section 5.3.1 par de premières

approches de classifications permettant de se faire une idée de la difficulté de la tâche. La sous-section 5.3.2 nous permettra alors de voir quelles sont les erreurs réalisées par les différents modèles de classifications.

5.3.1. Estimation de la difficulté de la tâche

Dans le but d'évaluer la faisabilité de la tâche de prédiction de la satisfaction client, je propose de comparer plusieurs algorithmes d'apprentissage en considérant la tâche comme un problème de classification. Étant donné que j'ai à disposition plusieurs questions, je vais considérer que chaque question est une sous-tâche. L'idée est donc d'apprendre un modèle spécialisé pour chaque question. Ceci permettra de constater si les fortes corrélations observées dans la section 5.2 se confirme dans les prédictions des réponses.

Les performances de plusieurs modèles sont comparées. Ceux-ci sont construits à partir des algorithmes d'apprentissage suivants :

— machine à vecteurs de support (SVM) ;

— réseau de neurones convolutifs (CNN) ;

— réseau Long-Short Term Memory (LSTM) ;

— LSTM avec mécanismes d'attention.

Pour chacun de ces modèles, uniquement les mots et le rôle des scripteurs associés (téléconseiller ou client) sont donnés en entrée. En effet, l'objectif est de voir s'il est possible de prédire la satisfaction uniquement à partir du contenu brut des conversations. On ne veut donc pas fausser les résultats en donnant des informations sur le contexte extérieur dans lequel s'est produit le dialogue (est-ce que le client est ancien, chemin parcouru sur le site dOrange avant d'arriver au tchat, etc.).

L'utilisation de ces différents modèles n'a pas pour simple but de déterminer le meilleur algorithme d'apprentissage pour répondre à la tâche. Chacun de ces modèles à une vue très différente sur le dialogue. Ceci permet d'évaluer l'importance de la forme « dialogue » pour répondre à la tâche, c.-à-d. l'ordre d'apparition mots, leurs positions dans la conversation ou de manière plus générale le contexte dans lequel les mots apparaissent. Ces modèles permettent alors d'avoir une première idée de l'intérêt que peut avoir la structure dialogique pour prédire la satisfaction client.

Les différents algorithmes d'apprentissage, ainsi que la manière dont les entrées sont considérées sont décrits dans les quatre paragraphes suivants :

Machine à vecteurs de support Le premier modèle est construit à partir d'un classifieur machine à vecteurs de support (SVM). En entrée, la conversation est un simple sac de mots, l'ordre des mots dans la conversation n'est donc pas conservé. Ceci est le modèle de base que nous considérons. Le but de ce modèle

est d'étudier si le simple fait qu'un mot soit présent ou non dans une conversation est suffisant pour prédire la satisfaction. Un modèle par sous-tâche est créé et l'implémentation des SVM à noyau linéaire de PEDREGOSA et al. [Ped+11] est utilisée.

Réseau de neurones convolutifs Le second modèle est construit à l'aide d'un CNN. Contrairement au SVM, celui-ci prend en entrée la séquence de mots ordonnée telle qu'elle apparait dans le dialogue. Lors de l'apprentissage, le CNN identifiera les n-grammes de mots intéressants pour répondre à la tâche. De ce fait, le CNN permet de prendre en compte un contexte local autour des mots. Afin de conserver un mode de fonctionnement simple, la structure en tour de parole n'est pas strictement reproduite dans l'architecture du réseau, en effet la conversation est vue comme un simple document « à plat ». Cependant, pour tout de même prendre en compte le fait que nous étudions un dialogue, après le dernier mot de chaque tour de parole, un symbole <EOT> (pour *end of turn*) est ajouté. Celui-ci permet de délimiter explicitement les tours de parole. Le but de ce modèle est de voir si avoir du contexte local permet de mieux prédire la satisfaction. Le réseau utilisé est fondé sur celui décrit par KIM [Kim14]. Cette architecture est constituée de plusieurs sous-réseaux convolutifs en parallèle ce qui permet d'utiliser des filtres de tailles différentes — et ainsi avoir des contextes locaux de tailles variables.

Soit $D = w_1 \dots w_n$ une conversation, le réseau est défini de la manière suivante :

$$
\begin{aligned}
x_t &= \text{embedding}(w_t, k) \\
x_{t:t+s} &= \bigoplus_{i=t}^{t+s} x_i \\
c_t(f,h) &= \text{relu}(W_{f,h} x_{t:t+h-1} + b_{f,h}) \\
\hat{c}_{f,h} &= \max\{c_1(f,h), c_2(f,h), \dots, c_{n-h+1}(f,h)\} \\
\hat{c} &= \bigoplus_{h \in H} \bigoplus_{f \in F} \hat{c}_{f,h} \\
p &= \text{softmax}(W_d \hat{c} + b_d)
\end{aligned}
$$

où k est la dimension des plongements de mots, $W_{f,h} \in \mathbb{R}^{hk}$ est un filtre correspondant à une fenêtre de mots de taille $h \in H$, $b_{f,h}$ est un biais associé au filtre, W_d et b_d sont les paramètres de la couche de décision, H est un ensemble de tailles de fenêtre et $F = \{1, \dots, m\}$ avec m l'hyperparamètre du nombre de filtre. Le symbole $\oplus$ est l'opérateur de concaténation entre vecteurs.

Réseau Long-Short Term Memory Ce modèle est construit à l'aide de réseaux recurrents de type LSTM. Comme pour le CNN, l'entrée est la séquence de mots du dialogue. Les séquences sont générées de la même manière que précédemment. Ce modèle permet de prendre explicitement en compte le fait que nous travaillons avec des séquences et permet donc si besoin d'identifier des dépendances lointaines entre les mots si elles existent. Le but de ce modèle est de voir si le contexte général, c.-à-d. la position des mots vis-à-vis des autres mots dans le dialogue, permet de mieux prédire la satisfaction.

Soit $D = w_1 \dots w_n$ un dialogue. Le réseau est défini de la manière suivante :

$$
\begin{aligned}
x_t &= \text{embedding}(w_t) \\
\overrightarrow{h}_t &= \overrightarrow{\text{LSTM}}(x_t, \overrightarrow{h}_{t-1}) \\
\overleftarrow{h}_t &= \overleftarrow{\text{LSTM}}(x_t, \overleftarrow{h}_{t-1}) \\
h_t &= [\overrightarrow{h}_t, \overleftarrow{h}_t] \\
h &= h_n \\
p &= \text{softmax}(W_d h + b_d)
\end{aligned}
$$

où W_d et b_d sont les paramètres de la couche de décision.

LSTM avec mécanisme d'attention Une variante du modèle basé sur des LSTM est également étudiée. Pour ce modèle, un mécanisme d'attention est ajouté à la sortie de la couche de LSTM. Même si en théorie les LSTM sont censés être capables d'identifier des dépendances lointaines et de mettre en avant uniquement les mots importants permettant de répondre à la tâche, en pratique, il a été constaté que ce n'est pas toujours le cas. Le mécanisme d'attention permet au réseau d'avoir dans son architecture un mécanisme ayant pour seul but de déterminer les différentes sorties du LSTM permettant au mieux de répondre à la tâche. De ce fait, il est plus difficile pour le réseau « d'oublier » des mots importants se trouvant en début de séquence.

Soit $D = w_1 \dots w_n$ un dialogue. Le mécanisme d'attention utilisé est le suivant :

$$
\begin{aligned}
u_t &= v^\top \tanh(W_a h_t + b_a) \\
\alpha_t &= \frac{\exp(u_t)}{\sum_{i=1}^{n} \exp(u_i)} \\
\text{attention}(h) &= \sum_{i=1}^{n} \alpha_i h_i
\end{aligned}
$$

où W_a et b_a sont des paramètres de la fonction calculant le score d'attention et v est le vecteur de contexte qui est initialisé aléatoirement.

Le réseau complet est défini de la manière suivante :

$$
\begin{aligned}
x_t &= \text{embedding}(w_t) \\
\overrightarrow{h}_t &= \overrightarrow{\text{LSTM}}(x_t, \overrightarrow{h}_{t-1}) \\
\overleftarrow{h}_t &= \overleftarrow{\text{LSTM}}(x_t, \overleftarrow{h}_{t-1}) \\
h_t &= [\overrightarrow{h}_t, \overleftarrow{h}_t] \\
h &= \{h_t \mid t \in [1,n]\} \\
c &= \text{attention}(h) \\
p &= \text{softmax}(W_d c + b_d)
\end{aligned}
$$

où W_d et b_d sont les paramètres de la couche de décision.

5.3.1.1. Protocole expérimental

Les expériences doivent permettre de répondre, au moins partiellement, aux questions suivantes :

1. Est-ce qu'il est possible de prédire la satisfaction client à partir des seules interactions dans une conversation ?
2. Qu'est-ce qui est utile dans la conversation pour réaliser les prédictions ? Par extension, est-ce que les structures dialogiques de la conversation ont un lien avec la satisfaction client ? Un premier élément de réponse pourra être apporté en exploitant le fait que les différents algorithmes ont une vue différente sur le dialogue (sacs de mots contre séquence de mots).
3. Est-ce que les corrélations observées entre les réponses aux questions se confirment dans les prédictions automatiques ? Est-il nécessaire de considérer toutes les questions dans les expériences ?

Afin d'apporter des réponses à ces questions, les différents algorithmes présentés dans la sous-section précédente sont utilisés pour entrainer des classifieurs. Pour chaque type de classifieur, un modèle est appris par sous-tâche. Les différentes sous-tâches correspondent aux différentes questions présentées dans la table 5.1. Cependant, deux questions sont exclues des expérimentations : `Attente` et `Recontacter`.

En effet, la première n'a qu'un rapport très faible avec la satisfaction du client vis-à-vis de la conversation, celle-ci dépendant non pas de son déroulement mais plutôt de facteurs extérieurs.

La seconde question se concentre principalement sur l'objet de la conversation plutôt que sur les interactions, c.-à-d. qu'en fonction du problème remonté par le client et des éventuelles solutions proposées par le téléconseiller, il sera parfois indispensable de réaliser une nouvelle conversation dépendant de facteurs encore une fois extérieurs. Dans les expériences qui suivent, les sous-tâches seront donc les questions `Accompagnement`, `Écoute`, `Recommander`, `Conseil` et `Solution`.

Lors de la phase d'entrainement, les modèles sont entrainés pendant 100 itérations, cependant, à chaque fois, le modèle qui est conservé correspond à l'itération qui obtient le score de perte le plus faible sur le corpus de développement. Ceci permet d'éviter de conserver un modèle ayant sur-appris sur le corpus d'entrainement.

Pour ces expérimentations, le CNN utilise des filtres de tailles 3, 4 et 5 avec 100 filtres pour chaque taille. Pour le LSTM, avec ou sans attention, les couches cachées ont pour taille 128. Pour les deux réseaux, les plongements de mots sont de dimension 100 et sont appris par le réseau lui-même lors de l'entrainement. La fonction de perte utilisée est la fonction d'entropie croisée. Les poids sont initialisés uniformément dans $[-0{,}1; 0{,}1]$ et optimisés avec l'algorithme ADAM. Un *dropout* de $0{,}5$ est appliqué en sortie de la couche de LSTM.

Un dernier traitement est effectué sur les données pour des raisons techniques. En effet, les différentes conversations peuvent avoir des tailles extrêmement différentes. Pour répondre à ce problème, nous limitons la taille des conversations à 1200 unités lexicales. Si une conversation est plus courte, un symbole de remplissage est ajouté autant de fois que nécessaire pour arriver à 1200 unités lexicales. Dans le cas contraire où une conversation serait trop longue, uniquement une partie de la conversation est conservée. Dans les expérimentations présentées ici, uniquement les derniers mots de la conversation sont conservés. Ceci ne concerne que 4% des conversations.

Pour évaluer les performances des différents modèles, la première métrique considérée est l'exactitude. Celle-ci est définie comme suit, pour une question donnée :

$$\text{exactitude} = \frac{\text{\# réponses correctement prédites}}{\text{\# conversations}}$$

L'avantage de cette métrique est qu'elle est très simple à mettre en œuvre et à interpréter. Une grande lacune de l'exactitude est qu'elle ne permet pas de déterminer si les erreurs sont « graves », c.-à-d. si les réponses prédites sont plus ou moins distantes des réponses attendues sur l'échelle de réponse.

Dans le but de prendre en compte cet aspect-ci, une deuxième métrique correspondant à l'erreur absolue moyenne (*Mean Absolute Error* en anglais) (MAE) est utilisée. Cette métrique est définie comme suit :

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n}$$

où n est le nombre de conversations, y_i est la réponse de référence et $\hat{y}_i$ est la réponse prédite par le modèle. Pour rappel, pour toutes les questions sauf `Recommander`, les réponses correspondent à des notes sur une échelle de 5. Pour `Recommander`, l'ordre est le suivant : `Détracteur` $<$ `Passif` $<$ `Promoteur`.

La MAE est difficile à interpréter sans référentiel, et donc il n'est pas aisé de tirer des conclusions générales sur la qualité d'un système en utilisant cette métrique. Néanmoins, elle reste utile pour comparer la qualité des prédictions produites par les différents modèles. En effet, lorsqu'un modèle obtient un score de MAE plus bas qu'un autre, cela indique que le premier modèle réalise en moyenne des erreurs moins « graves » que le deuxième.

5.3.1.2. Résultats

La table 5.2 présente les résultats obtenus par les différents modèles sur les différentes tâches de prédiction des réponses aux questions de satisfaction. Afin d'avoir un point de comparaison, j'y indique également les scores obtenus si uniquement la classe majoritaire était prédite.

Avant toute analyse fine par question, il est intéressant de constater que les résultats obtenus sont très similaires d'une tâche à l'autre. Ceci est cohérent avec les résultats obtenus par l'étude des corrélations entre questions réalisée dans la section 5.2.1.

La deuxième remarque générale est que tous les algorithmes d'apprentissage obtiennent des résultats assez proches, quelle que soit la question, avec environ 2 points d'exactitude de différence entre le meilleur et le moins bon modèle. De plus, les scores d'exactitude semblent être plutôt bas et relativement proches du score obtenu par l'approche majorité. Ceci montre donc que la tâche est difficile et que les algorithmes semblent avoir du mal à tirer parti du contenu des conversations pour prédire la satisfaction client.

On peut tout de même observer que les approches LSTM avec mécanismes d'attention et CNN permettent d'obtenir les meilleurs scores d'exactitude et de MAE. Même si la différence avec les SVM est très faible (environ 2 points d'exactitude), ces résultats n'invalident pas la possibilité que la connaissance de la structure dialogique puisse avoir une utilité pour prédire la satisfaction client, même si cette contribution semble être marginale. Des expériences additionnelles sont nécessaires pour avoir une réponse plus complète.

Il est intéressant de constater que sur la tâche `Recommander` le LSTM avec attention obtient la meilleure exactitude avec $1{,}26$ points de plus que l'approche CNN mais il obtient un MAE plus important que le CNN. Ceci semble indiquer que même si les LSTM avec attention permettent d'obtenir de meilleures prédictions de manière générale, les erreurs commises sont plus « graves ». Ceci est intriguant étant donné que cette tâche ne comporte que 3 classes, et que donc les erreurs « graves » ont de fortes chances d'être entre les deux classes extrêmes. Cet aspect sera davantage étudié dans la section 5.3.2 afin de déterminer la nature

Question (taille de l'échelle)	Approche	Exactitude (en %)	MAE
Accompagnement (5)	Majorité	48,48	0,974
	SVM	55,28	0,729
	CNN	**56,85**	0,64
	LSTM	55,68	0,644
	LSTM+Attn	56,82	**0,633**
Conseil (5)	Majorité	53,24	0,867
	SVM	59,84	0,613
	CNN	61,22	0,562
	LSTM	60,56	0,569
	LSTM+Attn	**61,43**	**0,556**
Solution (5)	Majorité	44,38	1,195
	SVM	54,62	0,788
	CNN	55,82	0,724
	LSTM	54,12	0,767
	LSTM+Attn	**56,21**	**0,713**
Écoute (5)	Majorité	54,57	0,833
	SVM	61,26	0,613
	CNN	62,77	0,54
	LSTM	61,91	0,554
	LSTM+Attn	**63,10**	**0,532**
Recommander (3)	Majorité	42,71	0,882
	SVM	56,31	0,593
	CNN	**57,51**	**0,562**
	LSTM	56,01	0,605
	LSTM+Attn	**57,53**	0,581

TABLE 5.2. – Résultats des classifieurs pour la prédiction des indicateurs de satisfaction. L'exactitude doit être la plus élevée possible et la MAE doit être idéalement proche de 0.

exacte des erreurs de prédiction.

À partir des résultats obtenus, on peut donc conclure que :

1. La tâche de prédiction de la satisfaction à partir du simple contenu des conversations est difficile. En effet, les différents modèles ne permettent d'obtenir que de faibles améliorations par rapport à la majorité, les résultats restent proches des $50 - 60\%$ d'exactitude. Il est toutefois impossible de dire si cela provient uniquement des données ou si les modèles utilisés sont également fautifs.
2. Le fait de prendre en compte des séquences de mots plutôt que des sacs de

mots en entrée ne semble pas avoir de grande importance pour répondre aux différentes tâches.

Le premier point nécessite donc d'explorer différentes approches afin de déterminer s'il est possible de corriger les erreurs produites par les modèles utilisés précédemment. Cet aspect-là sera exploré dans les deux sous-sections 5.3.2 et 5.3.3 qui suivent.

Le second point met en avant une question très importante concernant la problématique de l'utilisation de la satisfaction client dans le but d'obtenir des représentations du discours. En effet, il se peut que les modèles ne parviennent pas du tout à exploiter les structures complexes et séquentielles en entrées et qu'ils se contentent de déterminer si un mot (ou groupe de mots) apparait dans le dialogue. Dans ce cas de figure, cela impliquerait que l'utilisation d'une approche bout en bout qui reposerait sur la mesure de la qualité des interactions ne serait pas utilisable pour construire des représentations de la structure discursive. Il est donc primordial de réaliser davantage d'expériences pour étudier l'influence de la forme du dialogue dans les performances des modèles utilisés. Ceci sera fait dans la section 5.4.

5.3.2. Étude de la nature des erreurs de prédiction

Dans les premiers résultats obtenus sur la prédiction de la satisfaction, on a pu constater que la tâche est difficile. Pour en savoir plus, il est donc intéressant d'analyser les erreurs commises par les modèles afin de les comprendre et éventuellement d'apporter de solutions.

On a pu voir que les réponses données par un client sont plutôt corrélées et les résultats des prédictions semblent globalement confirmer cela. Afin de ne pas avoir à multiplier les analyses pour rien, dans la suite des travaux, je m'intéresserai uniquement à la question Recommander pour les raisons suivantes :

1. La question Recommander à l'avantage de n'avoir que trois classes, rendant ainsi une première analyse des erreurs plus simple.
2. Par ailleurs, les trois classes sont relativement bien équilibrées, contrairement à toutes les autres questions où une classe sur les 5 correspond à plus de 50% des réponses. La question Recommander permet donc d'éviter de devoir prendre en compte ce déséquilibre lors de l'apprentissage.
3. Cette question a l'intérêt d'être directement liée au calcul du NPS utilisé par les services clients pour analyser la satisfaction générale des clients.

Afin d'avoir une meilleure idée des erreurs commises, j'introduis deux nouvelles métriques : le taux d'erreurs sérieuses (TES) et la MacroF1.

Le taux d'erreurs sérieuses est utilisé afin de prendre en compte le fait que les erreurs entre classes éloignées sont plus graves que des erreurs entre classes proches. Contrairement à la MAE, le TES est construit autour du fait qu'il n'y a

que trois classes dans notre tâche et ne prend en compte que les confusions ayant lieu entre les classes `Promoteur` et `Détracteur`. La métrique est définie comme suit :

$$\text{TES} = \frac{\#\text{ confusions entre Promoteur et Détracteur}}{\#\text{ conversations}}$$

Idéalement, on souhaite donc obtenir un score proche de 0. L'intérêt du score ainsi obtenu est qu'il se focalise sur les erreurs graves. Bien entendu, dans un modèle parfait, il ne faudrait pas de confusions avec la classe `Passif` non plus. Cependant, les données obtenues sont produites directement par les clients sans avoir accès à un guide d'annotation précis, la classe `Passif` est par conséquent davantage exposée à des différences de traitements des notes par les clients. En revanche, on peut raisonnablement faire la supposition qu'une confusion entre `Promoteur` et `Détracteur` aura très peu lieu entre clients et donc les modèles de prédiction ne doivent surtout pas faire ces confusions, ce que permet de mesurer le TES.

La *MacroF1* est une métrique beaucoup plus « classique » construite à partir des scores de *F-mesure* (F1) de chacune des classes. Ceci permet par conséquent de déterminer les performances des modèles sur chacune des classes individuellement. Le score de F-mesure est la moyenne harmonique de deux autres métriques : le rappel et la précision. Celles-ci sont définies comme suit :

$$\text{rappel}(l) = \frac{\#\text{ conversations correctement prédites dans la classe } l}{\#\text{ conversations appartenant à la classe } l}$$

$$\text{précision}(l) = \frac{\#\text{ conversations correctement prédites dans la classe } l}{\#\text{ conversations prédites dans la classe } l}$$

$$\text{F1}(l) = 2 \cdot \frac{\text{rappel}(l) \cdot \text{précision}(l)}{\text{rappel}(l) + \text{précision}(l)}$$

où $l \in \{\texttt{Détracteur}, \texttt{Passif}, \texttt{Promoteur}\}$.

La MacroF1 est elle définie comme suit :

$$\text{MacroF1} = \frac{\text{F1}(\texttt{Détracteur}) + \text{F1}(\texttt{Passif}) + \text{F1}(\texttt{Promoteur})}{3}$$

Contrairement à l'exactitude, pour avoir un score élevé il est nécessaire d'obtenir de bonnes performances sur toutes les classes.

Ces nouvelles métriques sont utilisées afin d'évaluer les différents modèles introduits précédemment dans la section 5.3.1. Les résultats sont présentés dans la table 5.3.

En s'intéressant au TES, on constate qu'il est relativement élevé avec environ 15% des prédictions qui sont des erreurs sérieuses, quel que soit le modèle. Le SVM est le modèle réalisant le moins d'erreurs sérieuses ($14{,}7\%$) en étant environ 1 points plus bas que les deux autres modèles. Sur la MacroF1 on constate un phénomène similaire où le SVM obtient un score de $48{,}3\%$, un score plus élevé

Approche	Exact.	TES	MacroF1	F1 Dét.	F1 Pas.	F1 Pro.
Majorité	42.7	30.9	19.9	–	–	59.9
SVM	56.9	**14.7**	**48.3**	63.5	**14.9**	66.3
CNN	**57.5**	15.5	46.2	**64.4**	7.2	67.0
LSTM+Attn	**57.5**	15.8	44.5	64.3	1.8	**67.3**

TABLE 5.3. – Comparaison des différents modèles de classification sur la tâche Recommander

de 2 points par rapport au CNN et 4 points par rapport au LSTM+Attn.

Ces résultats entrent donc en contradiction avec le score d'exactitude qui favorise plutôt les deux approches neuronales. Cependant, en observant les scores F1 en détail par classe, le phénomène s'explique aisément. En effet, on peut constater que pour le CNN et le LSTM le score F1 sur la classe Passif est très faible en étant respectivement à 7,2% et 1,8%. En effet, il se trouve que cette classe n'est que très peu prédite par les deux modèles. Ceci a deux effets :

1. Le score de MacroF1 pénalise très fortement le fait qu'une classe ne soit pas prédite ;
2. Lorsqu'un modèle prédit la mauvaise classe (par exemple pour des conversations difficiles à classifier), il ne reste donc que la classe extrême opposée. Le TES est alors sans surprise affecté négativement.

Le SVM va quant à lui davantage prédire la classe Passif. Même si cela est légèrement au détriment des deux autres classes, cela permet de limiter les erreurs sérieuses. Cependant, le score F1 de la classe Passif reste très faible et n'améliore donc que très marginalement le TES et la MacroF1.

5.3.3. Limiter les confusions entre classes extrêmes

Nous avons pu constater sur les résultats présentés dans la section précédente que les classifieurs ont énormément du mal à prédire les classes intermédiaires. Ceci a pour effet que lorsqu'une erreur est faite par le classifieur, elle sera de fait entre les classes extrêmes de satisfaction. Ceci est réellement problématique car ce qui nous intéresse dans cette tâche n'est pas seulement d'obtenir le plus haut score possible d'exactitude, mais également qu'il y ait peu de confusions entre les classes Détracteur et Promoteur. Ce dernier aspect est important car s'il existe des confusions entre ces deux classes, cela signifie que le modèle n'est pas capable de réellement discriminer à partir du contenu de la conversation les différences de qualités des interactions entre scripteurs. Il est donc primordial de trouver des schémas de prédictions permettant de prendre en compte ces problématiques.

Trois approches différentes sont explorées pour limiter les confusions. Dans la sous-section 5.3.3.1, j'étudie le problème comme étant un problème de régression en exploitant le fait que les réponses sont originellement sur une échelle entre 0 et 10. Dans la sous-section 5.3.3.2, j'utilise les autres questions comme support additionnel aux classifieurs afin de construire des modèles qui doivent prédire toutes les réponses simultanément. Enfin, dans la sous-section 5.3.3.3, je considère de nouveau la tâche comme un problème de classification, cependant les classes extrêmes sont cette fois-ci vues comme deux tâches binaires indépendantes.

5.3.3.1. Étudier la tâche sous la forme d'un problème de régression

Les confusions entre la classe Promoteur et Détracteur sont probablement dues au fait de considérer la tâche comme un simple problème de classification. En effet, les classifieurs utilisés n'ont aucun moyen de déterminer qu'une erreur entre les deux classes extrêmes est plus grave qu'entre l'une des classes extrêmes et la classe Passif. L'un des moyens les plus simples pour considérer les réponses comme étant des notes est de voir le problème non pas comme un problème de classification mais plutôt comme un problème de régression.

Le fait de modéliser le problème comme une régression permet de directement utiliser la MAE comme fonction de perte à minimiser. Dans cette approche, uniquement un modèle de type CNN a été utilisé, le but étant ici de valider ou pas l'utilisation d'une régression.

Le CNN utilisé est donc le même que pour la classification mis à part le fait que la couche de sortie est une simple couche dense, supprimant ainsi la fonction $\mathrm{softmax}$ utilisée pour la classification.

Dans cette configuration, la tâche cible ne considère plus les 3 classes précédemment étudiées mais directement les 11 notes possibles. Afin d'avoir un point de comparaison, un classifieur de type CNN est également entrainé avec ces 11 notes en considérant chaque note comme une classe.

Afin de pouvoir utiliser les mêmes métriques d'évaluation que précédemment, les prédictions sur une échelle de 11 sont ensuite rangées dans leur classe correspondante sur l'échelle de 3 (c.-à-d. Détracteur si la prédiction est dans $[0;\ 6]$, Passif dans $]6;\ 8]$ et Promoteur dans $]8;\ 10]$). Une fois cette transformation réalisée, les métriques peuvent être utilisées directement.

La table 5.4 présente les résultats sur la question Recommander en utilisant une régression. Ces résultats sont comparés à ceux obtenus en réalisant une classification avec 3 classes et 11 classes. On peut constater que l'utilisation d'une régression permet de grandement améliorer le TES (-8 points par rapport à l'approche par classification) et la MacroF1 ($+6$ points). On constate en effet que la classe Passif est bien mieux prédite en ayant un score F1 de $38{,}6\%$. En revanche, l'utilisation d'une régression a pour effet que l'exactitude perd 5 points. Cette perte se confirme en observant les scores F1 des classes Détracteur et plus

Approche (Échelle)	Exact.	TES	MacroF1	F1 Dét.	F1 Pas.	F1 Pro.
Majorité	42,7	30,9	19,9	–	–	59,9
CNN Class. (3)	57,5	15,5	46,2	64,4	7,2	67,0
CNN Class. (11)	55,8	17,0	43,7	58,9	5,6	66,5
CNN Rég. (11)	52,3	7,6	52,5	62,3	38,6	56,6

TABLE 5.4. – Comparaison d'une approche par classification avec une approche par régression sur la tâche Recommander.

particulièrement Promoteur, cette dernière perdant 10 points et obtenant un score plus faible que la majorité.

On peut également observer que le fait de mieux prédire les notes intermédiaires n'est pas uniquement dû à l'utilisation d'une échelle de 11, mais que c'est bien grâce à la régression. En effet, on peut constater que le CNN en mode classification sur une échelle de 11 ne permet pas de mieux prédire la classe Passif que le CNN utilisant directement une échelle de 3.

Au vu des résultats obtenus, un modèle à base de régression permet de partiellement résoudre le problème de la classe Passif non prédite mais au détriment des performances générales, en particulier sur la classe Promoteur qui est majoritaire. La régression à elle seule n'est donc pas une réponse suffisante à la tâche de prédiction de la satisfaction à partir du seul contenu d'une conversation.

5.3.3.2. Ajouter de la confiance à l'aide des questions additionnelles

Une hypothèse qui peut être faite afin d'expliquer la faible prédiction de la classe Passif, que ce soit en classification ou en régression, est que les conversations qui sont censées être prédites avec cette classe contiennent peu d'informations discriminantes sur lesquels les modèles pourraient se fonder. Par ailleurs, les conversations se trouvant à la limite d'une des deux classes extrêmes pourraient également contenir des traits ressemblant à ceux trouvés dans les classes Détracteur et Promoteur. Ceci a pour conséquence de rendre difficile le fait de différencier des conversations de la classe Passif et de celles des deux autres classes.

Si cela est vrai, alors il faudrait donner un moyen au réseau de s'appuyer davantage sur les traits propres à la classe Passif. Un intérêt des questionnaires de satisfaction à disposition est qu'ils regroupent plusieurs questions en lien avec la satisfaction client. Même si les réponses aux questions sont très corrélées, il existe tout de même de légères variations qui pourraient servir à mettre en avant les traits spécifiques à la classe Passif.

Afin de prendre en compte les autres questions pour aider à la prédiction des réponses à la question Recommander, deux choix sont possibles :

Modèle	**Exact.**	**TES**	**MacroF1**	**F1** Dét.	**F1** Pas.	**F1** Pro.
Approche mono-tâche						
CNN-C	57,5	15,5	46,2	64,4	7,2	67,0
CNN-R	52,3	7,6	52,5	62,3	38,6	56,6
Approche multi-tâches						
CNN-C	57,4	11,8	52,4	64,3	26,3	66,5
CNN-R	53,0	7,2	52,9	61,9	38,0	58,9

TABLE 5.5. – Comparaison d'une approche mono-tâche avec une approche multi-tâche sur la tâche Recommander. CNN-C correspond à un modèle utilisant la classification et CNN-R à un modèle utilisant la régression.

1. Utiliser les réponses aux autres questions comme entrées du réseau ;
2. Prédire les autres réponses en même temps que la réponse à la question Recommander en utilisant un modèle multi-tâches.

La première solution n'est en réalité pas méthodologiquement correcte. En effet, on a pu constater que les différentes questions sont très corrélées entre elles et les réponses apportées par les clients pour une question ne sont clairement pas indépendantes des autres réponses. De plus, cette configuration est très artificielle : il existe très peu de conversations sur lesquelles il y a une réponse apportée à chaque question, hormis Recommander.

De ce fait, uniquement la seconde approche correspond à une solution utilisable en pratique et permet de contribuer à la validation de l'hypothèse formulée en début de section. L'architecture CNN utilisée dans les sections précédentes est donc enrichie afin de permettre le multi-tâches. On obtient alors un modèle où toutes les couches sont partagées sauf les couches de décisions qui sont individuelles à chaque tâche. La fonction de perte est alors la somme des fonctions de perte de chaque tâche. Les différentes tâches sont alors de prédire les réponses aux questions Recommander, Accompagnement, Conseil, Solution et Écoute. Cette architecture est à la fois utilisée en mode régression (CCN-R) en utilisant l'échelle de 11 pour la question Recommander et en mode classification (CNN-C) en utilisant l'échelle de 3.

L'évaluation ne se fait que sur la tâche Recommander en utilisant les mêmes métriques que précédemment.

La table 5.5 présente les résultats obtenus en utilisant une approche multi-tâche. Afin de pouvoir aisément comparer ces résultats avec les approches mono-tâches utilisées jusqu'à présent, ceux-ci sont également reportés dans la table.

Il est intéressant de constater que l'utilisation d'une approche multi-tâches ne produit pas les mêmes comportements lors d'une classification et d'une régression. En effet, avec des classifications (CNN-C), on peut constater un gain très

important ($+19$ points) sur le score F1 de la classe Passif par rapport au même type de modèle en mono-tâche. On observe également une amélioration de la TES (-4 points) et de la MacroF1 ($+6$ points). Par ailleurs, on constate des scores presque équivalents sur l'exactitude et les scores F1 des deux autres classes. En mode classification, l'utilisation du multi-tâches permet donc d'obtenir une véritable amélioration de la qualité des prédictions en réduisant le nombre d'erreurs sérieuses, tout en conservant une exactitude similaire à l'approche mono-tâche.

En utilisant des régressions (CNN-R), on constate qu'avec une approche multi-tâches, celles-ci n'offrent pas les mêmes types de gains. On peut observer des de très légers gains sur l'exactitude ($+0{,}7$), la TES ($-0{,}4$), la MacroF1 ($+0{,}4$) et surtout le score F1 de la classe Promoteur ($+2$ points). Cependant, ce dernier score reste tout de même inférieur au score obtenu par la majorité. De plus, les scores F1 des classes Détracteur et Passif baissent très légèrement. Contrairement à ce qui a pu être constaté en mode classification, en mode régression, il ne semble pas y avoir de réels gains à utiliser une approche multi-tâches.

L'utilisation d'une approche multi-tâches est donc intéressante, mais uniquement dans le cadre d'une classification. Il peut paraitre surprenant que ce ne soit pas bénéfique dans le cas d'une régression mais cela peut potentiellement s'expliquer par le fait qu'une régression permet de mieux modéliser des incertitudes qu'une classification dans le cas de notes.

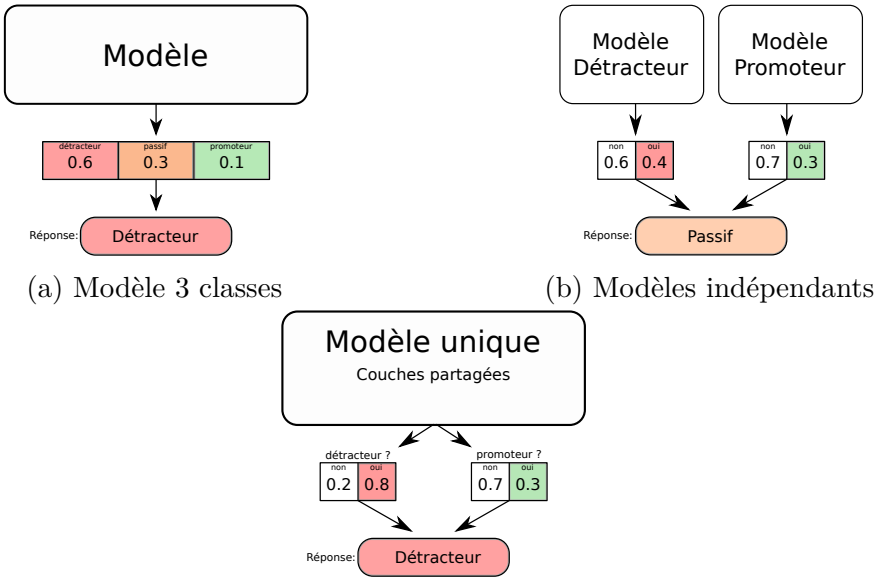
Le fait que sur la classification le multi-tâches permette de réduire les erreurs sérieuses entre autres en augmentant le nombre de prédictions dans la classe Passif semble confirmer l'hypothèse que les modèles mono-tâches ont des difficultés à déterminer des traits intéressants permettant de distinguer les conversations de la classe Passif des deux autres classes. En effet, le fait de donner du contexte supplémentaire en incitant le réseau à trouver des traits intéressants pour d'autres questions semble porter ses fruits.

5.3.3.3. Considérer les classes extrêmes comme des tâches indépendantes

Les deux solutions proposées dans les sous-sections précédentes ont permis de montrer que les modèles ont des difficultés importantes à prédire la classe intermédiaire Passif et qu'il est possible de limiter les erreurs en aidant les modèles à utiliser d'autres traits de la conversation.

Si on s'intéresse en détail à la classe Passif, il est en réalité assez difficile de définir les types de conversations qu'on doit y trouver. Elle correspond aux conversations ayant reçu des notes de 7 ou 8 et donc pour lesquelles les clients ont eu à priori des raisons d'être satisfaits, mais également quelques insatisfactions mineures empêchant de donner les meilleures notes. Cependant, elles peuvent également correspondre à des personnes ayant un avis neutre sur la question. Les conversations dans cette classe peuvent donc être de natures différentes, contrairement aux deux autres classes qui sont plus faciles à définir précisément.

Dès lors, il peut être intéressant de construire un nouveau schéma de classi-



(a) Modèle 3 classes

(b) Modèles indépendants

(c) Modèle partagé

FIGURE 5.2. – Les différents schémas de classification utilisés

fication ne considérant plus les trois classes au même niveau, mais plutôt de considérer que la question principale est de déterminer si le client est un Promoteur ou un Détracteur. La classe Passif serait alors une classe secondaire correspondant aux conversations n'entrant pas dans les deux autres catégories, c.-à-d. une classe de rejet des deux autres classes. Cette modélisation du problème est inspirée de ce qui peut se faire notamment dans le domaine de l'analyse des sentiments [AS12] où les textes peuvent être positifs, négatifs, neutres ou objectifs.

Pour répondre à cette nouvelle définition du problème, je vais construire des modèles spécialisés dans l'identification de conversations de type Détracteur (resp. Promoteur). Ces modèles sont donc de simples classifieurs binaires où les prédictions sont ensuite combinées afin d'obtenir la prédiction finale composée des trois classes :

— Si le modèle Détracteur prédit 1 et le modèle Promoteur 0 alors la prédiction finale sera Détracteur.

— Si le modèle Détracteur prédit 0 et le modèle Promoteur 1 alors la prédiction finale sera Promoteur.

— Si les deux modèles prédisent 0 alors la prédiction finale sera Passif.

— Le dernier cas possible (c.-à-d. 1 et 1) ne s'est jamais produit dans mes diverses expérimentations, mais il y aurait deux possibilités : soit on considère la prédiction finale comme étant Passif, soit on considère la prédiction finale comme étant la classe ayant obtenu la plus grande probabilité dans leur modèle respectif.

Modèle	Schéma	Exact.	TES	MF1	F1-D.	F1-Pa.	F1-Pr.
SVM	3 classes	56,9	14,7	48,3	63,5	14,9	66,3
CNN Mono	3 classes	57,5	15,5	46,2	64,4	7,2	67,0
CNN Multi	3 classes	57,4	11,8	52,4	64,3	26,3	66,5
LSTM+Attn	3 classes	57,5	15,8	44,5	64,3	1,8	67,3
CNN Mono	Régression	52,3	7,6	52,5	62,3	38,6	56,6
SVM	Indépendants	52,7	6,2	52,7	58,4	39,6	60,0
CNN	Indépendants	55,2	7,7	53,8	61,0	36,1	64,2
LSTM+Attn	Indépendants	53,5	6,5	53,0	58,2	39,4	61,3
CNN	Partagé	55,0	7,6	53,5	59,8	36,3	64,4
LSTM+Attn	Partagé	53,5	6,5	52,7	58,0	37,7	62,4

TABLE 5.6. – Comparaison des modèles utilisant des schémas de classification à 3 classes et à 2 classes avec rejet.

Dans mes expériences, plusieurs algorithmes d'apprentissage différents sont comparés. Notre nouveau schéma de classification remplace l'unique tâche de classification par deux tâches de classification binaire. Il est donc possible de résoudre ce problème de deux méthodes différentes. La première consiste à simplement créer des modèles indépendants, c.-à-d. un modèle par tâche. La seconde est d'utiliser un unique modèle et c'est uniquement lors de la prise de décision que les deux tâches sont différenciées, à la manière d'un système multi-tâches. L'aspect multi-tâches a l'avantage de permettre au classifieur de partager des informations permettant de répondre aux deux tâches et donc potentiellement d'améliorer l'apprentissage. Dans notre cas, uniquement les classifieurs neuronaux peuvent être utilisés en mode partagé. Pour le modèle SVM, nous n'aurons donc que la variante à deux modèles indépendants. La figure 5.2 illustre les différents schémas de classification proposés permettant de répondre à la tâche de prédiction de la satisfaction.

La table 5.6 présente les résultats obtenus par les deux schémas de classification proposés précédemment (indépendants et partagé). À des fins de comparaison, les meilleurs résultats obtenus par les différents modèles présentés jusqu'ici sont également inclus, que ce soit en mode classification directement sur les 3 classes ou en mode régression.

En s'intéressant dans un premier temps aux performances obtenues par les deux nouveaux schémas comparées à celles des schémas plus classiques, on constate que les scores d'exactitude sont inférieurs (de 2 à 4 points) à ceux obtenus par les schémas à 3 classes comme ce fut le cas pour le CNN en mode régression, mais en obtenant des scores de TES et de MacroF1 légèrement meilleurs que la régression. En outre, on constate que contrairement à la régression, qui obtient un score F1 sur la classe `Promoteur` inférieur à la majorité, les nouveaux

schémas de classification permettent de limiter cette baisse en obtenant pour les CNN des scores F1 de 64% — seulement trois points de moins que le schéma à 3 classes.

En observant les performances par type d'algorithme d'apprentissage, on constate qu'avec des schémas indépendants ou partagés, les SVM et LSTM se comportent de manières très similaires au CNN en mode régression, c.-à-d. en favorisant davantage la classe `Passif`, au détriment des classes `Détracteur` et `Promoteur`. L'architecture à base de CNN, avec les nouveaux schémas, offre elle une sorte de compromis obtenant pour chacune des métriques des performances moindres que le meilleur modèle de chaque métrique mais en ne favorisant aucune métrique, permettant ainsi d'obtenir des performances intéressantes dans chacune d'entre elles.

Enfin, on peut observer que les schémas de classification indépendants et partagés semble être relativement équivalents. Le partage d'une partie du modèle ne permet donc pas d'améliorer la qualité des prédictions. Cependant, le fait que ce dernier schéma obtienne des performances similaires est tout de même très intéressant lorsque l'on s'inscrit dans le cadre plus général de la construction d'une approche bout en bout afin d'en extraire des représentations de la structure discursive du dialogue. En effet, le schéma de classification partagé s'inscrit entièrement dans ce cadre-là en permettant par exemple l'utilisation des couches cachées communes comme représentations distributionnelles de la conversation.

5.3.4. Conclusion

Dans cette section, j'ai déterminé si la tâche de prédiction de la satisfaction client était une tâche réalisable. Cette étape est indispensable avant d'envisager de l'utiliser pour construire des représentations du discours conversationnel.

Les expériences réalisées permettent de tirer deux conclusions importantes :

1. La tâche est très difficile en ne se basant que sur le contenu des conversations. En effet, il est probable que de nombreux facteurs extérieurs à la conversation sont pris en compte dans la notation. En outre, ces premiers résultats semblent indiquer que ces facteurs extérieurs ne se traduisent pas nécessairement dans les interactions entre clients et téléconseillers durant la conversation. Par ailleurs, la tâche est extrêmement subjective, les notes correspondent à un ressenti « sur le moment » des clients en ne se fondant sur aucun guide d'annotation.
2. Le lexique utilisé par les interlocuteurs semble être le facteur ayant le plus de poids pour prédire la satisfaction client. Ceci est confirmé par le fait qu'un simple modèle à base de sacs de mots obtient des résultats proches de ceux obtenus par des approches prenant les entrées sous formes séquentielles. En l'état actuel des choses, il est difficile de dire si le déroulement du dialogue (et donc le discours conversationnel) est utile pour répondre

à la tâche. Il semblerait qu'un contexte local autour des mots puisse être marginalement utile mais là encore, les résultats ne permettent pas de tirer des conclusions certaines.

Le deuxième point nécessite d'être davantage approfondi. En effet, si la structure du dialogue apparait comme étant non corrélée à la satisfaction client, cela signifierait qu'il sera difficile de se baser sur cette dernière pour induire une structure du discours conversationnel. La section 5.4 va s'intéresser à fournir davantage de réponses sur ce point-là.

5.4. Influence du contenu du dialogue sur les prédictions

Dans les sections précédentes, nous avons pu voir qu'il est possible d'obtenir une idée de la satisfaction client uniquement à partir du contenu d'une conversation, même si celui-ci n'est pas strictement suffisant. Les résultats obtenus en considérant les conversations en entrées de différentes manières nous poussent à croire que seule la présence ou non de certains mots est utilisée pour réaliser les prédictions. Si ceci est vrai, alors il ne serait pas possible d'utiliser la tâche de prédiction de la satisfaction client comme d'un proxy pour construire des représentations du discours conversationnel.

Afin de confirmer ou d'infirmer cette hypothèse, je vais réaliser deux expériences dont l'objectif est de voir si certains tours de parole et mots sont porteurs d'informations pour la prédiction de la satisfaction.

Dans la première expérience, je vais étudier l'apport que peuvent avoir les deux rôles de scripteurs sur le déroulement de la conversation pour la prédiction de la satisfaction. Étant donné que celle-ci ne concerne que le client, il est possible que cela ait une influence sur les parties du dialogue nécessaires pour répondre à la tâche. Cette expérience permettra de déterminer si les interventions des deux locuteurs sont nécessaires pour bien modéliser la satisfaction client.

Pour la deuxième expérience, je vais me concentrer sur l'utilisation faite du lexique par les modèles. En effet, on a pu constater précédemment que la présence ou l'absence de certains mots semblait être un facteur important lors de la prédiction de la satisfaction. Je proposerai donc une expérience afin de déterminer si dans un premier temps cela est bien le cas, et, dans un second temps, quelle taille de vocabulaire est nécessaire pour les modèles.

5.4.1. Influence des scripteurs

Les conversations étudiées ont la particularité d'être constituées d'échanges entre deux scripteurs ayant des rôles asymétriques. Le questionnaire de satisfaction qui est utilisé est uniquement proposé au client. Par ailleurs, on peut

Entrées	**SVM**	**CNN**	**LSTM+Attn**
Tous les tours	52,7	53,5	52,7
Tours du client	51,1	52,3	52,5
Tours du téléconseiller	47,2	48,2	47,6

TABLE 5.7. – Comparaison des différents modèles en utilisant la MacroF1 en fonction de l'entrée utilisée.

s'attendre à ce que le téléconseiller se contente de respecter un protocole fixé par son entreprise.

Une hypothèse pouvant être formulée est que le client est le seul participant du dialogue produisant des tours de parole informatifs et nécessaires pour prédire la satisfaction client. Si tel est le cas, alors cela indiquerait que l'analyse des interactions entre les interlocuteurs n'est pas importante pour la déterminer la satisfaction d'un client.

Une autre hypothèse qui peut être formulée est au contraire de considérer que les tours de parole du téléconseiller peuvent être des indices de la satisfaction, en complément de ceux du client. En effet, on peut s'attendre à ce que si le téléconseiller parvient à apporter des solutions, ou à construire un échange efficace alors le client à davantage de chances d'être satisfait. Au contraire, si le téléconseiller ne répond pas au client, pose des questions superflues et ne permettant pas de produire une solution alors le client aura des chances d'être plutôt insatisfait. Si tel est le cas, alors un certain niveau de compréhension des interactions entre interlocuteurs serait utile pour bien répondre à la tâche.

Afin de donner une première réponse à ces questions, j'ai réalisé une expérience qui consiste à ne considérer que les tours de parole d'un seul scripteur lors de l'apprentissage et de l'évaluation des modèles de prédiction. Dans les expériences qui suivent, je propose d'évaluer les performances d'un modèle par type d'algorithme d'apprentissage utilisés jusqu'ici. Les modèles choisis sont ceux utilisant les schémas de classification basés sur 2 classes avec rejet. Pour le CNN et le LSTM avec attention, j'utilise l'approche à un seul modèle partagé car cela correspond au modèle qui pourrait être utilisé en pratique dans une approche bout en bout. Pour le SVM, j'utilise l'approche à deux modèles indépendants étant donné que c'est la seule disponible avec ce schéma-ci.

Pour l'évaluation, je vais uniquement utiliser la métrique MacroF1. L'objectif n'est ici pas d'évaluer la qualité des prédictions, mais plutôt d'avoir une bonne synthèse du comportement des modèles, ce que permet la MacroF1. En effet, cette métrique permet de partiellement évaluer les performances globales tout en permettant de constater si les classes sont prédites de manière équilibrée ou non.

La table 5.7 présente les résultats de cette expérience. Dans un premier temps, on observe que les trois algorithmes obtiennent des performances comparables

même lorsqu'on limite les entrées à celles d'un seul scripteur. Ceci correspond à ce qui a été constaté jusqu'ici. En s'intéressant désormais aux entrées, on constate de manière attendue que les performances les plus élevées sont obtenues lorsque l'ensemble des tours de parole sont pris en compte. Cependant, les performances ne s'effondrent pas pour autant lorsqu'un seul des scripteurs est disponible avec une perte d'environ 5 points dans le pire des cas. Les tours du client sont tout de même les plus informatifs puisqu'ils permettent d'atteindre des performances proches (1 point de différence) de celles obtenues lorsque tous les tours sont disponibles.

Le fait qu'il y ait tout de même un léger gain en ajoutant les tours du téléconseiller semble indiquer que certaines de ses interventions peuvent avoir une influence sur la satisfaction du client. Ceci n'invalide donc pas le fait que la modélisation des interactions entre locuteurs puisse être nécessaire pour correctement prédire la satisfaction client. Néanmoins, cet apport resterait mineur au vu des performances des modèles qui ne considèrent que les clients.

5.4.2. Étude de l'importance du lexique

En manipulant les différents algorithmes d'apprentissage, nous avons constaté que la forme des entrées semble avoir peu d'influence sur les performances. En effet, le fait d'utiliser un simple sac de mots ou une séquence de mots a peu de conséquences sur les prédictions. Le fait qu'un sac de mots en entrée permette d'obtenir des performances aussi proches que des séquences de mots en entrée pousse à croire que les liens discursifs entre les messages sont peu utiles. Au contraire, les prédictions semblent davantage se fonder sur la présence ou l'absence de certains mots dans les dialogues.

Afin d'évaluer l'influence précise du lexique, mais aussi de déterminer quels mots du lexique sont utiles, je propose de réaliser une expérience où le but est de supprimer certains mots du lexique en fonction du nombre d'occurrences des mots dans le corpus DatchaSat. Le comportement attendu est que plus le lexique est faible, plus les performances des modèles sont dégradées.

Dans cette expérience, j'utilise les mêmes modèles et la même métrique que dans la section précédente. Pour les besoins de l'expérimentation, six lexiques sont définis, allant de LEX1 à LEX6 [2], chacun ayant une taille différente. Pour une configuration avec un lexique donné LEX, tous les mots qui n'apparaissent pas dans le corpus sont supprimés en amont de l'entrainement et de l'évaluation. Dans le cas où la suppression d'un mot entrainerait la disparition d'un tour, le symbole <EMPTY> est inséré à la place afin de conserver l'information qu'il y avait un tour. Les lexiques sont construits en ne conservant que les mots apparaissant un nombre suffisant de fois dans le corpus d'entrainement de DatchaSat et sont les suivants :

2. Les lexiques LEX5 et LEX6 sont disponibles en annexes B et C pour les plus curieux.

Lexique	Nombre d'occurrences min.	Taille du lexique
all	0	119 165
lex1	10 000	308
lex2	20 000	173
lex3	30 000	128
lex4	40 000	108
lex5	50 000	92
lex6	100 000	42

La figure 5.3 présente les résultats de cette expérience. Le SVM se comporte comme attendu, c.-à-d. plus le lexique est faible, plus la MacroF1 décroit. Cependant, on peut tout de même constater qu'avec uniquement 308 mots, les performances du modèle restent plutôt correctes en étant proches des 50%. Ceci est remarquable pour un modèle n'utilisant que des sacs de mots et qui est donc très sensible à la modification du lexique.

Les deux autres modèles ont un comportement qui est plus intriguant. En effet, on peut observer que les performances ne baissent que très légèrement, voire pas du tout, jusqu'à LEX5. Une baisse importante n'est réellement constatée qu'avec LEX6 qui ne contient que 42 mots. En observant le contenu des lexiques LEX5 et LEX6, on constate qu'une grande partie des mots ayant disparu d'un lexique à l'autre sont des mots appartenant à des classes ouvertes, ne laissant dans LEX6 presque que des mots de classes fermées. De ce fait, il n'est pas surprenant que la baisse soit davantage marquée en ayant uniquement accès au dernier lexique.

La robustesse des modèles est remarquable car cela signifie que très peu de mots sont réellement nécessaires pour prédire la satisfaction. En revanche, on constate également que la présence de tous les mots est importante, sinon le modèle SVM obtiendrait des performances semblables. Les deux modèles neuronaux montrent que la prise en compte des entrées sous forme séquentielle est intéressante. On peut supposer que le fait de prendre en compte en entrée des séquences permet de retrouver certains mots disparus en utilisant le contexte local dans lequel les mots étaient produits. Étant donné que le CNN obtient des performances semblables au LSTM, on peut supposer le contexte global dans la conversation est peu utile (c.-à-d. qu'il n'y a pas de dépendances lointaines). Le CNN ne pouvant prendre en compte que des contextes locaux [3], contrairement au LSTM avec attention, cela donne un indice supplémentaire pour minimiser l'importance de la prise en compte des interactions entre locuteurs dans les modèles de prédiction de la satisfaction client.

3. Même s'ils peuvent devenir un peu moins locaux lors de la suppression de mots, les fenêtres sont plutôt petites et un grand nombre de mots de classes fermées ne sont pas supprimés.

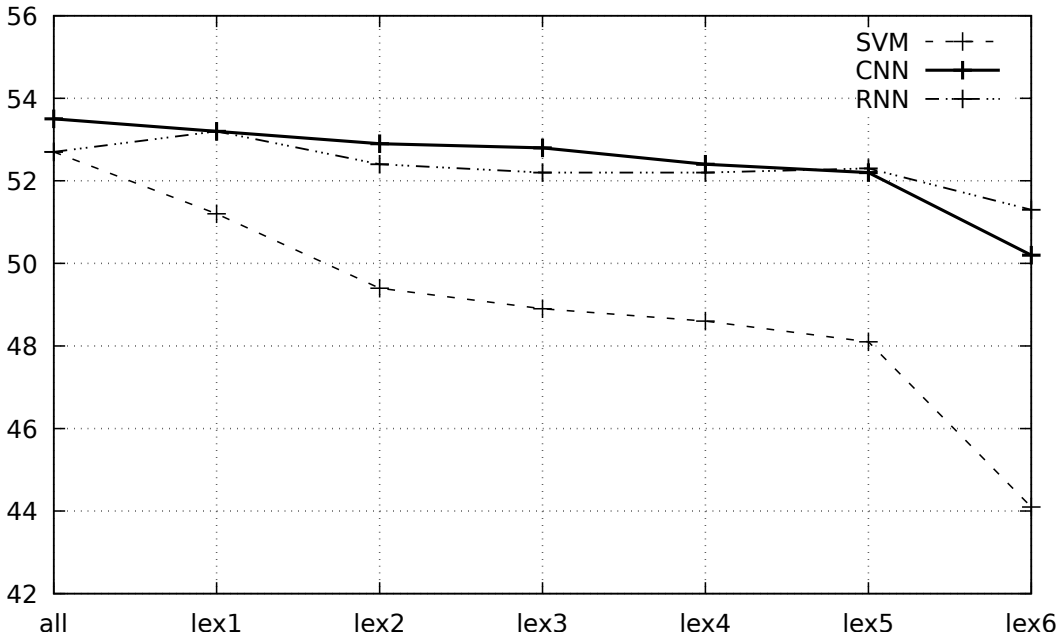


FIGURE 5.3. – Performances (MacroF1) des SVM, CNN et LSTM en fonction de la taille du lexique.

5.5. Conclusion

Dans ce chapitre, l'objectif était de trouver une solution permettant de construire des représentations du discours conversationnel en se fondant sur des ressources ne nécessitant pas d'annotations additionnelles. L'approche considérée a été d'exploiter le fait qu'à l'issue des conversations, les clients sont invités à répondre à un questionnaire de satisfaction. Les réponses à ce questionnaire peuvent alors être utilisées comme tâche support pour construire un modèle bout en bout qui permet de produire des représentations distributionnelles prenant en compte le discours conversationnel. En effet, j'ai fait l'hypothèse que, même si ce n'est pas le seul facteur, les interactions ayant lieu entre les locuteurs ont un lien important avec la satisfaction du client. De ce fait, il serait possible de déduire de manière implicite la structure discursive des dialogues.

Afin de déterminer si cette hypothèse est vraie, j'ai dans un premier temps réalisé des expériences pour estimer la difficulté de la tâche de prédiction de la satisfaction client en utilisant uniquement le contenu de la conversation. Les résultats obtenus indiquent que celle-ci est difficile, les résultats dépassant difficilement les 50% d'exactitude. Ceci est probablement dû au fait que les clients prennent en compte des facteurs extérieurs à la conversation, tels que leurs expériences personnelles avec Orange. Par ailleurs, les réponses à ces questionnaires de satisfaction sont très subjectives. On aurait pu espérer que les facteurs extérieurs puissent être retrouvés dans la manière dont se déroulent les conversations, mais

cela ne semble pas être le cas, ou tout du moins ce n'est pas évident à exploiter.

Les résultats mettent également en avant le fait que les mots sont un facteur disponible extrêmement important pour répondre à la tâche. D'autres résultats laissent penser que les interactions entre locuteurs ont une influence limitée sur la satisfaction client, cette influence se réduisant probablement aux choix réalisés par les scripteurs dans leur vocabulaire en fonction des tours précédents. Il reste tout de même difficile de déterminer l'apport exact de tous les facteurs étant donné que les résultats sur la tâche sont relativement mauvais.

Bien que ces diverses constatations n'invalident pas totalement l'hypothèse qu'il serait possible d'inférer une structure discursive à l'aide de la satisfaction client, il parait déraisonnable de continuer dans cette voie pour notre problématique de réaliser des analyses du discours dans le cadre de communications médiées par ordinateur. En effet, un des objectifs de ma thèse est d'explorer des approches permettant de réaliser des analyses de discours nécessitant peu de supervisions issues d'annotations complexes. Or, le lexique est un très gros attracteur et il est donc difficile d'estimer l'apport concret des interactions sur la satisfaction client.

Une solution serait d'avoir des informations au niveau des tours de parole ou sous-dialogues mettant en avant explicitement les échanges entre locuteurs ayant eu une influence sur la satisfaction du client. Ceci est problématique pour deux raisons :

— cela nécessiterait une annotation manuelle d'un grand nombre de conversations, nécessitant ainsi une définition précise des interactions pouvant avoir un effet. Ceci revient à partiellement réaliser une analyse profonde du discours mais en s'imposant une contrainte supplémentaire de lien avec la satisfaction du client.

— la tâche en elle-même parait très subjective, le client de la conversation étant probablement le seul annotateur pouvant correctement répondre à la question. Il n'est pas envisageable de demander un tel niveau d'annotation à tous les clients.

Une autre solution pourrait être de s'appuyer non pas sur le point de vue du client mais sur celui de l'entreprise pour évaluer la qualité du déroulement des dialogues. En effet, dans le domaine de l'assistance clientèle, deux modes d'évaluation existent pour juger la qualité des interactions : les supervisions directes qui correspondent aux questionnaires de satisfaction client présents dans le corpus DATCHA, et les supervisions indirectes produites par des experts à partir du contenu des conversations. L'intérêt de la première approche provient principalement du fait qu'elle est facile à mettre en place et permet d'avoir directement l'avis des clients. Toutefois, ces avis sont très subjectifs et prennent en compte de nombreux facteurs extérieurs n'ayant pas de liens avec le déroulement du dialogue.

À l'inverse, une supervision indirecte permet d'avoir une évaluation objective

fondée sur des critères précis. En particulier, il serait possible d'avoir un jugement objectif sur la qualité des interactions entre client et téléconseiller. Malheureusement, de telles évaluations nécessitent de produire une liste précise des critères à prendre en compte (c.-à-d. un guide d'annotation) et d'ensuite demander à des annotateurs de réaliser ces évaluations. Il ne serait donc pas possible de simplement s'appuyer sur les données brutes disponibles « gratuitement » avec le corpus.

Étant donné qu'il ne parait pas simple de se fonder uniquement sur des approches bout en bout afin de caractériser les interactions entre locuteurs, je vais dans la partie suivante de la thèse me concentrer sur des approches se reposant davantage sur des annotations en lien direct avec le discours conversationnel.

Troisième partie

Étudier les interactions à partir de représentations explicites du discours

Chapitre 6.

Représentation vectorielle des tours de parole

Sommaire

6.1 Introduction 143
6.2 Étude de la qualité des plongements de phrases sur des tâches de discours conversationnel 144
6.2.1 Les plongements de phrases et les dialogues 145
6.2.2 Entrainement de plongements de phrases 146
6.2.3 Évaluation des plongements de phrases 147
6.3 Prendre en compte explicitement les interactions dans les représentations 151
6.3.1 Apprentissage de plongements de phrases reposant sur les actes de dialogue 152
6.3.2 Évaluation des plongements de phrases 155
6.4 Conclusion 163

6.1. Introduction

Le chapitre 5 a montré qu'il est difficile d'extraire des représentations du discours conversationnel à partir d'approches bout en bout. En effet, ces approches nécessitent d'utiliser des tâches où le lien avec le discours est bien défini, ce qui n'est pas le cas de la prédiction de la satisfaction client. Afin de ne pas se confronter de nouveau à des problèmes de subjectivité liés à la satisfaction client, je vais me concentrer sur des méthodes me permettant de déterminer explicitement ce que doivent contenir les représentations construites afin de caractériser les interactions entre locuteurs.

Outre les questionnaires de satisfaction, le corpus DATCHA contient également — en bien moins grande quantité — des annotations en lien avec le discours conversationnel sous la forme d'actes de dialogue. Ceux-ci correspondent au discours conversationnel de surface et donnent les fonctions de communication de

chaque énoncé. Ils peuvent donc être utilisés dans des tâches d'évaluation afin de déterminer si une partie du discours conversationnel est correctement prise en compte.

Dans DATCHA, les actes de dialogue portent sur les tours de parole. De ce fait, je vais chercher à construire des représentations distributionnelles portant sur les tours de parole et leurs contextes de production. Un tour de parole correspond généralement à une phrase, ce qui permet l'utilisation de nombreuses approches déjà utilisées pour créer des représentations distributionnelles de phrases.

Comme on a pu le voir dans le chapitre Apprentissage de représentations il existe de nombreuses approches permettant de construire et d'évaluer des plongements de phrases. Toutefois, les évaluations prennent peu en compte la nature des textes utilisés, et les plongements sont généralement appris sur des corpus généralistes (et donc pas des dialogues). En outre, les évaluations qui prennent en compte une partie du contexte de production restent très incomplètes (elles se limitent généralement à de l'inférence ou à des associations légende-image) et ne prennent pas en compte explicitement le discours, en particulier conversationnel. Or, il serait intéressant d'étudier le comportement des plongements de phrases dans des contextes dialogiques.

Dans ce chapitre je vais donc étudier les représentations issues du comportement d'algorithmes de création de plongements de phrases sur les dialogues. Je vais définir un cadre d'évaluation se fondant sur les actes de dialogue afin de déterminer si les modèles de plongements de phrases modélisent bien les interactions entre scripteurs. Ce cadre d'évaluation sera alors utilisé sur des modèles de plongements produits à l'aide d'algorithmes existants. Ceci permettra de déterminer si les algorithmes de production usuellement utilisés sont adaptés aux dialogues et permettent de représenter le discours conversationnel. Ce travail sera présenté dans la section 6.2.

Par ailleurs, je vais également proposer de nouveaux plongements de phrases spécifiquement construits afin de prendre explicitement en compte le discours conversationnel. Ces représentations distributionnelles, en les comparant aux plongements de phrases existants à l'aide du nouveau cadre d'évaluation, permettront de juger si les algorithmes de productions usuels permettent de capturer implicitement les interactions entre locuteurs. Dans le cas contraire, ces nouvelles représentations vectorielles des tours de parole pourraient alors être utilisées dans le cadre de tâches se fondant sur le discours. Ces nouveaux plongements seront présentés et évalués dans la section 6.3.

6.2. Étude de la qualité des plongements de phrases sur des tâches de discours conversationnel

Une particularité des dialogues par rapport aux monologues est que les phrases sont produites dans le but de communiquer explicitement des intentions entre

interlocuteurs. De ce fait, le contexte de production des phrases dans le dialogue dépend des intentions des interlocuteurs et des interactions entre eux. Très peu de cadres d'évaluation prennent en compte le fait qu'on peut avoir des interactions entre plusieurs personnes et il ne parait par évident que les algorithmes de plongements de phrases usuellement utilisés permettent de prendre cela en compte. En effet, dans ces derniers la dimension interactive des dialogues n'est jamais explicitée.

Dans cette section, je vais donc étudier comment se comportent certains modèles de plongements de phrases fréquemment utilisés dans le contexte de l'analyse du discours conversationnel. Dans la sous-section 6.2.1, je présente le problème de l'évaluation des plongements de phrases dans le cadre conversationnel, et une solution que je propose au problème. Dans la sous-section 6.2.2, j'introduis les algorithmes existants de plongements de phrases que j'utilise pour entrainer des modèles de plongements sur le corpus DATCHA. Enfin, dans la sous-section 6.2.3, j'évalue ceux-ci afin de déterminer s'ils sont adaptés à l'analyse du discours conversationnel.

6.2.1. Les plongements de phrases et les dialogues

Les plongements de phrases et les tâches d'évaluation sont généralement construits dans le contexte de textes journalistiques, romanesques ou au mieux informels (par exemple des *tweets*). Très peu de travaux ont étudié les plongements de phrases dans le contexte précis des dialogues.

Les dialogues sont organisés très différemment des autres types de documents. On y trouve plusieurs participants qui utilisent potentiellement des registres de langues et des styles d'écriture différents et il peut également y avoir beaucoup de sous-dialogues évoquant des sujets différents et qui s'entremêlent. Ceci nécessite de prendre en compte le discours conversationnel afin d'obtenir le bon contexte autour de la production d'une phrase.

PRAGST et al. [Pra+18] s'intéressent au problème des plongements de phrases dans les dialogues. Pour cela, ils étudient le comportement d'une version modifiée de WORD2VEC permettant d'apprendre des plongements de phrases dans ce contexte des dialogues. Ils évaluent les représentations en comparant des groupements (*clusters* en anglais) calculés à partir des plongements de phrases avec des actes de dialogue manuellement assignés à chaque groupement. La question est de savoir si les phrases d'un même groupement partagent les mêmes actes de dialogues. Ces travaux permettent de visualiser si les plongements appris sont capables de bien modéliser les informations liées au contexte dialogique. Toutefois, ces plongements ne sont pas directement utilisés dans une tâche liée au discours conversationnel alors que ceci permettrait de directement constater si les informations présentes dans les plongements sont effectivement suffisantes.

Afin de savoir si les algorithmes existants de création de plongements de phrases permettent de bien prendre en compte le discours conversationnel, je

propose d'utiliser deux tâches d'évaluation fondées sur les actes de dialogue. Les actes de dialogue sont le premier niveau d'analyse du discours conversationnel. De ce fait, il est indispensable que les modèles de plongements de phrases soient capables de les prendre en compte correctement pour que ceux-ci soient capables de caractériser le discours conversationnel dans son ensemble. En outre, les actes de dialogue ont été très étudiés et constituent donc une base solide de travail. Les deux tâches d'évaluation que je propose sont les suivantes :

Prédiction de l'acte courant La première tâche est simplement de prédire l'acte de dialogue associé à un tour de parole donné mais uniquement à partir du plongement de phrases du tour en question, ainsi que des plongements de phrase des tours précédents. Cette tâche est plutôt simple car de manière générale, l'acte de dialogue pour un tour donné est extrêmement lié au contenu du tour en lui-même. La tâche a donc pour objectif de vérifier que les plongements de phrases contiennent bien les informations lexicales nécessaires pour pouvoir retrouver les actes de dialogue correspondants.

Prédiction de l'acte suivant La deuxième tâche est elle davantage liée au discours conversationnel. Pour cette tâche-ci, nous considérons que le discours conversationnel est représenté par une séquence d'actes de dialogue. La tâche est, pour un tour de parole donné dans la conversation, de prédire l'acte de dialogue du tour de parole suivant sans avoir accès à son plongement correspondant mais uniquement aux plongements précédents. Afin de pouvoir correctement répondre à cette question, les plongements de phrases doivent nécessairement être capables de capturer des informations liées au contexte dans lequel le tour associé est produit.

6.2.2. Entrainement de plongements de phrases

L'apprentissage de plongements de phrases en utilisant un corpus de dialogues est indispensable pour pouvoir juger si les algorithmes de production permettent la création de modèles prenant en compte les interactions dialogiques. Pour ce faire, nous utilisons en entrée de ces algorithmes existants le sous-ensemble d'apprentissage du corpus DatchaSat. Certaines conversations de DatchaAct se trouvant également dans DatchaSat, ces conversations sont supprimées de ce dernier. Le sous-ensemble d'apprentissage de ce corpus contient $47\,685$ dialogues correspondant à $2\,034\,751$ tours de parole. Par ailleurs, ce corpus est intéressant car il couvre tous les services d'assistance clientèle d'Orange, ce qui permet d'avoir une plus grande variété dans les tours de parole étudiés.

Un inconvénient des données Datcha est qu'elles contiennent énormément d'erreurs orthographiques, comme on a pu le constater dans le chapitre 4. Ces erreurs auront pour effet de créer un grand nombre de mots inconnus, ce qui peut

être problématique lors de la création de plongements de mots et de phrases. Bien que le corpus DATCHASAT soit très volumineux et contient probablement la majorité des erreurs fréquemment rencontrées, cela ne couvre pas l'intégralité des erreurs possibles. Afin d'être capable de produire une représentation pour tous les mots, j'utilise l'algorithme d'encodage par paires d'octets (*Byte Pair Encoding* en anglais) (BPE) afin de prétraiter tous les corpus manipulés. Cet algorithme permet de remplacer tous les mots par des séquences de sous-mots. Les plongements sont alors appris sur les sous-mots et non les mots, ce qui permet de produire des représentations même pour les mots mal orthographiés.

Deux algorithmes différents sont utilisés pour apprendre les plongements de phrases. Le premier algorithme crée des plongements en ne prenant en compte que les mots du tour de parole sans prise en compte des autres tours. Cela est fait à l'aide d'une simple moyenne des plongements de mots de la phrase. Les plongements de mots sont eux-mêmes créés en utilisant l'algorithme FASTTEXT [Boj+17] entrainés sur DATCHASAT. Ce modèle de plongements de phrases joue le rôle de base de référence. En effet, on peut s'attendre à ce que ces plongements prennent très peu en compte le discours conversationnel étant donné qu'aucune information sur le contexte de production dialogique n'est utilisée pour la création des plongements.

Pour le deuxième algorithme, le but est de faire en sorte que le contexte autour du tour de parole soit également pris en compte. Pour cela, l'algorithme SKIP-THOUGHT [Kir+15] est utilisé. Les plongements de phrases issus de cet algorithme sont appris afin d'être capable de prédire les tours de paroles précédent et suivant. De ce fait, ces plongements de phrases sont censés, pour une phrase donnée, capturer les informations sur le contexte dans lequel elle est produite. Ceci est comparable au fonctionnement de SKIP-GRAM dans WORD2VEC [Mik+13] qui essaie de prédire les mots autour du mot cible lors de l'apprentissage. Ce modèle de plongements de phrases me permet de déterminer si le discours conversationnel est bien modélisé lorsque le contexte de production immédiat (la prédiction des tours précédent et suivant) est pris en compte.

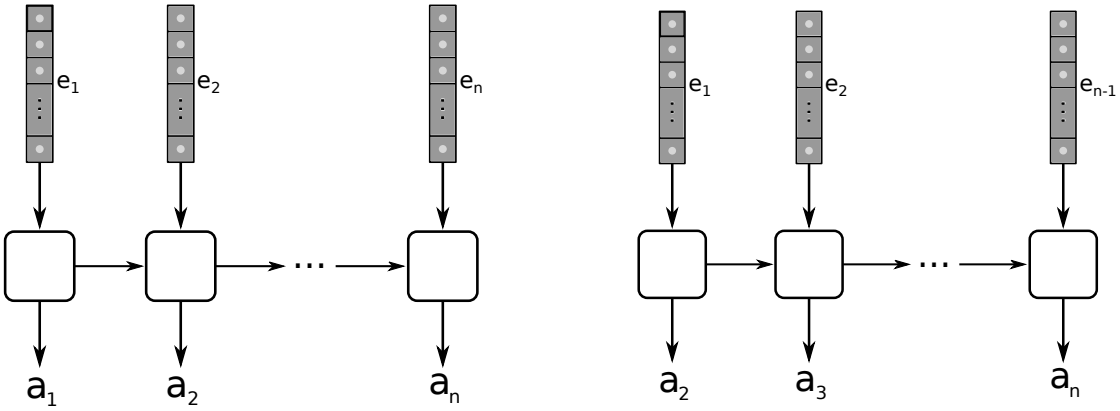
6.2.3. Évaluation des plongements de phrases

Les cadres d'évaluation usuellement utilisés pour déterminer si des plongements de phrases sont de bonnes qualités ne permettent pas d'explicitement prendre en compte les interactions dans les dialogues. De ce fait, il est impossible de savoir si ces plongements capturent des informations sur le contexte dialogique dans lequel les phrases sont produites.

Afin d'évaluer de manière explicite la qualité de la modélisation du discours conversationnel par les plongements de phrases, je propose deux tâches d'évaluation relativement simples se fondant sur la prédiction d'actes de dialogue. Comme indiqué dans la sous-section 6.2.1, les deux tâches sont la prédiction de l'acte de dialogue courant et la prédiction de l'acte de dialogue suivant.

6.2.3.1. Protocole expérimental

Pour les deux tâches d'évaluation, j'utilise des réseaux de neurones récurrents de type LSTM. En effet, ceux-ci sont adaptés aux dialogues qui sont des séquences de plongements de phrases où l'ordre est très important. De plus, à cause des phénomènes d'enchevêtrements, le modèle peut avoir besoin de tours de parole qui ne sont pas directement adjacents pour réaliser les prédictions d'actes de dialogue. Étant donné que la production d'un tour de parole ne dépend pratiquement que des tours passés (les scripteurs réagissent aux tours précédents et n'ont pas de vision sur le futur), j'utilise un réseau récurrent unidirectionnel. Ceci a également l'intérêt de simplifier le modèle et de donner davantage d'importance à la qualité des plongements de phrase (le modèle étant alors peu capable de retrouver les liens dialogiques manquants s'ils ne sont pas présents dans les plongements).



(a) Prédiction de l'acte courant

(b) Prédiction de l'acte suivant

FIGURE 6.1. – Schémas des RNN utilisés pour l'évaluation des plongements de phrases

La figure 6.1 présente schématiquement les réseaux utilisés pour la prédiction de l'acte de dialogue courant (6.1a) et pour la prédiction de l'acte de dialogue suivant (6.1b). Soit $D = t_1 t_2 \dots t_n$ une conversation composée de n tours de parole $t_i = w_{i,1} w_{i,2} \dots w_{i,m}$ où $w_{i,j}$ est un mot et m est la longueur du tour. Un vecteur e_i est le plongement de phrases du tour t_i. Ce vecteur est fixé et n'est donc pas mis à jour lors de l'entrainement du réseau. Plus formellement, le réseau prédisant l'acte courant est défini de la manière suivante :

$$
\begin{aligned}
e_i &= \text{embedding}(t_i) \\
h_i &= \text{LSTM}(e_t, h_{i-1}) \\
p_i &= \text{softmax}(W_d h_i + b_d)
\end{aligned}
$$

où W_d et b_d sont les paramètres de la couche de décision.

On obtient donc en sortie une distribution de probabilité sur les différents actes de dialogue possibles. La fonction objectif est l'entropie croisée et l'algorithme de descente du gradient utilisé est ADAM avec un taux d'apprentissage de $0{,}001$. La couche caché du LSTM est de taille 512.

Le corpus utilisé pour évaluer les plongements de phrases est DatchaAct. Le sous-ensemble d'apprentissage est utilisé pour apprendre le modèle, et j'utilise le sous-ensemble de développement afin de choisir les meilleurs paramètres. Le sous-ensemble d'évaluation est utilisé pour évaluer la qualité des prédictions produites par les étiqueteurs utilisant les différents plongements de phrases.

Afin d'évaluer la qualité des différents modèles de plongements de mots, deux métriques sont utilisées : l'exactitude et la MacroF1. L'exactitude me permet d'évaluer les performances brutes des étiqueteurs. La MacroF1 me permet d'avoir une interprétation plus fine de l'exactitude. En effet, certaines étiquettes apparaissent plus de fois que d'autres et la MacroF1 permet alors de constater si les modèles de prédictions se contentent de prédire uniquement les classes très fréquentes ou s'ils réalisent des prédictions pour toutes les étiquettes.

Sur la tâche de prédiction de l'acte courant, on peut s'attendre à ce que les scores d'exactitude et de MacroF1 soient relativement bons. Comme indiqué dans la section 3.5 du chapitre 3, un CRF ayant accès à d'autres caractéristiques que les plongements de phrases obtient un score d'exactitude de 86%. Il est probable que les seuls plongements de phrases ne soient pas suffisants pour atteindre un même score — ceux-ci n'étant pas produits dans le but de réaliser de la prédiction d'actes de dialogue. On peut donc s'attendre à avoir des performances légèrement inférieures au CRF. Plus un modèle de plongement permettra d'atteindre un score proche des 86%, plus ces plongements modéliseront bien les caractéristiques nécessaires pour prédire les actes de dialogue.

La tâche de prédiction de l'acte suivant est sensiblement plus difficile. En effet, cette tâche revient à être capable de prédire en partie le tour de parole suivant (en se limitant à l'acte de dialogue dans notre cas). Les résultats sur cette tâche doivent permettre de mettre en évidence si certains modèles de plongements de phrases modélisent mieux que d'autres le discours conversationnel.

Outre les performances obtenues en utilisant les deux types de plongements de phrases présentés dans la section 6.2.2, les performances obtenues par des plongements initialisés aléatoirement et mis à jour lors de l'apprentissage des étiqueteurs sont également présentées. Les plongements ainsi créés sont donc entièrement spécialisés pour la tâche d'étiquetage en actes de dialogue (courant ou suivant). Cependant, pour apprendre des plongements de qualité, il est généralement nécessaire d'avoir un corpus suffisamment grand. Étant donné que le corpus DatchaAct est un corpus plutôt petit, il est donc probable que les performances obtenues par ces plongements spécialisés ne soient pas meilleures que les performances obtenues par des plongements pré-entrainés sur un corpus beaucoup plus grand, même si ces plongements ne sont pas créés dans le but de prédire des actes de dialogue.

6.2.3.2. Résultats

	Acte Courant		Acte Suivant	
Plongements de phrases	Exactitude	MacroF1	Exactitude	MacroF1
Aléatoires+Affinés	**83,69**	78,15	46,21	26,45
Moyenne FastText	82,96	**79,47**	48,26	**30,09**
Skip-Thought	82,50	75,73	**48,30**	28,61

Table 6.1. – Résultats (en %) des évaluations des plongements de phrases sur les tâches de prédictions de l'acte courant et de l'acte suivant

En observant les résultats obtenus dans la table 6.1, dans un premier temps on peut constater, de manière prévisible, que les performances sont bien meilleures dans la tâche de prédiction de l'acte de dialogue courant avec 83% d'exactitude que dans la tâche de prédiction de l'acte de dialogue suivant avec 48% d'exactitude. Ce comportement était attendu, car sur la deuxième tâche en plus de devoir prédire des actes de dialogue, il est également nécessaire de prédire ce qui va se passer dans le futur, ce qui est de manière générale une tâche difficile. Ce qui est davantage intéressant avec ces résultats est d'observer ce qu'il se passe lorsqu'on change les types de plongements de phrases utilisés.

Sur la tâche de prédiction de l'acte de dialogue courant, les résultats des différents plongements sont proches. En considérant l'exactitude, les plongements aléatoires obtiennent un score supérieur de 0,73 points par rapport au score obtenu par les plongements issus de la moyenne des plongements FastText. En revanche, en considérant la MacroF1, ce sont ces derniers plongements qui obtiennent les meilleurs scores avec 79,47% de MacroF1. Le fait que les plongements Skip-Thought obtiennent une moins bonne MacroF1 (75,73%) est étonnant. Le score d'exactitude étant similaire, cela semble indiquer que certains actes de dialogue, moins fréquents, sont moins bien prédits lors de l'utilisation de ces plongements, alors que la prise en compte du contexte lors de l'apprentissage de ceux-ci semblait être pertinente pour cette tâche.

Sur la tâche de prédiction de l'acte de dialogue suivant, les résultats obtenus ne sont pas du tout ceux attendus. En effet, les plongements Skip-Thought n'obtiennent pas les meilleurs scores et sont même inférieurs de 1,48 points par rapport aux plongements issus d'une moyenne. Ceci est très surprenant étant donné que les plongements Skip-Thought sont entrainés pour prédire la phrase suivante (et précédente). Ceci peut être dû au fait que le discours conversationnel n'est pas suffisamment pris en compte par ces plongements de phrases qui ne permettent pas de faire mieux que des plongements de phrases ne s'intéressant qu'au contenu de la phrase elle-même. En effet, il est probable que les plongements Skip-Thought prennent en compte davantage le contexte sémantique

que le contexte discursif étant donné que l'entrainement se fonde totalement sur la prédiction de séquences de mots.

Afin de confirmer cela, il est donc nécessaire de construire des représentations semblables à SKIP-THOUGHT mais qui se focalisent davantage sur le discours conversationnel que sur les mots des phrases lors de leur apprentissage.

6.3. Prendre en compte explicitement les interactions dans les représentations

On a pu constater précédemment que les plongements SKIP-THOUGHT modélisent mal le contexte discursif des tours de parole. Or ces vecteurs ont initialement été créés dans le but de prendre en compte le contexte de production des phrases. Cet algorithme se fonde sur une approche entièrement non supervisée n'exploitant que les séquences de mots des phrases. Afin de déterminer s'il est possible de mieux prendre en compte le discours conversationnel, il est donc probablement nécessaire de guider l'apprentissage afin que les interactions entre tours de parole soient prises en compte de manière plus explicite.

Dans ce but, l'approche que je propose consiste à guider l'apprentissage des représentations en utilisant explicitement des annotations sur le discours conversationnel. L'apprentissage de plongements de phrases nécessite beaucoup de données et il est donc indispensable d'utiliser des annotations que l'on peut obtenir de manière automatique — il n'est pas envisageable d'annoter manuellement un corpus très volumineux. Pour cela, les actes de dialogue sont de bons candidats pour trois raisons :

1. Les étiqueteurs existants en actes de dialogue sont relativement performants (86% d'exactitude sur les conversations qui nous intéressent) et il est donc possible d'envisager une propagation automatique des annotations sur un grand corpus.
2. Contrairement à des annotations discursives mettant en relation les tours de parole, les actes de dialogue ne portent que sur un unique tour de parole et n'ont pas de liens directs avec les autres tours. Une erreur d'annotation sur un tour de parole a donc moins de risque de rendre l'annotation sur l'ensemble du dialogue invalide.
3. Ils sont également utilisés par nos tâches d'évaluation des plongements de phrases. Cela permettra de se mettre dans des conditions optimales pour obtenir les meilleures performances possibles sur ces tâches. En effet, l'objectif est de voir s'il est important que les modèles de plongements de mots capturent explicitement le discours conversationnel.

Dans cette section, je propose de nouveaux modèles de plongements de phrases en utilisant une annotation en actes de dialogue comme supervision. Ces plongements doivent ainsi me permettre de répondre aux questions suivantes :

1. Est-ce que l'apprentissage de plongements de phrases issues des dialogues est suffisant pour modéliser le discours conversationnel ? Est-il nécessaire d'utiliser des algorithmes de productions prenant en compte explicitement les interactions ? Une réponse partielle a déjà été apportée dans la section précédente et sera donc complétée ici.
2. Peut-on construire des représentations de phrases modélisant le discours conversationnel ? Quelles interactions dans les dialogues qui sont mal prises en compte dans les plongements usuels peuvent être mieux modélisées par des représentations spécialisées ? Pour répondre à ces questions, je réaliserai une analyse fine des résultats par étiquette.

De manière générale, la production de plongements s'appuyant explicitement sur des annotations en actes de dialogue doit me permettre de juger si les modèles de plongements existants capturent implicitement la dimension interactive des dialogues. Deux cas de figure se présentent :

1. Les plongements spécialisés ne permettent pas une amélioration des résultats sur les tâches d'évaluation par rapport aux plongements de phrases usuels. Dans ce cas, cela montrerait que ces derniers permettraient une prise en compte implicite du discours conversationnel et qu'une prise en compte explicite des actes de dialogue lors de l'apprentissage des plongements ne serait pas utile pour avoir une meilleure modélisation du discours.
2. Les plongements spécialisés améliorent les résultats. Dans ce cas, on pourrait conclure que les plongements usuels ne permettraient pas une prise en compte suffisante du discours conversationnel et qu'il existerait une marge de progression afin de mieux le prendre en compte. Par ailleurs, l'utilisation explicite d'actes de dialogue lors de l'apprentissage serait une solution à ce problème.

Dans la sous-section 6.3.1 je décris le processus de création de différents modèles de plongements de phrases utilisant des annotations en actes de dialogue. Dans la sous-section 6.3.2, j'évalue ces différents modèles afin de pouvoir les comparer aux plongements de phrases étudiés dans la section précédente.

6.3.1. Apprentissage de plongements de phrases reposant sur les actes de dialogue

Afin de forcer les plongements de phrases à modéliser, au moins en partie, le discours conversationnel, je propose d'utiliser la prédiction d'actes de dialogue comme tâche support. L'approche la plus basique est de simplement s'appuyer sur une tâche d'étiquetage en actes de dialogue et d'extraire les vecteurs correspondant aux représentations des tours de parole dans l'étiqueteur. Toutefois, on peut déjà constater plusieurs limites à une telle approche :

1. L'acte de dialogue est très lié au contenu du tour de parole et il y a donc le risque que le contexte de production soit peu pris en compte, même si en théorie le contexte est nécessaire pour correctement étiqueter certains tours de parole.
2. Même si le corpus d'apprentissage est très volumineux, il contient de nombreuses erreurs. Le fait de ne s'appuyer que sur l'acte de dialogue du tour risque de trop reproduire ces erreurs.

Il est donc important d'utiliser une approche prenant en compte le contexte de production des actes de dialogue et qui soit robuste au bruit. À la manière des vecteurs SKIP-THOUGHT, une façon de prendre en compte le contexte de production est de considérer les actes de dialogue des tours adjacents.

Afin de prendre en compte les différents aspects listés précédemment, je propose d'utiliser plusieurs tâches supports — reposant toutes sur les actes de dialogue — pour créer des plongements de phrases prenant en compte explicitement les interactions dans les dialogues. Le but de ces plongements est d'ensuite les comparer aux plongements présentés dans la section 6.2.2. Cette comparaison me permettra de déterminer si les plongements usuellement utilisés permettent une prise en compte implicite du discours conversationnel, ou si au contraire il est nécessaire d'avoir cette information de manière explicite lors de l'apprentissage.

Par ailleurs, les différentes tâches supports doivent permettre d'évaluer l'importance de la prise en compte du contexte de production des tours de parole lors de l'apprentissage des plongements. En effet, dans les résultats de la section 6.2, les plongements SKIP-THOUGHT semblent être moins intéressants que ceux issus d'une moyenne FASTTEXT. Est-ce dû à l'inutilité du contexte de production pour répondre aux tâches liées aux actes de dialogue ou est-ce dû à une mauvaise prise en compte du contexte dialogique ?

Afin de répondre à ces différentes interrogations, je propose quatre tâches supports, schématisées dans la figure 6.2 et présentées dans les paragraphes suivants :

RNN Acte Courant Ces plongements sont issus d'une tâche d'étiquetage classique en actes de dialogue, où pour chaque tour de parole, le réseau doit prédire l'acte de dialogue correspondant. Ce réseau est inspiré de celui utilisé dans la section 3.5 pour la tâche d'évaluation « Acte courant ». Pour un tour donné, l'étiqueteur n'a accès qu'au tour de parole courant et aux tours de parole passés. Le but de cette tâche est de créer des plongements optimaux pour la tâche d'évaluation correspondante « Acte courant ». Néanmoins, je m'attends à ce que ces plongements soient très peu adaptés à la tâche d'évaluation « Acte suivant » étant donné que le contexte est très peu pris en compte dans ces représentations.

RNN Acte Suivant Ces plongements sont issus d'une tâche très similaire à celle utilisée pour créer les plongements RNN ACTE COURANT à la différence que

cette fois-ci l'acte de dialogue à prédire pour un tour donné n'est pas l'acte de dialogue correspondant mais l'acte de dialogue du tour de parole suivant. Cette tâche est très semblable à la tâche d'évaluation « Acte suivant ». Le but de cette tâche est de créer des plongements optimaux pour la tâche d'évaluation correspondante.

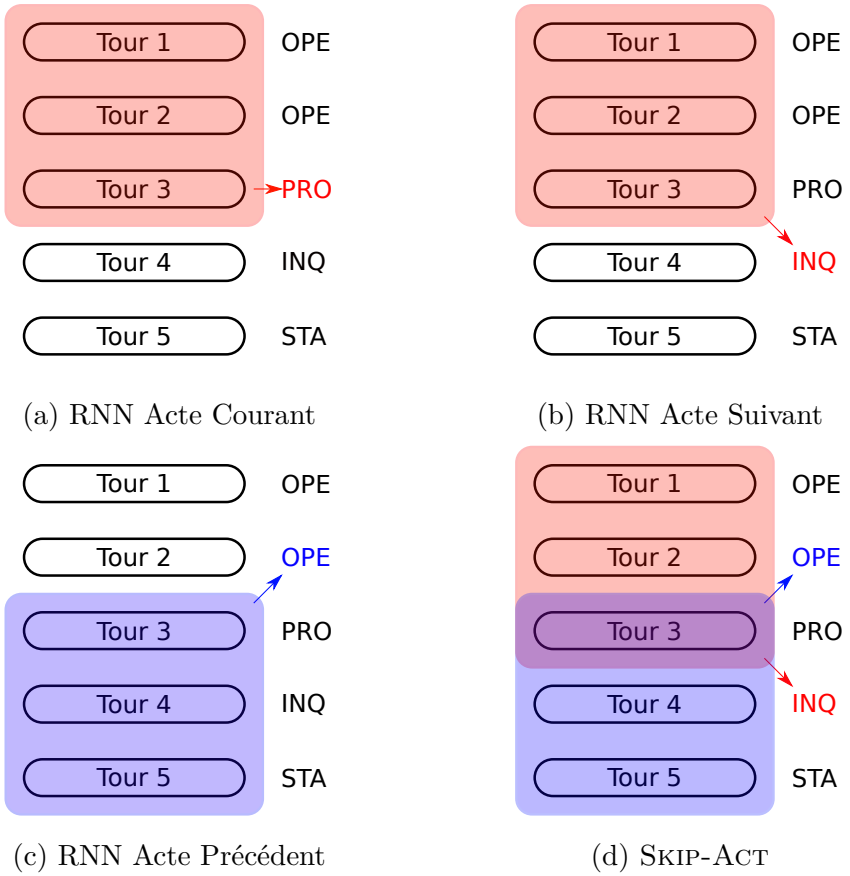
RNN Acte Précédent Ces plongements sont créés par la tâche symétrique de celle utilisée pour créer les plongements RNN ACTE SUIVANT. La tâche supervisée est donc pour un tour donné de prédire l'acte de dialogue précédent uniquement à partir du tour de parole courant et des tours de parole futurs.

Skip-Act Les *vecteurs* SKIP-ACT combinent le principe de création des plongements RNN ACTE SUIVANT et RNN ACTE PRÉCÉDENT. La tâche d'apprentissage est donc de prédire pour un tour donné à la fois l'acte de dialogue du tour suivant et celui du tour précédent. L'idée derrière ces plongements est similaire à celle derrière les vecteurs SKIP-THOUGHT, c.-à-d. de prédire les actes de dialogue précédent et suivant afin de modéliser le contexte de production discursif du tour de parole. Un peu à la manière des plongements de mots issus de SKIP-GRAM qui modélisent le sens des mots à partir des contextes d'apparition, je suppose que le fait de prendre en compte à la fois l'acte précédent et l'acte suivant permet de modéliser le rôle discursif des tours de parole à partir des différents contextes de production dans le discours de ceux-ci.

Pour produire ces quatre type de plongements, j'utilise un réseau de neurones hiérarchique [1]. Le réseau est donc sur deux niveaux : le premier niveau permet d'encoder les tours de paroles en construisant une représentation distributionnelle et le second niveau permet de produire une séquence d'actes de dialogue correspondant à la séquence de tours de parole en entrée. Le premier niveau prend en entrée les mots et produit un encodage à l'aide d'une couche LSTM. Le second niveau prend en entrée les représentations produites par le premier niveau et fonctionne ensuite de la même manière que l'étiqueteur utilisé pour l'évaluation des plongements de phrases (voir la sous-section 6.2.3) en adaptant simplement les cibles des sorties du réseau pour correspondre aux tâches supports — c.-à-d. prédire l'acte courant, suivant ou précédent. La figure 6.3 est un schéma du réseau utilisé pour apprendre les plongements RNN ACTE COURANT.

Pour les plongements SKIP-ACT, le réseau est légèrement différent étant donné qu'il doit résoudre deux tâches simultanément (prédire les actes suivant et précédent). Le modèle est simplement un modèle multi-tâche où le premier niveau du réseau reste commun aux deux tâches et le second niveau est en revanche indépendant pour chacune des deux tâches. La figure 6.4 est un schéma du réseau hiérarchique multi-tâche ainsi utilisé.

1. Architecture qui se calque sur l'aspect hiérarchique des dialogues : les mots constituent des tours qui constituent le dialogue.



(a) RNN Acte Courant (b) RNN Acte Suivant

(c) RNN Acte Précédent (d) Skip-Act

FIGURE 6.2. – Les différentes tâches de support utilisées pour construire des plongements de phrases prenant en compte le contexte dialogique

Une fois la phase d'apprentissage terminée, le modèle permet de générer des plongements de tours à partir des sorties des encodeurs du premier niveau du réseau hiérarchique. Sur le schéma, ceci correspond aux sorties e_i des parties en rouge du réseau. Le reste du réseau peut être ignoré. Lors de la génération des plongements de tours, il n'est donc plus nécessaire d'avoir une annotation en actes de dialogue.

6.3.2. Évaluation des plongements de phrases

On avait pu constater dans la section 6.2 que des plongements de phrases ne se fondant que sur le lexique, avec ou sans contexte, semblent ne pas très bien modéliser le discours conversationnel. Or modéliser le discours conversationnel est nécessaire si on souhaite que ces représentations soient utilisables dans des tâches liées au dialogue, et en particulier des tâches se fondant sur la cohérence de la structure dialogique (désenchevêtrement de sous-dialogues, prédiction des tours suivants, etc.).

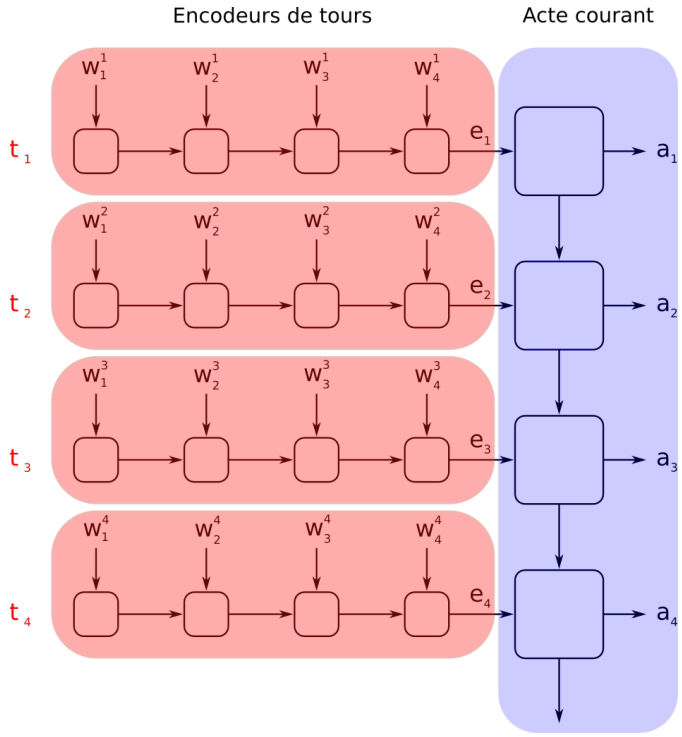


FIGURE 6.3. – Schéma du réseau utilisé pour apprendre les plongements de phrase prenant en compte le discours conversationnel. La partie en rouge correspond à l'encodeur permettant de produire les plongements de tours

Dans la section précédente, j'ai proposé plusieurs modèles de plongements de phrases qui utilisent des annotations en actes de dialogue lors de la phase d'apprentissage. Ces modèles n'ont pas pour objectif d'être les meilleurs sur l'ensemble des tâches où des plongements de phrases peuvent être utilisés. En revanche, le but est de voir s'ils permettent d'améliorer les résultats sur des tâches directement liées au discours conversationnel. Si les résultats sont positifs, on pourrait alors tirer deux conclusions :

— j'aurai été capable de produire des représentations distributionnelles du discours conversationnel au niveau des tours de parole ;

— j'aurai mis en évidence qu'il est nécessaire de prendre en compte le discours conversationnel lors de l'apprentissage de modèles de plongements de phrases.

6.3.2.1. Protocole expérimental

Les plongements de phrases RNN ACTE COURANT, RNN ACTE SUIVANT, RNN ACTE PRÉCÉDENT et SKIP-ACT sont appris en utilisant le corpus DATCHASAT annoté automatiquement en actes de dialogue, en ayant supprimé les dialogues

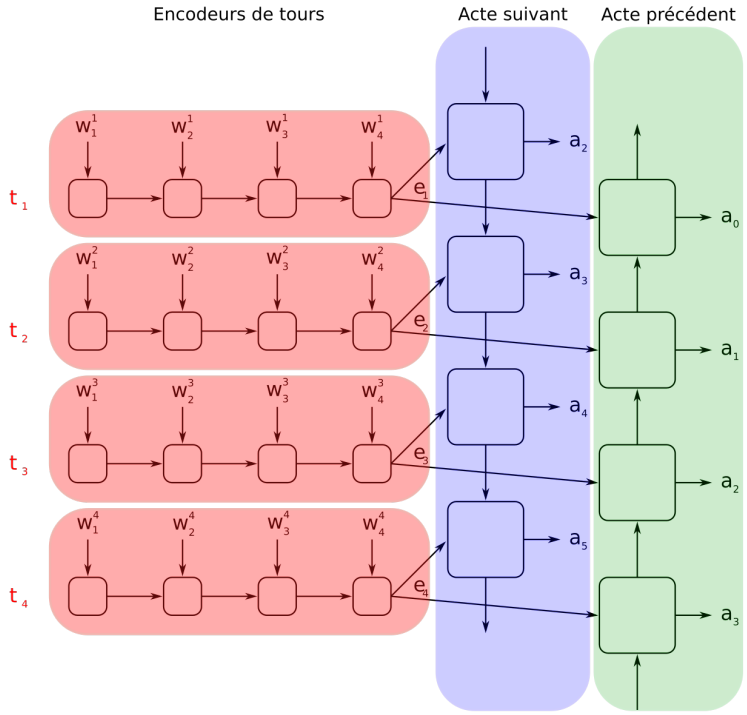


FIGURE 6.4. – Schéma du réseau utilisé pour apprendre les vecteurs SKIP-ACT. La partie en rouge correspond à l'encodeur permettant de produire les plongements de tours.

se trouvant également dans DATCHAACT. J'utilise également le sous-ensemble de développement (15 898 dialogues) pour sélectionner les plongements de tours en fonction de leur score de perte.

Afin de déterminer si les quatre nouveaux modèles de plongements permettent effectivement de modéliser le discours conversationnel, je propose de les évaluer sur les tâches d'évaluation « Acte Courant » et « Acte Suivant ». Les architectures et hyperparamètres utilisés pour les tâches d'évaluation sont les mêmes que dans la section 6.2.3. Encore une fois, le corpus utilisé pour évaluer les plongements de phrases est DATCHAACT.

Les métriques utilisées sont encore une fois l'exactitude et la MacroF1. Toutefois, le score F1 de chaque étiquette est également utilisé afin de pouvoir déterminer si certains actes de dialogue sont mieux modélisés que d'autres par les plongements de phrases. Les résultats des expériences utilisant les quatre nouveaux modèles sont présentés dans la sous-section 6.3.2.2.

Toutes les expériences réalisées jusqu'à présent portent sur l'ensemble des tours de parole. Or, il est évident que les actes de dialogue ne sont pas utilisés de la même manière ou dans les mêmes proportions par les clients et les téléconseillers. En particulier, on a pu voir dans la figure 3.4 du chapitre 3 que les clients produisent principalement des actes STA, PRO, INQ et ACK alors que

	Acte Courant		Acte Suivant	
Plongements de phrases	Exactitude	MacroF1	Exactitude	MacroF1
Aléatoires+Affinés	83.69	78.15	46.21	26.45
Moyenne FastText	82.96	79.47	48.26	30.09
Skip-Thought	82.50	75.73	48.30	28.61
RNN Acte Courant	84.74	80.47	48.54	31.42
RNN Acte Suivant	84.40	81.42	**49.97**	34.47
RNN Acte Précédent	83.02	80.44	48.77	31.96
Skip-Act	**85.24**	**82.16**	**49.96**	**35.33**

Table 6.2. – Évaluation des différents plongements de phrase

le téléconseiller a une production beaucoup plus variée. Il est donc probable que la tâche d'évaluation soit beaucoup plus difficile sur les tours du téléconseiller et il peut donc être intéressant d'observer comment se comportent les plongements en fonction des scripteurs. Dans la sous-section 6.3.2.3, je présente les résultats des expériences précédentes en ne prenant en compte que les prédictions des étiquettes des tours d'un des scripteurs choisi.

6.3.2.2. Résultats généraux

La table 6.2 présente les résultats obtenus par les différents types de plongements sur les deux tâches d'évaluation : la prédiction de l'acte courant et la prédiction de l'acte suivant. Les résultats obtenus sur les plongements construits à partir de méthodes déjà existantes sont également inclus dans cette table afin de pouvoir facilement comparer les performances.

Dans un premier temps, on peut constater que les performances des nouveaux plongements de phrases sont meilleures que les plongements étudiés précédemment sur les deux tâches d'évaluation. En particulier, si on compare les plongements Skip-Act avec les plongements issus d'une moyenne FastText, le gain est de $2{,}74$ points sur l'exactitude sur la tâche « Acte Courant » et de $1{,}7$ points sur l'exactitude sur la tâche « Acte Suivant ». On observe un phénomène semblable sur la MacroF1. Ceci montre que les interactions entre locuteurs sont peu modélisées implicitement par les modèles de plongements « usuels ». En effet, les résultats montrent une amélioration des performances lorsque l'apprentissage est adapté afin de prendre en compte les spécificités des dialogues explicitement, en utilisant par exemple des annotations en actes de dialogue.

Par ailleurs, les plongements Skip-Act, RNN Acte Courant et RNN Acte Suivant obtiennent également de meilleurs scores que les plongements affinés lors de l'apprentissage du modèle d'évaluation (respectivement $85{,}24\%$, $84{,}74\%$ et $84{,}40\%$ contre $83{,}69\%$ d'exactitude). Cela montre bien que la taille du corpus

DatchaAct ne suffit pas pour apprendre des représentations distributionnelles pertinentes. En effet, même s'il y a du bruit dans les annotations en actes de dialogue du corpus DatchaSat, celui-ci (par l'intermédiaire des plongements RNN Acte Courant et RNN Acte Suivant) permet d'obtenir de meilleurs résultats sur les deux tâches d'évaluation qu'en utilisant le corpus DatchaAct (utilisé pour apprendre les plongements initialisés aléatoirement et affinés).

En comparant cette fois-ci les nouveaux plongements entre eux, on peut constater que sur la tâche de prédiction de l'acte courant, étonnamment les meilleurs résultats ne sont pas obtenus par les plongements RNN Acte Courant. On aurait pu s'attendre à ce que ces plongements soient les plus adaptés à la tâche étant donné que ces plongements sont appris sur exactement la même tâche. Au contraire, les meilleurs résultats sont obtenus par les plongements Skip-Act qui ne sont pas entrainés à prédire l'acte courant mais à prédire le contexte de production des actes de dialogue. Ces comportements peuvent probablement en partie être expliqués par le fait que le corpus d'apprentissage DatchaSat n'a pas une annotation en acte de dialogue manuelle. Il est donc possible qu'il y ait eu un phénomène de sur-apprentissage sur les annotations bruitées avec les plongements RNN Acte Courant. Au contraire, les plongements Skip-Act sont probablement plus robustes au bruit en se focalisant sur le contexte de production des actes de dialogue.

Sur la tâche de prédiction de l'acte suivant, les plongements Skip-Act et RNN Acte Suivant obtiennent les meilleurs résultats avec un léger avantage pour les Skip-Act qui obtiennent un meilleur score de Macro-F1 ($35{,}33\%$ contre $34{,}47\%$). La prise en compte du contexte de production des actes de dialogue semble donc être bénéfique pour mieux prédire l'ensemble des actes de dialogue.

De manière générale, les plongements Skip-Act montrent que la prise en compte du contexte de production des actes de dialogue permet une meilleure modélisation des interactions entre les tours de parole. On peut également constater que les performances des Skip-Act se rapprochent beaucoup des performances obtenues par le CRF ayant produit les annotations automatiques sur DatchaSat (pour rappel, le CRF atteint les 86% d'exactitude). Ces plongements permettent donc de retrouver une grande majorité des caractéristiques nécessaires pour prédire les actes de dialogue.

On peut constater que certaines différences entre plongements sont plus marquées en observant la MacroF1. Ceci indique que certains modèles prédisent correctement certaines étiquettes bien spécifiques, alors que d'autres couvrent mieux l'ensemble d'entre elles. La figure 6.5 présente les scores F1 sur chaque type d'acte de dialogue pour chacun des types de plongements sur la tâche de prédiction l'acte de dialogue courant. Si on s'intéresse dans un premier temps aux scores par acte, on peut constater que la majorité des actes de dialogue ont des scores autour des 80%. Seuls les actes PRO avec environ 70% de score F1 et plus particulièrement OTH avec au mieux environ 50% de score F1 obtiennent des performances plus basses. Pour OTH, ceci est probablement dû au fait que

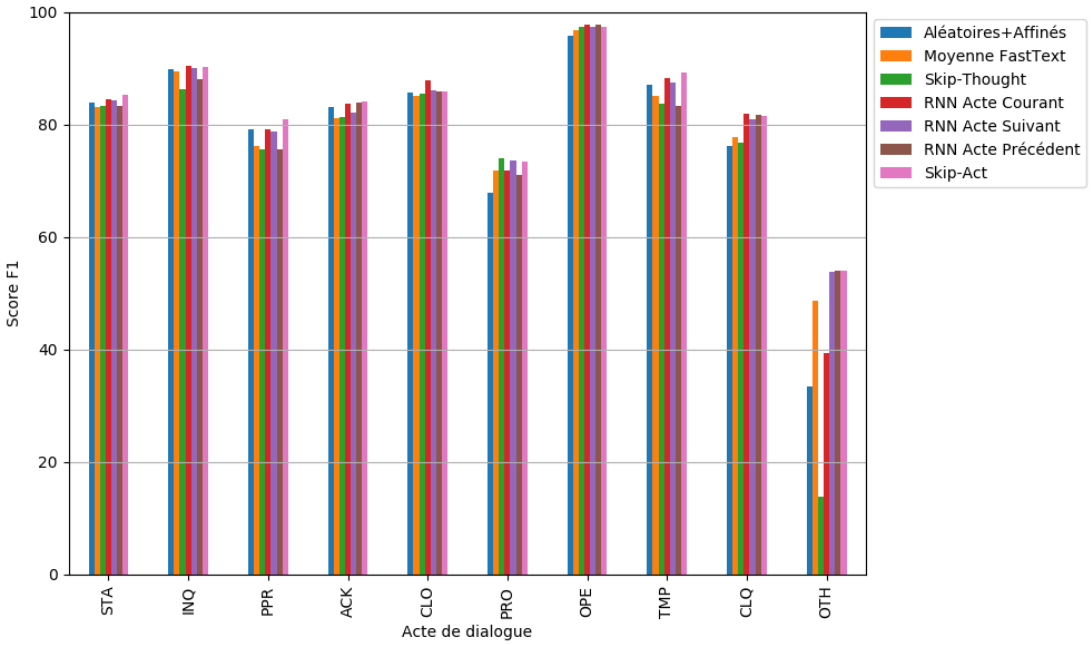


FIGURE 6.5. – Scores F1 obtenus par les différents plongements sur la tâche d'évaluation Acte Courant

c'est un acte de dialogue pouvant capturer énormément de types de tours de parole différents, ce qui rend nécessairement la prédiction de cet acte de dialogue plus difficile. Pour PRO, ceci peut sans doute être expliqué par le fait que cet acte nécessite une meilleure compréhension du contenu du tour afin de bien identifier qu'il s'agit d'une description du problème à résoudre.

En analysant cette fois les résultats obtenus par type de plongements, on constate que les vecteurs SKIP-ACT permettent d'améliorer sensiblement les résultats par rapport aux plongements plus classiques de type SKIP-THOUGHT ou moyenne FASTTEXT. En particulier, pour les actes PPR, CLQ et TMP le gain est de 5 points, pour l'acte ACK le gain est de 3 points et pour l'acte OTH le gain est de 5 points par rapport aux plongements de type moyenne FASTTEXT et de 40 points par rapport aux plongements SKIP-THOUGHT. Pour les autres actes, les vecteurs SKIP-ACT ne dégradent jamais les performances. Les gains observés confirment bien que les plongements SKIP-ACT modélisent mieux le discours conversationnel que les plongements « usuels », et ceci sur l'ensemble du discours.

Sur la figure 6.6, on peut observer les performances des différents plongements sur la tâche de prédiction de l'acte suivant. Les phénomènes observés sont cette fois-ci très différents. Outre le fait que les performances sont de manière générale bien plus basses, on peut également très facilement distinguer deux groupes d'actes de dialogue avec les actes STA, CLO, PRO et OPE obtenant des scores proches de 60% et avec les autres actes de dialogue obtenant des scores

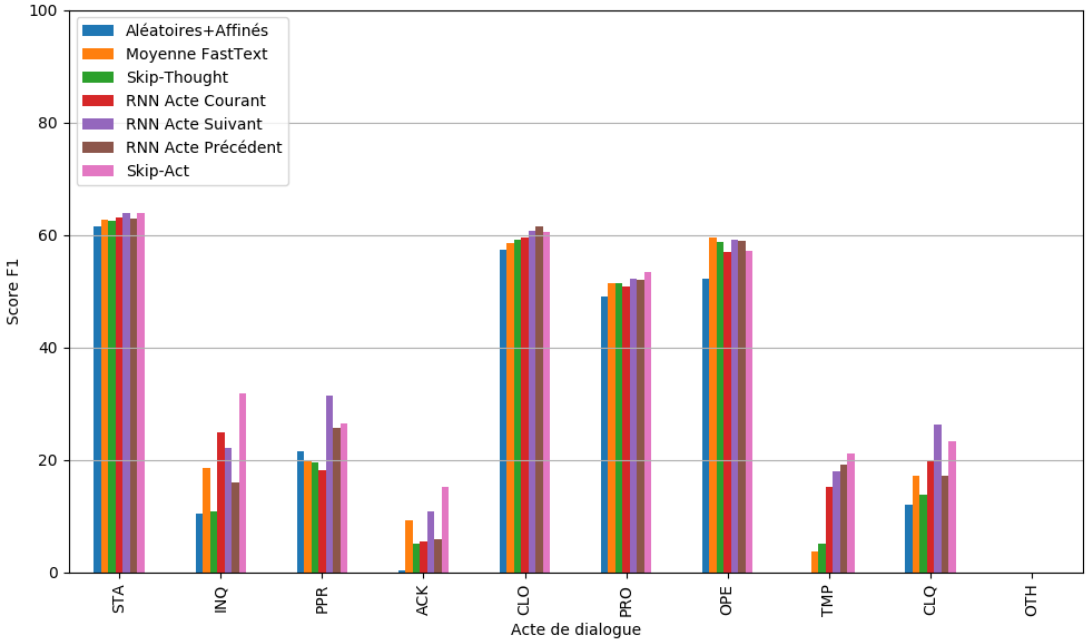


FIGURE 6.6. – Scores F1 obtenus par les différents plongements sur la tâche d'évaluation Acte Suivant

bien inférieurs au maximum autour de 20% et même 0% pour l'acte OTH. En comparant les différents types de plongements, on peut constater que une fois encore, les vecteurs SKIP-ACT permettent d'obtenir des gains plutôt importants sur certains actes de dialogue. Les gains sont surtout présents sur les actes ayant des performances plutôt faibles de manière générale. En particulier, sur INQ et TMP le gain est de plus de 15 points par rapport aux plongements classiques, sur ACK, PPR et CLQ on observe une amélioration d'au moins 5 points. Cependant, cette fois-ci les plongements SKIP-ACT ne sont pas systématiquement les meilleures représentations. En effet, pour PPR et CLQ les plongements RNN ACTE SUIVANT — appris sur la même tâche que la tâche d'évaluation — permettent d'obtenir respectivement 5 points et 3 points en plus que les plongements SKIP-ACT.

Ces résultats permettent de clairement identifier les types d'actes où une modélisation du discours conversationnel permet d'améliorer les prédictions. En effet, on peut voir sans trop de surprise que les actes de dialogue localisés presque toujours aux même endroits dans les conversations, tels que les ouvertures et les fermetures, sont prédits sans trop de difficultés quel que soit le modèle de plongements utilisé. En revanche, les actes de dialogue qui sont beaucoup plus dépendant du déroulement du dialogue tels que les questions, les demandes d'attente ou les réponses différentes des affirmations bénéficient clairement du fait d'utiliser des plongements construits pour prendre en compte le discours conversationnel pour les prédire.

	Uniquement le téléconseiller		Uniquement le client	
Plongements de phrases	Exactitude	MacroF1	Exactitude	MacroF1
Aléatoires+Affinés	84.22	77.38	83.01	58.58
Moyenne FastText	82.48	77.31	83.59	60.97
Skip-Thought	80.36	74.75	**85.31**	59.13
RNN Acte Courant	84.70	79.01	84.78	**64.16**
RNN Acte Suivant	84.30	82.42	84.54	63.20
RNN Acte Précédent	83.24	80.11	82.74	61.88
Skip-Act	**85.48**	**82.94**	84.93	63.99

Table 6.3. – Évaluation des différents plongements de phrases sur la tâche « Acte Courant » en ne conservant que les tours d'un des deux scripteurs.

Un deuxième aspect intéressant de ces résultats est que de manière étonnante, les plongements construits simplement à partir d'une moyenne de plongements de mots parviennent à obtenir des performances régulièrement meilleures que les performances obtenues par les plongements Skip-Thought pourtant construits de manière à prendre en compte le contexte dans lequel se trouve la phrase. En outre, les bonnes performances des vecteurs Skip-Act (comparées à celles des vecteurs Skip-Thought) montrent qu'il est nécessaire d'avoir explicitement des informations en lien avec le discours conversationnel — sous la forme d'actes de dialogue dans notre cas — afin de correctement prendre en compte le contexte de production dialogique lors de l'apprentissage des plongements.

6.3.2.3. Étude des performances en fonction des scripteurs

Étant donné que la distribution des différents actes de dialogue est très différente en fonction du scripteur, il est intéressant d'étudier le comportement des différents plongements sur les tâches d'évaluation en fonction du scripteur ayant produit les tours de parole.

Les tables 6.3 et 6.4 présentent les résultats obtenus en ne prenant en compte que les tours de parole du téléconseiller ou du client, respectivement sur les tâches « Acte courant » et « Acte suivant ». Quelle que soit la tâche, on peut constater que les résultats diffèrent en fonction du scripteur considéré. En s'intéressant uniquement aux tours du téléconseiller, on constate que les résultats sont très proches de ceux obtenus lorsque l'ensemble des tours sont utilisés, avec les vecteurs Skip-Act qui obtiennent les meilleurs résultats. Toutefois, on remarque que les différences entre plongements sont davantage marquées (5 points d'exactitude d'écart entre les vecteurs Skip-Act et Skip-Thought, 8 points avec la MacroF1).

En revanche, lorsqu'on s'intéresse uniquement aux tours du client, les résultats

	Uniquement le téléconseiller		Uniquement le client	
Plongements de phrases	Exactitude	MacroF1	Exactitude	MacroF1
Aléatoires+Affinés	35.87	23.16	59.48	21.13
Moyenne FastText	37.78	27.02	61.71	21.80
Skip-Thought	37.07	25.39	**62.70**	20.49
RNN Acte Courant	38.90	29.00	60.89	21.74
RNN Acte Suivant	41.29	32.60	61.09	22.91
RNN Acte Précédent	38.80	28.81	61.56	21.73
Skip-Act	**42.30**	**33.56**	59.78	**23.79**

Table 6.4. – Évaluation des différents plongements de phrases sur la tâche « Acte Suivant » en ne conservant que les tours d'un des deux scripteurs.

deviennent très différents. Il est intéressant de constater que les scores de MacroF1 s'effondrent totalement (en perdant 20 points sur la tâche « Acte courant » et 10 points sur « Acte suivant »). Par ailleurs, les vecteurs Skip-Act n'obtiennent plus les meilleurs scores d'exactitude, pour laisser la place aux vecteurs Skip-Thought.

Les résultats sur les tours du client ne sont pas surprenants. En effet, la distribution des actes de dialogue étant très déséquilibrée chez le client, il est plus important de bien prédire l'acte STA que les autres actes. Bien prédire les actes de dialogue du client nécessite donc peu de contexte. Au contraire, sur les tours du téléconseiller, il y a une beaucoup plus grande variété d'actes de dialogue, le téléconseiller étant le meneur du dialogue. Par conséquent, il est beaucoup plus difficile d'anticiper les actes de dialogue à venir, comme on peut le voir dans les résultats sur la tâche « Acte suivant ». Il est donc beaucoup plus important que les plongements prennent en compte le contexte dialogique et les interactions entre scripteurs pour pouvoir prédire l'acte de dialogue suivant. Par conséquent, les tours du téléconseiller sont plus intéressants pour évaluer la qualité des représentations des tours de parole et permettent de valider le fait que les vecteurs Skip-Act permettent une meilleure prise en compte du discours conversationnel que les plongements Skip-Thought ou moyenne FastText. En effet, contrairement à ces derniers, les vecteurs Skip-Act permettent à l'étiqueteur de mieux prédire les actes de dialogue dépendant des interactions entre scripteurs tels que les questions, les acquiescements ou les mises en attente.

6.4. Conclusion

Le but de ce chapitre était d'étudier la manière dont le discours conversationnel de surface peut être caractérisé à l'aide de représentations distributionnelles

des phrases. Les cadres d'évaluation de plongements de phrases sont, dans leur très grande majorité, utilisés afin de les évaluer dans le contexte de documents produits par une seule personne (articles, *tweet*, légendes, etc.). Or, une particularité des dialogues est qu'au moins deux personnes interagissent ensemble, et les locuteurs ne sont pas capables d'anticiper tous les énoncés qui vont être produits. De ce fait, les interlocuteurs vont communiquer leurs intentions aux autres et ils vont réagir aux différents énoncés produits par leurs interlocuteurs.

Lors de la création de représentations des phrases dans le cadre de conversations, il est donc important de prendre en compte le discours conversationnel afin de correctement capturer les différentes interactions qu'il peut y avoir dans le dialogue. Afin de déterminer si des modèles de plongements de phrases sont capables de modéliser le discours conversationnel, j'ai proposé un cadre d'évaluation se fondant sur deux tâches de prédictions d'actes de dialogue : Acte Courant et Acte Suivant. Ces tâches ne permettent pas d'évaluer l'ensemble des relations dialogiques qui existent, mais elles permettent d'avoir une première idée sur la qualité des représentations pour modéliser le discours de surface — c.-à-d. les fonctions de communications des différents énoncés.

En utilisant ce cadre d'évaluation, j'ai évalué deux modèles de plongements se fondant principalement sur le contenu des phrases, avec ou sans contexte de production. On a pu constater que ceux-ci modélisent peu le discours conversationnel. En particulier, les résultats obtenus avec les plongements SKIP-THOUGHT montrent qu'il n'est pas suffisant de se fonder sur la seule dimension lexicale.

Dans le but d'obtenir des représentations distributionnelles du discours conversationnel de surface, je propose un nouveau modèle de plongements de phrase : les vecteurs SKIP-ACT. Ces vecteurs sont construits afin de prendre explicitement en compte le contexte de production des actes de dialogue. On a pu constater que ces vecteurs obtiennent de meilleurs résultats sur les deux tâches d'évaluation proposées que les plongements étudiés auparavant. En particulier, on a pu observer que ces plongements permettent de mieux modéliser l'ensemble des actes de dialogue, en particulier sur les actes directement en lien avec les interactions dans le dialogue.

Toutefois, le travail sur les vecteurs SKIP-ACT n'est pas terminé. Il serait intéressant d'évaluer ces représentations sur des tâches d'évaluations de structures syntaxiques et sémantiques. En outre, il pourrait être intéressant de chercher à combiner les architectures permettant de créer les vecteurs SKIP-THOUGHT et SKIP-ACT afin d'obtenir des vecteurs modélisant explicitement à la fois le contexte lexical et le contexte discursif.

Par ailleurs, il serait intéressant d'évaluer d'autres modèles de plongements tels que les vecteurs INFERSENT — appris à l'aide d'une tâche d'inférence — ou des vecteurs issus de BERT. Ceci permettrait d'avoir une vision plus globale sur la manière dont le discours conversationnel pourrait être modélisé de manière implicite.

Enfin, le cadre d'évaluation proposé ici se repose uniquement sur le discours de

surface. Le discours de surface est intéressant car il permet de mettre en évidence les différentes intentions des interlocuteurs, mais il ne permet pas de mettre en relation les différents tours de parole. En particulier, le cadre d'évaluation tel qu'il est défini ne permet pas de déterminer si un modèle de plongements est capable de modéliser des relations entre des tours de parole éloignés.

Dans le chapitre suivant, je vais construire des représentations du discours conversationnel mettant en évidence les relations entre tours de parole. Il sera alors intéressant de voir si les plongements SKIP-ACT permettent d'aider à la construction de telles représentations.

Chapitre 7.

Analyse profonde du discours conversationnel avec peu d'annotations

Sommaire

7.1 Introduction 166
7.2 Annotation du discours conversationnel en dépendance 168
7.2.1 Spécificités du corpus DATCHA 168
7.2.2 Schéma d'annotation utilisé 169
7.2.3 Annotation du corpus DATCHA 176
7.3 Entrainer un analyseur du discours avec peu de données 176
7.3.1 Induire une grammaire hors-contexte 178
7.3.2 Apprendre une grammaire hors-contexte probabiliste 186
7.3.3 Induire les arbres discursifs à partir d'une PCFG 188
7.3.4 Expérimentations 189
7.4 Utiliser davantage de données à l'aide d'annotations automatiques . 195
7.4.1 Résoudre les problèmes de couverture des PCFG 196
7.4.2 Prise en compte du lexique par un analyseur en dépendance 197
7.4.3 Expérimentations 199
7.5 Conclusion 206

7.1. Introduction

Dans les contributions présentées jusqu'à présent, les choix des approches étudiées ont à chaque fois été guidés par un fait : la disponibilité des données annotées en quantités suffisantes afin d'entrainer divers modèles, en particulier neuronaux. Le chapitre 5 a montré qu'il était difficile d'exploiter des données n'ayant qu'un lien indirect avec le discours conversationnel pour construire des représentations discursives de manière bout en bout. C'est pourquoi nous avons fait le choix dans le chapitre 6 de construire des représentations à partir d'actes

de dialogues qui sont des annotations explicites du discours de surface d'une conversation. Cette approche m'a ainsi permis de construire des plongements de tours de parole conçus de sorte à prendre en compte le contexte local du discours de surface : les vecteurs SKIP-ACT.

Ces nouvelles représentations sont intéressantes car elles permettent de modéliser concrètement le discours sous la forme de représentations distributionnelles qui sont aisément utilisables en entrée de divers modèles, en particulier de modèles neuronaux. Cependant, elles ne permettent pas de construire une structure portant sur l'ensemble d'une conversation. En particulier, il n'est pas possible de déterminer les liens existants entre les différents sous-dialogues, ni de déterminer quels sont les différents enjeux dialogiques et les cheminements permettant d'y répondre (ou pas) dans le dialogue.

Dans le chapitre 1, j'ai décrit différentes approches permettant de modéliser le discours conversationnel. L'une d'entre elle consiste à réaliser une analyse profonde du discours en construisant des arbres — ou des graphes — discursifs permettant de mettre en relation chaque tour de parole ou sous-dialogue avec les autres parties du dialogue. C'est ce type de structures que nous allons chercher à prédire ici.

À l'échelle du dialogue, ces constructions arborescentes du discours permettraient de mettre en lumière les différents enjeux dialogiques introduits par les différents locuteurs, et ainsi d'identifier les différents sous-dialogues de la conversation ainsi que leur rôle dans le discours : répondre au problème général donnant lieu au dialogue, résoudre une sous-problématique, ou simplement introduire des sous-dialogues de politesse, protocolaires ou de réaction. Ces modélisations pourraient alors permettre de réaliser divers traitements sur les dialogues tels que des opérations de désenchevêtrements ou encore de construire des mesures de similarité entre dialogues et sous-dialogues. Cependant, une limite majeure de ces schémas d'annotation est que le processus d'annotation est complexe et couteux. Ceci pousse à aller vers des approches reposant sur peu d'annotations, tout en permettant une bonne modélisation du discours conversationnel.

Une approche possible est d'enrichir le peu de corpus disponible en augmentant artificiellement la quantité de données disponibles. Sur les images, l'enrichissement des images est un procédé fréquemment utilisé afin d'augmenter la taille du corpus en se basant sur les données existantes [PW17 ; SK19]. Pour faire cela, diverses opérations telles que des rotations ou des changements de couleurs sont appliqués sur les données originelles.

On peut alors se demander s'il ne serait pas envisageable d'enrichir un petit corpus de conversations ayant été annoté manuellement avec les relations dialogiques entre tours de parole. Dans ce chapitre et dans le cadre d'une analyse du discours, je vais chercher à déterminer s'il est préférable d'essayer d'enrichir les données — même si cela introduit du bruit — ou si au contraire il est préférable de n'utiliser que peu de données de bonne qualité.

Je présenterai dans la section 7.2 le schéma d'annotation utilisé sur le corpus DATCHA. Ensuite, j'introduirai dans la section 7.3 de premiers modèles construits à partir de grammaires permettant d'obtenir une première analyse profonde du discours sur les données DATCHA. Enfin, dans la section 7.4 j'essaierai d'exploiter les premiers modèles à base de grammaires afin d'étudier s'il est possible d'utiliser des approches plus gourmandes en données sans réaliser d'annotations manuelles additionnelles.

7.2. Annotation du discours conversationnel en dépendance

Afin de pouvoir répondre à la question de l'utilité de l'enrichissement du corpus, il est nécessaire de disposer d'annotations des conversations permettant de mettre en évidence les relations entre les tours de parole. Nous disposons déjà d'annotations en actes de dialogue mais ceux-ci décrivent uniquement les intentions des tours de parole et ne permettent pas de déterminer les relations que peuvent avoir les tours de parole entre eux.

Dans la section 1.5 du chapitre 1 (Le dialogue et l'analyse du discours), plusieurs schémas d'annotation ont été mis en avant dans le but d'annoter des dialogues avec des structures arborescentes modélisant le discours. Nous devons donc voir comment se fonder sur ces schémas afin de déterminer un schéma d'annotation adapté aux données se trouvant dans DATCHA et aux problématiques qui se posent dans cette thèse.

Dans la section 7.2.1, nous regarderons quelles sont les particularités du corpus DATCHA pouvant avoir une influence sur l'analyse du discours est donc sur le schéma d'annotation utilisé. Ensuite, dans la section 7.2.2 le schéma d'annotation utilisé sur le corpus DATCHA sera décrit.

7.2.1. Spécificités du corpus Datcha

Dans la section 1.5 du chapitre 1, les dialogues étudiés sont de natures très différentes, ce qui peut avoir une influence sur la manière dont les schémas d'annotation ont été définis. De ce fait, il est important d'étudier les particularités du corpus DATCHA afin d'identifier les différents aspects des dialogues qu'il est important de capturer dans l'analyse du discours.

Les dialogues de DATCHA sont issus de l'assistance clientèle d'Orange et par conséquent, certains phénomènes linguistiques sont directement liés à l'assistance, qui sont peu étudiés dans les travaux de AFANTENOS et al. [Afa+15] et de XUE et al. [XSJ16]. En premier lieu, ce sont des conversations très protocolaires en ouverture et fermeture des dialogues et il est donc important de bien modéliser ces protocoles dans l'analyse du discours. Un autre phénomène très

présent dans DATCHA est le fait que les téléconseillers vont fréquemment demander aux clients de réaliser des actions externes au dialogue. Ces demandes se font généralement de manière impérative, formes qui ne sont pas forcément très fréquentes dans d'autres types de dialogue. De manière similaire, les téléconseillers peuvent également souvent avoir besoin de réaliser des recherches ou des actions en dehors du dialogue pouvant prendre un certain temps. Dans ce cas, les téléconseillers vont généralement indiquer aux clients qu'il va y avoir une pause dans le dialogue.

Les dialogues qui constituent DATCHA ont également la particularité d'être très asymétriques où le téléconseiller est généralement celui qui guide le déroulement de la conversation. Ceci est très différent des dialogues étudiés dans STAC, par exemple, où tous les scripteurs peuvent à tour de rôle mener la discussion.

7.2.2. Schéma d'annotation utilisé

Afin de construire des structures arborescentes du discours dans le cadre de DATCHA, j'ai défini un schéma d'annotation spécifique au corpus DATCHA. Celui-ci se fonde sur le schéma utilisé dans le cadre de STAC [Afa+15]. Cependant, quelques modifications sont réalisées afin de mieux correspondre aux conversations rencontrées dans le corpus DATCHA.

Une autre différence majeure par rapport au schéma d'annotation utilisé dans STAC est que la structure obtenue n'est pas un graphe orienté mais un arbre. En effet, dans STAC plusieurs interlocuteurs peuvent faire référence à un même tour de parole et il est possible pour un interlocuteur de produire un tour de parole en réaction à plusieurs tours en même temps. Ces phénomènes imposent la modélisation sous forme d'un graphe. Or, les conversations dans DATCHA sont uniquement entre deux interlocuteurs. Il est donc toujours possible de déterminer un unique tour de parole ayant mener à la production d'un autre tour et ainsi se limiter à un arbre.

Comme dans le sous-corpus DATCHAACT annoté en actes de dialogue, les tours de parole sont définis par les scripteurs directement et aucune segmentation additionnelle n'est réalisée (c.-à-d. que la fin d'un tour est marquée par l'utilisation de la touche « Entrée » du scripteur). J'ai décidé de ne pas remettre en cause les décisions des scripteurs pour deux raisons :

— je ne m'intéresse pas aux relations discursives intra-tours, il n'est donc pas nécessaire d'avoir un niveau de segmentation fin ;

— ceci me permet de pouvoir utiliser les annotations en actes de dialogue dans DATCHAACT qui utilisent la même définition des tours de parole.

Les relations de dépendance entre tours peuvent être divisées en deux catégories : les relations discursives et les relations dialogiques. Une *relation discursive* décrit des phénomènes discursifs provenant généralement d'un même

Relation	Tag	Description
Opening	Ope	Tours d'ouverture du dialogue
Specification	Spe	Introduction d'un nouvel enjeu dialogique
Response	Res	Apport d'une nouvelle information attendue par le tour du gouverneur
Imperatif	Imp	Le scripteur demande qu'une action extérieur soit réalisée
Temporisation	Tmp	Introduction d'une pause dans le dialogue
Acknowledgement	Ack	Acquiescement des propos de l'interlocuteur
Ending	End	Le scripteur indique le souhait de clore un sous-dialogue
Closing	Clo	Tours de fermeture du dialogue
Opinion	Opi	Information subjective non nécessaire pour la résolution de l'enjeu dialogique
Relation Discursive	Ela	Étiquette générique utilisée pour toutes les relations discursives
Explicit Relation Demand	D + tag	Demande explicite à l'interlocuteur d'utiliser une relation spécifique

TABLE 7.1. – Étiquettes des relations dialogiques

scripteur. Ces relations peuvent décrire des phénomènes d'*élaborations*, de *corrections*, d'*explications*, de *narrations*, de *résumés* ou de *contrastes*. Une *relation dialogique* est une relation décrivant généralement des phénomènes d'échanges et de communication entre scripteurs différents.

Dans le cadre de mes travaux, c'est surtout les interactions entre interlocuteurs qui m'intéressent. J'ai donc fait le choix de me concentrer sur les relations dialogiques. Les relations discursives seront considérées comme étant une seule et même relation dans notre schéma d'annotation. Ceci a évidemment pour conséquence d'appauvrir la structure du discours, cependant cela permet de grandement simplifier l'annotation en mettant en avant la modélisation des interactions entre interlocuteurs.

Dans le schéma d'annotation utilisé, l'étiquette d'une relation permet de préciser la fonction d'un tour de parole dans le dialogue vis-à-vis d'un autre tour de parole — son gouverneur. Dans le contexte de l'analyse du discours, le *gou-*

verneur d'un tour de parole désigne le tour de parole qui provoque la production du tour de parole étudié. Un tour de parole ne peut avoir qu'un seul gouverneur dans notre schéma d'annotation, et un gouverneur spécial — la racine — est utilisé lorsqu'un tour n'est gouverné par aucun tour de parole.

Les différents types de relations étudiées sont brièvement décrites dans la table 7.1. Dans ce qui suit, les différentes relations dialogiques sont davantage détaillées. Il est également possible de voir leur utilisation dans le contexte de la conversation se trouvant en annexe A.

Specification Cette relation est utilisée lorsqu'un scripteur cherche à expliciter un enjeu dialogique à son interlocuteur. Elle permet d'orienter le dialogue vers un problème ou un sous-problème et ouvre ainsi un nouveau sous-dialogue. Cette relation peut aussi bien servir à décrire l'enjeu principal (le problème général du client) que des enjeux secondaires (le téléconseiller demande des informations sur le client). Le gouverneur est le tour qui a amené un scripteur à expliciter un nouvel enjeu.

Response Cette relation est utilisée pour décrire les tours qui apportent une information nouvelle attendue explicitement par le scripteur ayant produit le gouverneur. Le but est de répondre à un enjeu dialogique en cours. Les réponses sont généralement des affirmations ou négations. Le gouverneur d'une relation `Response` est le tour de parole introduisant l'enjeu dialogique auquel une réponse est apportée.

Imperative Cette relation est utilisée lorsqu'un scripteur demande à son interlocuteur de réaliser explicitement une action externe au dialogue. Ces tours sont produits lorsqu'il parait nécessaire d'agir sur l'environnement extérieur au dialogue afin de faire évoluer l'enjeu dialogique en cours. Les tours produits sont souvent des phrases sous la forme impérative ou interrogative. Le gouverneur d'un tel tour de parole est le tour ayant motivé l'action externe, généralement un tour décrivant un enjeu dialogique.

Les relations `Specification`, `Imperative` et `Response` peuvent être considérées comme étant celles qui construisent le squelette principal permettant de résoudre la problématique introduite dans les dialogues trouvés dans le corpus DATCHA. Voici un exemple de sous-dialogue avec son sous-arbre introduisant ces trois relations :

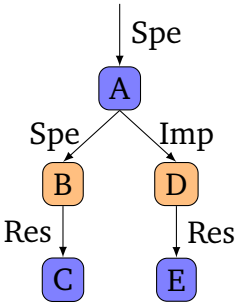
A. **CLIENT**: Mon décodeur TV est tombé en panne hier. Je souhaiterais connaitre la démarche à suivre pour le remplacer.

B. **TC**: Avez-vous branché le décodeur sur une autre prise murale ?

C. **CLIENT**: Absolument

D. **TC**: Je vous invite à éteindre/allumer le décodeur, au bout de 2 minutes le décodeur doit afficher 00 :00.

E. **CLIENT**: Déjà testé et ça ne corrige rien.



Opinion Théoriquement, dans le but de résoudre efficacement les différents enjeux dialogiques, les interlocuteurs s'échangent des arguments et informations objectives. Cependant, en pratique les interlocuteurs produisent également des tours de parole contenant des informations qui n'ont pas pour but de faire progresser la réalisation des enjeux dialogiques. Ces tours peuvent cependant avoir une influence sur les tours de parole suivant, par exemple dans le cas d'une opposition au propos de l'autre interlocuteur. Ces tours ont pour gouverneur le tour de parole ayant provoqué la volonté de produire cette production d'opinion.

Ending Cette étiquette est utilisée pour décrire la production de tours explicitant une volonté d'un des scripteurs de clore le dialogue, ou un sous-dialogue ouvert. Ces tours peuvent intervenir lorsqu'un des interlocuteurs considère qu'un enjeu dialogique a été suffisamment résolu, ou souhaite orienter la conversation vers un autre enjeu. Le gouverneur d'un tel tour est la racine du sous-dialogue dans lequel il se trouve.

Temporisation Cette relation décrit les tours où un scripteur suspend le dialogue temporellement. Le but peut être pour le scripteur d'indiquer qu'il va réaliser une action externe au dialogue et d'éviter une coupure définitive de ce dernier. Cette relation est fortement liée à l'acte de dialogue de temporisation (`TMP`). Le gouverneur est le tour précédent.

Acknowledgement Cette étiquette est utilisée pour décrire deux phénomènes : les acquiescements et les *backchannels*. Les scripteurs se servent des acquiescements afin d'expliciter la bonne réception et la compréhension des propos produits par l'interlocuteur. Le *backchannel*, surtout utilisé à l'oral mais existant également à l'écrit, permet au scripteur de ré-affirmer sa présence dans la conversation. Cette relation est liée à l'acte de dialogue d'acquiescement (`ACK`), mais il peut arriver que l'acte de dialogue soit une affirmation (`STA`).

Voici quelques exemples de tours de parole souvent associés à une relation Acknowledgement :

— Ok

— Oui

— Merci, je patiente.

— D'accord.

Les quatre relations Opinion, Ending, Temporisation et Acknowledgement sont des relations décrivant des tours de paroles qui ne sont pas indispensables à la résolution de l'enjeu dialogique principal. Toutefois, l'apparition ou non de certains de ces tours permet de voir la manière dont les deux interlocuteurs co-opèrent sur le plan de la communication. Voici un exemple de sous-dialogue introduisant des relations Ending, Opinion et Acknowledgement :

A. **CLIENT**: Ma livebox a pris la foudre. Que faire ?

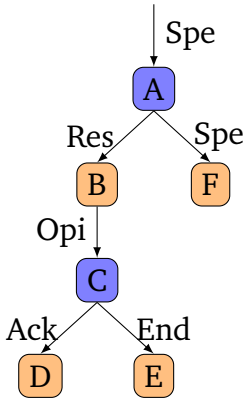
B. **TC**: Nous allons vous en envoyer une nouvelle dans les plus brefs délais.

C. **CLIENT**: Vous êtes vraiment efficace, c'est cool !

D. **TC**: Merci

E. **TC**: Ce n'est pas tout à fait terminé, revenons à votre problème.

F. **TC**: Quelle est votre adresse ?



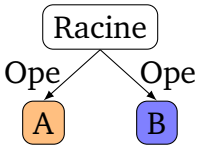
Opening Cette étiquette décrit la production d'un tour de parole mettant en relation des scripteurs. Le sujet de la discussion n'est pas encore précisé. Les interlocuteurs peuvent s'introduire, vérifier qu'ils ont l'interlocuteur désiré et que la communication est effective. C'est aussi à ce moment-là que les scripteurs présentent leur rôle dans la conversation. Cette relation est fortement liée à l'acte de dialogue d'ouverture (OPE).

Aucun enjeu dialogique n'est introduit dans ces tours. Le gouverneur d'un tel tour est soit la racine du dialogue s'il n'y a pas encore de contexte, soit un tour précédent, généralement lui-même d'ouverture, nécessaire pour la compréhension du tour.

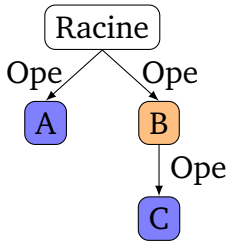
Voici deux exemples de sous-dialogues introduisant des relations Opening :

A. **TC**: Bonjour, je m'appelle TC

B. **CLIENT**: Bonjour



A. **CLIENT**: Bonjour
B. **TC**: Bonjour, je m'appelle TC
C. **CLIENT**: Enchanté TC



Closing Ces tours correspondent au phénomène de fermeture protocolaire du dialogue. C'est une étiquette fonctionnant de manière semblable à Opening. Cette relation est fortement liée à l'acte de dialogue de fermeture (CLO). Le gouverneur est la racine du dialogue.

À la différence de Ending, les scripteurs produisant cette relation sont convaincus que la conversation est bien terminée et s'attendent à n'avoir que des tours de fermeture à la suite.

Voici quelques exemples de tours de paroles hors-contextes associés à la relation Closing :

— Au revoir

— Je vous souhaite une agréable journée

— Bonne fin d'après-midi

Relations discursives Dans certains cas, les tours de parole ne sont pas produits dans le but d'interagir avec un interlocuteur. De multiples relations discursives sont habituellement utilisées afin de décrire ces phénomènes. Dans mon cas, je regroupe ensemble toutes les relations discursives au sein d'une même étiquette Elaboration. Les relations intra-tours ne sont pas l'objet de mon étude et la segmentation utilisée ne permet pas de correctement modéliser ces phénomènes.

Toutefois, les scripteurs vont parfois faire le choix de segmenter leurs propos en plusieurs tours de parole consécutifs. Pour ces cas-là, la relation Elaboration peut alors être utilisée. Le gouverneur d'une Elaboration est le tour précédent produit par le même scripteur.

Explicit Relation Demand Toutes les relations précédentes ont un point commun : le scripteur décide de produire un tour dans le but de communiquer une fonction illocutoire. Cependant, il peut arriver que l'un des scripteurs considère que son interlocuteur n'interagit pas de manière attendue. Dans ce cas, il peut indiquer quelles sont ses attentes afin que son interlocuteur produise le tour de parole souhaité. Toutes les étiquettes précédentes peuvent avoir une étiquette D+Étiquette associée permettant donc de demander à l'interlocuteur la production d'un tour de parole ayant la fonction dialogique de l'étiquette. Généralement, les sous-dialogues introduits par les étiquettes D+Étiquette pourraient être supprimés sans changer le sens de la conversation.

En pratique, seules trois relations D+Étiquette ont été utilisées sur nos données : DSpe, DRes et DImp. Je vais donc également décrire ces trois relations dialogiques.

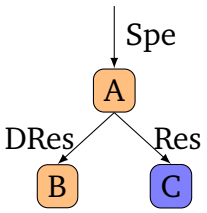
Demand Specification Cette relation est utilisée lorsqu'un scripteur souhaite savoir si son interlocuteur souhaite mettre en avant un nouvel enjeu. Sur nos données, c'est généralement le téléconseiller qui va faire ces demandes avec par exemple les questions suivantes :

— Que puis-je pour vous ?

— Avez-vous d'autres questions ?

Demand Response Cette relation a deux cas d'utilisation. Le premier est simplement lorsqu'un scripteur souhaite obtenir une réponse de son interlocuteur qui n'a pas encore été donnée. Voici un exemple de sous-dialogue, avec son sous-arbre dialogique, permettant d'illustrer ce premier cas :

A. TC: Quelle est votre adresse ?
B. TC: (Après quelques secondes sans réponses) Alors ?
C. CLIENT: 163 Avenue de Luminy



Le deuxième cas correspond aux situations où un scripteur souhaite demander une confirmation à son interlocuteur. Quelques exemples de questions correspondant à cette relation :

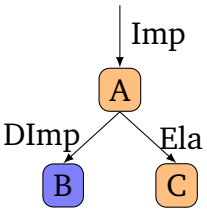
— Êtes-vous toujours là ?

— Vous êtes certains ?

— (Adresse donnée dans un tour précédent) C'est bien votre adresse ?

Demand Imperative Cette relation est utilisée lorsque le scripteur pense qu'il doit réaliser une action externe et demande donc à son interlocuteur ce qu'il doit faire. Voici un exemple de sous-dialogue avec son sous-arbre dialogique :

A. TC: Appuyez sur le bouton A.
B. CLIENT: Et ensuite, que dois-je faire ?
C. TC: Ensuite appuyez sur le bouton B.



7.2.3. Annotation du corpus Datcha

Le schéma d'annotation ayant été construit dans le but d'analyser les dialogues du corpus DATCHA, un sous-ensemble de conversations a été annoté manuellement. Afin d'avoir le plus d'annotation sur le discours possible, les conversations annotées proviennent du corpus DATCHAACT où les conversations sont annotées avec des actes de dialogue.

Relation	# Dev	# Test
Opening	118	133
Specification	454	435
Response	749	875
Imperative	82	50
Temporisation	66	78
Acknowledgement	260	368
Ending	19	1
Closing	144	183
Opinion	20	9
Elaboration	520	386
Explicit Relation Demand	191	235

TABLE 7.2. – Distribution des relations dialogiques dans DATCHAREL

Uniquement 182 conversations ont été annotées manuellement avec une analyse en dépendance. En effet, il est beaucoup plus long et difficile de réaliser une telle annotation que de réaliser une annotation en actes de dialogue. Ce corpus, nommé DATCHAREL, est divisé en un sous-ensemble de développement (82 dialogues, 2 623 tours de parole) et un sous-ensemble de test (100 dialogues, 2 752 tours de parole). La partie de test est elle-même un sous-ensemble de la partie de test de DATCHAACT. La table 7.2 présente la distribution des différentes relations dialogiques dans DATCHAREL.

7.3. Entrainer un analyseur du discours avec peu de données

Dans le but de prédire automatiquement les arbres dialogiques de conversations, je cherche à construire un analyseur du discours capable de prendre en entrée les conversations segmentées en tours de parole et étiquetées en actes de dialogue pour produire en sortie des arbres dialogiques. La prédiction de structures arborescentes à partir d'une séquence est un problème très étudié pour

l'analyse syntaxique de phrases. Il parait donc important de s'appuyer sur ces travaux pour réaliser des analyses profondes du discours conversationnel.

Il existe deux familles d'analyse syntaxique : l'analyse syntagmatique et l'analyse en dépendance. L'*analyse syntagmatique* construit des *arbres syntagmatiques* qui utilisent les nœuds internes de l'arbre pour identifier des sous-phrases avec leur rôle dans la syntaxe (les syntagmes) et les feuilles pour identifier les mots (les constituants). Ce type d'analyse d'une phrase est la méthode « historique » mise en avant par CHOMSKY [Cho57]. L'arbre de la figure 7.1 illustre un tel arbre.

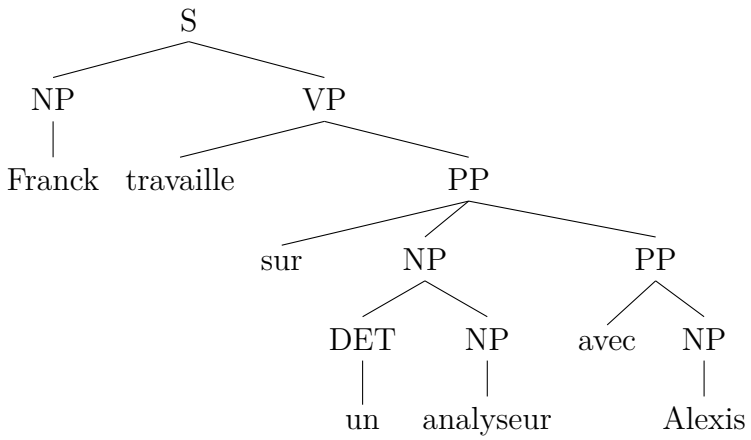


FIGURE 7.1. – Exemple d'analyse syntagmatique

L'analyse en dépendance est elle déjà introduite dans la section 3.2.2 du chapitre 3. Pour rappel, ce type d'analyse repose sur des arbres de dépendance où les nœuds correspondent aux mots de la phrase et les arcs sont les relations syntaxiques entre les mots. La figure 7.2 illustre un tel arbre.

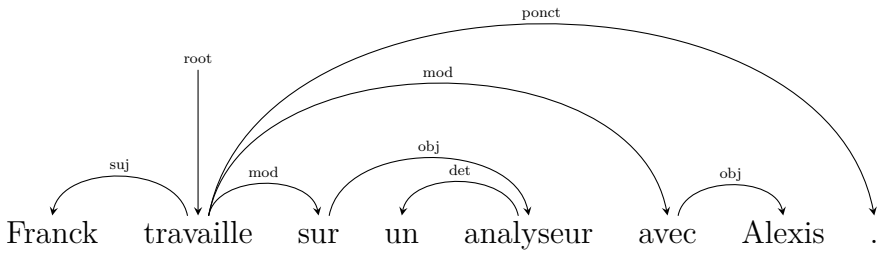


FIGURE 7.2. – Exemple d'analyse en dépendance

Nos données sont annotées en utilisant des relations de dépendance. Il paraitrait alors naturel d'utiliser des analyseurs en dépendance usuellement utilisés tels que ceux basés sur des graphes [MP06] ou en transition [YM03]. Un inconvénient de ces méthodes est qu'elles requièrent une quantité ou une qualité des données suffisante pour pouvoir généraliser.

Afin de donner une solution à la problématique de la quantité de données, je propose d'enrichir le corpus DATCHA en utilisant une approche faiblement supervisée pour construire un premier analyseur du discours. Celui-ci nous permettra par la suite d'enrichir automatiquement le corpus DATCHA.

Dans le cadre de l'analyse syntagmatique, il existe de nombreuses approches — utilisant des grammaires hors-contexte probabilistes (PCFG) — qui permettent de réaliser des apprentissages non-supervisés d'analyseurs syntaxiques. Je vais me fonder sur ces approches-ci pour réaliser un premier analyseur du discours.

Comme je l'ai dit auparavant, notre annotation est sous la forme d'arbres de dépendance. Or, pour utiliser les approches syntagmatiques existantes, je dois d'abord obtenir des arbres syntagmatiques. Il est important de noter que dans mes travaux, la structure utilisée n'est pas la problématique la plus importante. Ce qui m'intéresse davantage est d'être capable de déterminer les différents enjeux et sous-enjeux présents dans le dialogue, c.-à-d. mettre en avant le squelette principal du dialogue. De ce fait, les arbres syntagmatiques me sont tout aussi intéressants que les arbres de dépendance. Dans les sous-sections qui suivent, je décrirai l'approche utilisée pour faire la transformation d'un arbre de dépendance vers un arbre syntagmatique, et inversement.

L'apprentissage de ce premier analyseur du discours se fait en trois étapes :

1. Induire une grammaire hors-contexte (CFG) automatiquement à partir du corpus DATCHAREL qui est annoté avec des arbres en dépendance dialogiques (section 7.3.1) ;
2. Entrainer une PCFG sur le corpus DATCHAACT— uniquement annoté avec des actes de dialogue — en utilisant un processus non-supervisé fondé sur l'algorithme Inside-Outside — algorithme de la famille Expectation-Maximisation qui permet de calculer les probabilités de chaque règle d'une CFG uniquement à partir de séquences d'observables (section 7.3.2) ;
3. Utiliser la PCFG sur des conversations, en appliquant l'algorithme Cocke Younger Kasami (CYK), afin de construire automatiquement zéro, un ou plusieurs arbres discursifs par dialogue (section 7.3.3).

7.3.1. Induire une grammaire hors-contexte

La première étape de mon approche consiste à créer une CFG à partir d'un corpus de référence. Une *grammaire formelle* est un formalisme permettant de décrire un langage formel. Une grammaire G est définie par un 4-uplet (N,Σ,Π,S) où :

— N est un ensemble fini de symboles non-terminaux, c.-à-d. des symboles devant se réécrire en d'autres symboles.

— Σ est un ensemble fini de symboles terminaux, c.-à-d. l'alphabet permettant de construire les mots du langage.

— Π est l'ensemble des règles de productions, c.-à-d. les règles qui permettent de transformer une séquence de symboles en une autre séquence de symbole.

— S est le symbole (non-terminal) de départ.

Un exemple simple de grammaire pourrait être :

$$\begin{aligned} S &\rightarrow xA|yA \\ xA &\rightarrow xz \\ yA &\rightarrow yxA \end{aligned}$$

où S,$A \in N$ et x,y,$z \in \Sigma$. Cette grammaire permet de générer les mots xz et yxz.

Une grammaire hors-contexte est une grammaire formelle avec des contraintes sur ce qui peut se trouver dans la partie gauche des règles de productions. En particulier, la partie gauche ne peut être constituée que d'un unique non-terminal, il n'est pas donc pas possible d'ajouter du contexte en indiquant les symboles qui doivent se trouver à gauche et à droite. Les règles sont alors de la forme :

$$A \rightarrow \alpha$$

où A est un unique symbole non-terminal et α une séquence de symboles terminaux et/ou non-terminaux. Il n'est donc pas possible d'ajouter du « contexte » en imposant la présence d'un autre symbole dans la partie gauche de la règle. Les CFG sont intéressantes d'un point de vue algorithmique car elles permettent l'utilisation d'algorithme polynomiaux en temps, contrairement aux grammaires contextuelles par exemple où les algorithmes sont PSPACE-complets, c.-à-d. que tous les algorithmes connus s'exécutent en temps exponentiels.

Les CFG ne sont pas utilisées telles quelles dans de nombreux algorithmes, dont les algorithmes CYK et Inside-Outside que j'utiliserai dans les sections à venir. En effet, il est nécessaire que les CFG soient sous forme normale de Chomsky. Une CFG est dite en forme normale de Chomsky si toutes les règles de production sont de la forme :

$$\begin{aligned} A &\rightarrow BC && \text{, ou} \\ A &\rightarrow a && \text{, ou} \\ S &\rightarrow \epsilon \end{aligned}$$

où A, B et C sont des symboles non-terminaux, a un symbole terminal, S le symbole de départ et ϵ le mot vide. Toute CFG G peut être transformée en une CFG G' sous forme normale de Chomsky équivalente [1].

Pour notre problème de l'analyse du discours, une première approche permet-

1. Le langage engendré par G est le même que celui engendré par G' ($L(G) = L(G')$)

tant de construire une grammaire hors-contexte pourrait être de demander à des humains d'en déterminer les règles manuellement. Cette approche aurait l'avantage d'être immédiatement interprétable par un humain. Cependant, elle a également l'inconvénient d'être très couteuse en temps, en particulier pour constituer une grammaire entièrement cohérente. Par ailleurs, ce travail a déjà été implicitement réalisé dans le guide d'annotation utilisé pour annoter manuellement le corpus DATCHAREL. Pour ces raisons-là, j'ai décidé d'induire automatiquement la grammaire à partir des arbres discursifs du corpus de développement DATCHAREL.

Plusieurs transformations sont nécessaires afin de pouvoir obtenir une CFG utilisable par les algorithmes CYK et Inside-Outside :

1. Projectiviser les arbres de dépendance ;
2. Transformer les arbres de dépendance en arbres syntagmatiques ;
3. Binariser les arbres syntagmatiques ;
4. Induire une CFG à partir des arbres binaires syntagmatiques.

Projectivisation La toute première étape qu'il est nécessaire de traiter avant de pouvoir obtenir une CFG est de rendre projectifs les arbres discursifs de dépendance de tous les dialogues. Un arbre de dépendance est dit *projectif* si et seulement si tous les arcs de l'arbre sont projectifs. Un arc $w_i \rightarrow w_k$ est dit projectif si et seulement si, pour chaque élément w_j apparaissant entre w_i et w_k, il existe un chemin de w_i à w_j dans le sous-arbre enraciné en w_i. En d'autres termes, un arc est projectif s'il ne croise jamais un autre arc. L'arbre de dépendance illustré dans la figure 7.3 est un arbre projectif, contrairement à celui dans la figure 7.4.

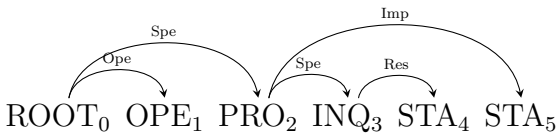


FIGURE 7.3. – Exemple d'un arbre en dépendance discursif projectif tel qu'il peut être trouvé dans le corpus DATCHAREL

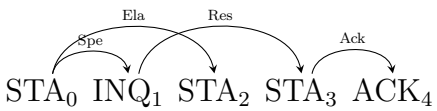


FIGURE 7.4. – Exemple d'un sous-arbre en dépendance discursif non-projectif

Les arbres discursifs en dépendance des dialogues sont très fréquemment non-projectifs, généralement dû à des enchevêtrements de sous-dialogues. Rendre projectif les arbres en dépendance permet d'utiliser la majorité des algorithmes

disponibles permettant de réaliser des analyses syntaxiques (et donc discursives). Afin de conserver certaines informations permettant de partiellement reconstruire la non-projectivité des arbres, les travaux de NIVRE et NILSSON [NN05] sont utilisés. Ces derniers proposent plusieurs approches permettant de projectiviser les arbres en dépendance tout en ajoutant une quantité variable d'informations aux étiquettes des arcs. Ceci permet de pouvoir partiellement réaliser l'opération de « déprojectivisation » par la suite.

NIVRE et NILSSON [NN05] introduisent plusieurs méthodes qui jouent sur un compromis entre l'explosion du nombre des étiquettes et la capacité à reconstituer correctement l'arbre d'origine. Étant donné que je manipule peu de conversations et que je réalise un nombre assez important de manipulation sur les arbres, j'ai opté pour l'approche de projectivisation que les auteurs nomment HEAD, car cette approche réalise uniquement des transformations sur les étiquettes des arcs non-projectifs, contrairement aux autres approches qui modifient potentiellement les étiquettes de tous les arcs.

L'idée de cette approche est de simplement remplacer itérativement les arcs non-projectifs $w_g \xrightarrow{r_d} w_d$ par de nouveaux arcs ayant le même dépendant w_d mais où le gouverneur est modifié. Supposons qu'on ait l'arc $w_p \xrightarrow{r_g} w_g$ alors le gouverneur de w_d devient w_p, construisant ainsi l'arc $w_p \xrightarrow{r_d} w_d$. Durant cette opération, l'étiquette de l'arc est également modifiée pour encoder une information aidant à reconstituer l'arc original lors de la déprojectivisation. L'information ajoutée à l'étiquette de base est simplement l'étiquette r_g de l'arc $w_p \xrightarrow{r_g} w_g$ (où w_g est le dépendant). On obtient alors l'arc $w_p \xrightarrow{r_d/r_g} w_d$. Lors de l'opération de déprojectivisation, cette information permet alors d'obtenir une approximation raisonnable de l'endroit où repositionner l'arc précédemment projectivisé en cherchant, dans le sous-arbre décrivant la sous-séquence $w_p \dots w_d$, la relation r_g. Ceci permet de retrouver le dépendant w_g pour recréer l'arc $w_g \xrightarrow{r_d} w_d$. La déprojectivisation peut provoquer de mauvais rattachements, toutefois, sur des arbres syntaxiques 92% des arcs non-projectifs sont correctement reconstruits. La figure 7.5 permet de montrer le résultat de l'opération de projectivisation sur l'arbre de la figure 7.4.

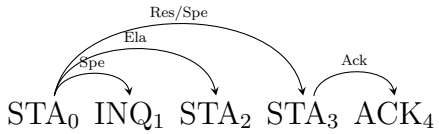


FIGURE 7.5. – Exemple d'un sous-arbre en dépendance discursif projectivisé

Transformation en arbres syntagmatiques Pour les raisons déjà évoquées, la transformation en arbres syntagmatiques est nécessaire afin de pouvoir utiliser l'algorithme Inside-Outside. Cette transformation est triviale et réversible. L'approche utilisée est basée sur celles de GAIFMAN [Gai65] et de HAYS [Hay64]

dans lesquelles les nœuds d'un arbre syntaxique en dépendance, c.-à-d. les éléments lexicaux, sont remplacés par des symboles non-terminaux, généralement leur partie du discours. Chaque nœud non-terminal possède alors un fils additionnel correspondant à l'élément lexical associé. Dans notre cas, les nœuds non-terminaux correspondent aux relations dialogiques et les nœuds terminaux aux actes de dialogue[2]. La figure 7.6 présente un exemple d'arbre syntagmatique ainsi obtenu à partir d'un arbre en dépendance, présenté par la figure 7.3.

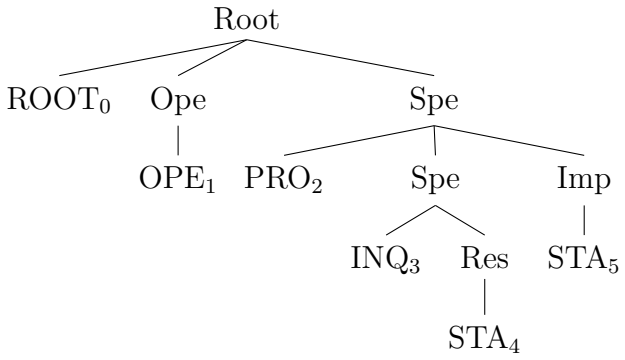


FIGURE 7.6. – Arbre syntagmatique équivalent à l'arbre en dépendance de la figure 7.3

Binarisation des arbres syntagmatiques Les arbres syntagmatiques obtenus jusqu'à présent correspondent à des traces d'application des règles des CFG utilisées pour engendrer les séquences d'actes de dialogue. Les algorithmes CYK et Inside-Outside nécessitent que les CFG soient sous forme normale de Chomsky (FNC), or les arbres obtenus ne correspondent pas à de telles grammaires. De la même manière qu'il est possible de transformer toute CFG en une CFG équivalente sous FNC, il est possible de transformer tout arbre syntagmatique en un arbre binaire équivalent permettant d'engendrer la même séquence. Binariser les arbres syntagmatiques revient à mettre sous FNC les CFG manipulées.

À partir d'un corpus donné, on ne connait pas la CFG qui permettrait d'engendrer correctement toutes les séquences d'actes de dialogue possibles. Il est uniquement possible de faire des suppositions à partir des traces d'arbres syntagmatiques que l'on a à disposition. La méthode la plus immédiate serait de simplement induire à partir de nos arbres les règles de la CFG G puis de la mettre sous FNC (G'). Par exemple, si on a une règle $A \rightarrow BCD$ dans G, dans G' celle-ci deviendrait deux règles :

— $A \rightarrow BA'$

— $A' \rightarrow CD$

2. Cela signifie que le contenu lexical des tours n'est pas pris en compte. Je reviendrai là-dessus plus tard.

Une limite très importante de cette approche est qu'elle considère que le corpus utilisé pour induire la grammaire capture tous les phénomènes pouvant se produire, ce qui n'est très probablement pas le cas quand le corpus est très petit.

Afin de ne pas être limité par la taille de mon corpus, j'utilise une autre approche pour binariser mes arbres. Afin d'avoir davantage de souplesse, je considère qu'il n'est pas envisageable de fixer l'ordre et le nombre de symboles dans la partie droite des règles de production, c.-à-d. l'ordre d'apparition des sous-arbres (fils). La seule chose qu'on souhaite fixer est qu'un nœud donné ne peut avoir qu'un ensemble fixé de fils possibles. L'étape de binarisation nous permet de le faire simplement en transformant les règles en règles récursives gauches. Par exemple, pour la règle $A \rightarrow BCD$ dans G, on obtiendrait alors dans G' les règles :

— $A \rightarrow AD$

— $A \rightarrow AC$

— $A \rightarrow AB$

Un grand intérêt de cette approche est qu'elle apporte une très grande généralisation, en permettant de prendre en compte des exemples n'apparaissant pas dans le corpus [3]. Toutefois, en supprimant toute contrainte d'ordre et de nombre, on a alors un risque important de sur-généralisation en rendant la grammaire beaucoup plus ambigüe, permettant ainsi la production de séquences à partir d'arbres qui n'ont pas de sens. Or obtenir des arbres de bonnes qualités est primordial afin de pouvoir enrichir automatiquement le corpus DATCHA. J'apporterai une solution à ce problème dans l'étape suivante.

La figure 7.7 présente le résultat du processus de binarisation sur l'exemple étudié précédemment de la figure 7.6.

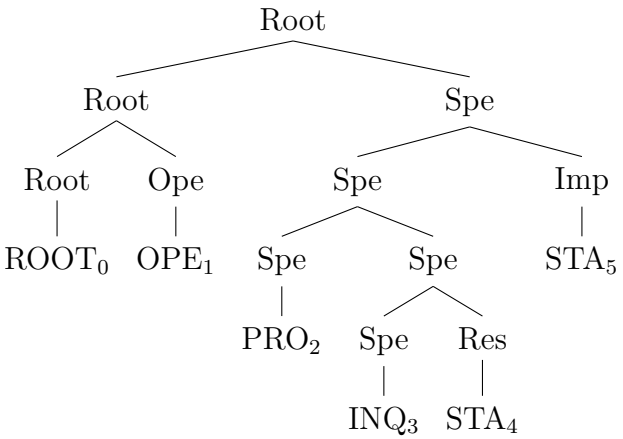


FIGURE 7.7. – Resultat de la binarisation de l'arbre de la figure 7.6

3. Il est important de noter que la CFG sous FNC ainsi obtenue n'est alors pas équivalente à la CFG d'origine.

Induction des CFG Une fois les arbres binaires produits, l'induction d'une CFG est très simple. Il suffit d'associer une règle de production à chaque nœud interne n de l'arbre tel que la partie gauche de la règle soit le symbole non-terminal n et la partie droite soit composé des fils de n : s'il n'y a qu'un fils, c'est un symbole terminal, sinon ce sont deux symboles non-terminaux.

La grammaire Γ_{basic} induite à partir de l'arbre de la figure 7.7 est présentée dans la table 7.3.

(1) Root → Root Spe	(6) Spe → Spe Spe
(2) Root → Root Ope	(7) Spe → Spe Res
(3) Root → ROOT	(8) Spe → PRO
(4) Ope → OPE	(9) Spe → INQ
(5) Spe → Spe Imp	(10) Imp → STA
	(11) Res → STA

TABLE 7.3. – Une grammaire Γ_{basic} induite obtenue à partir de l'arbre de la figure 7.7

Formellement, Γ_{basic} est définie comme suit. Soit $\mathcal{A}$ l'ensemble de tous les actes de dialogue et $\mathcal{R}$ l'ensemble de toutes les relations discursives. Γ_{basic} est un 4-uplet $(N_0,\Sigma_0,\Pi_0,\text{START}_0)$ où :

— $N_0 = \mathcal{R}$;

— $\Sigma_0 = \mathcal{A}$;

— Π_0 est l'ensemble des règles de production, constitué de règles unaires et binaires. Les règles binaires sont de la forme $r_g \rightarrow r_g r_d$, avec r_g et $r_d \in N_0$. De telles règles ajoutent un dépendant ayant l'étiquette r_d à un gouverneur ayant l'étiquette r_g. Les règles unaires sont de la forme $r \rightarrow a$, indiquant qu'un acte de dialogue de type a peut être étiqueté avec r.

— $\text{START}_0 \in N_0$ est le symbole de départ (Root dans notre cas)

On peut noter que Γ_{basic} fait une hypothèse d'indépendance très forte. Par exemple, il n'y a aucune contrainte liant un acte de dialogue et les étiquettes des dépendances qu'il gouverne. En effet, bien qu'il y ait des règles unaires indiquant qu'un acte de dialogue (partie droite) ne peut être produit que s'il y a une relation dialogique particulière, un même acte de dialogue peut être produit par plusieurs types de relations dialogiques. Par exemple, dans la table 7.3, l'acte STA peut être produit aussi bien par une relation Imp (10) que Res (11). Or, il est évident que les structures dialogiques produisant un STA via une relation Imp ou via une relation Res ne sont pas exactement les mêmes. Pourtant, Γ_{basic} va réaliser une sorte d'union des structures dialogiques possibles pour créer un acte de dialogue donné, ne mettant aucune contrainte sur les relations utilisées, la CFG se contentant de s'assurer que la génération de la séquence d'acte de dialogue soit possible.

Ceci a pour conséquence de provoquer une sur-génération et ainsi :

— autoriser des séquences d'actes de dialogue qui n'apparaissent jamais dans les données ;

— utiliser des règles — correspondant à des relations dialogiques — qui ne sont pas censées être utilisées avec certaines séquences d'actes de dialogue.

Afin de limiter le phénomène de sur-génération, des contraintes sont ajoutées à Γ_{basic}. En effet, un moyen simple d'ajouter des contraintes dans les CFG est d'ajouter davantage d'informations contextuelles dans les symboles non-terminaux de la grammaire. Cette caractéristique est utilisée afin de construire la grammaire $\Gamma_{sibling}$. Deux contraintes sont désormais prises en compte par $\Gamma_{sibling}$:

1. Un non-terminal (c.-à-d. une relation dialogique) doit contraindre les terminaux (c.-à-d. les actes de dialogue) qu'il peut étiqueter. Pour cela, il faut que la décision de l'acte de dialogue soit prise dès le début (et non simplement lors de l'utilisation des règles unaires), signifiant que lorsque l'on décide d'utiliser une relation dialogique, cela doit également imposer un acte de dialogue — les non-terminaux deviennent alors des couples relation dialogique-acte de dialogue. Ceci permet alors d'interdire l'utilisation de règles binaires qui n'ont aucun lien avec les actes de dialogue produits par les règles unaires.
2. Il ne faut pas considérer l'existence d'une relation dialogique comme étant indépendante de l'existence d'autres relations dialogiques dans l'arbre. En particulier, on souhaite prendre en compte le fait que la production d'un sous-dialogue est généralement conditionnée par la présence d'un autre sous-dialogue auparavant. Pour faire cela, $\Gamma_{sibling}$ doit donc faire en sorte que l'existence d'un non-terminal r_d soit conditionnée par l'existence d'un frère gauche adjacent r_s dans l'arbre produit. Ceci permet alors de limiter le nombre de sous-arbres autorisés en réduisant la portée de la généralisation qu'apporte la binarisation de l'arbre syntagmatique.

Pour pouvoir définir $\Gamma_{sibling}$, un élément additionnel EOD (pour *End of Dialogue*) est ajouté en fin de chaque séquence d'actes de dialogue. Une relation dialogique technique Eod est alors également ajouté à l'ensemble $\mathcal{R}$. Formellement, $\Gamma_{sibling} = (N_1,\Sigma_1,\Pi_1,\text{START}_1)$ est définie comme suit :

— $N_1 = (\mathcal{R} \cup \{\texttt{Eod}\}) \times (\mathcal{A} \cup \{\texttt{EOD}\}) \times (\mathcal{R} \cup \{\#,\texttt{Eod}\})$. Le symbole $\#$ est utilisé afin d'indiquer qu'il n'y a pas ou plus de dépendants ;

— $\Sigma_1 = \mathcal{A} \cup \{\texttt{EOD}\}$

— Π_1 est l'ensemble des règles de production. Les règles binaires sont de la forme $(r_g,a_g,r_d) \rightarrow (r_g,a_g,r_s)(r_d,a_d,r_n)$ où $(r_g,a_g,r_d),(r_g,a_g,r_s),(r_d,a_d,r_n) \in N_1$ et les règles unaires sont de la forme $(r,a,\#) \rightarrow a$ où $(r,a,\#) \in N_1$ et $a \in \Sigma_1$

— $\text{START}_1 \in N_1$ est le symbole de départ.

(0)	(Root, ROOT, Eod) → (Root, ROOT, Spe) (Eod, EOD, #)
(1a)	(Root, ROOT, Spe) → (Root, ROOT, Ope) (Spe, PRO, Imp)
(1b)	(Root, ROOT, Spe) → (Root, ROOT, Ope) (Spe, PRO, Spe)
(1c)	(Root, ROOT, Spe) → (Root, ROOT, Ope) (Spe, PRO, #)
(2)	(Root, ROOT, Ope) → (Root, ROOT, #) (Ope, OPE, #)
(3)	(Root, ROOT, #) → ROOT
(4)	(Ope, OPE, #) → OPE
(5)	(Spe, PRO, Imp) → (Spe, PRO, Spe) (Imp, STA, #)
(6a)	(Spe, PRO, Spe) → (Spe, PRO, #) (Spe, INQ, Res)
(6b)	(Spe, PRO, Spe) → (Spe, PRO, #) (Spe, INQ, #)
(7)	(Spe, INQ, Res) → (Spe, INQ, #) (Res, STA, #)
(8)	(Spe, PRO, #) → PRO
(9)	(Spe, INQ, #) → INQ
(10)	(Imp, STA, #) → STA
(11)	(Res, STA, #) → STA
(12)	(Eod, EOD, #) → EOD

TABLE 7.4. – Une grammaire $\Gamma_{sibling}$ induite obtenue à partir de l'arbre de la figure 7.7

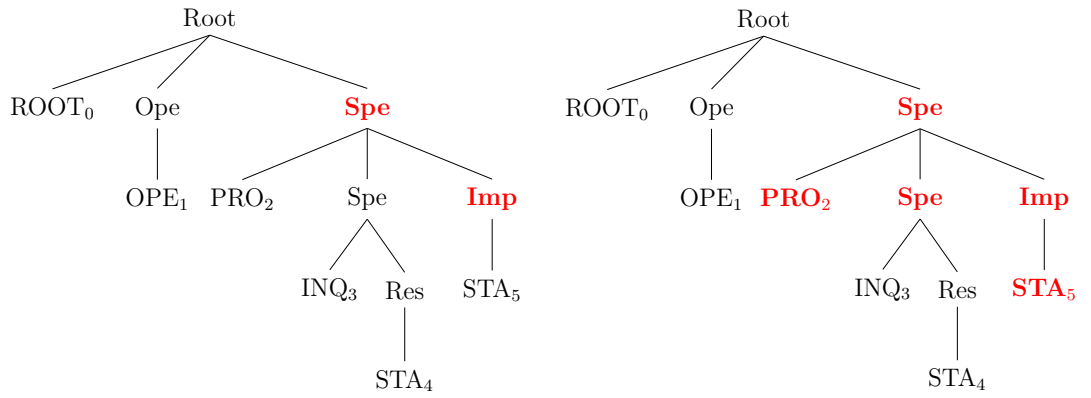
La figure 7.4 présente un exemple de grammaire $\Gamma_{sibling}$ obtenue à partir de l'arbre de la figure 7.7.

La figure 7.8 permet de visualiser (en rouge) les différents « contextes » qui sont capturés par les règles (5) de Γ_{basic} (table 7.3) et $\Gamma_{sibling}$ (table 7.4) sur l'arbre syntagmatique présenté précédemment dans la figure 7.6. On peut noter qu'avec $\Gamma_{sibling}$ le nombre de règles explose par rapport à Γ_{basic}. Ceci est simplement dû au fait que les contraintes de $\Gamma_{sibling}$ imposent davantage de suivre des règles précises en fonction de l'arbre produit.

Afin de ne pas passer un temps trop important sur la définition des CFG, je ne présente que deux types de grammaires avec Γ_{basic} et $\Gamma_{sibling}$. Γ_{basic} permet de présenter la grammaire de « base » construite à partir des arbres syntagmatiques alors que $\Gamma_{sibling}$ est la grammaire sur laquelle j'ai obtenu les meilleurs résultats sur les données à disposition. Toutefois, d'autres variantes peuvent facilement être créées pour ajouter ou supprimer des contraintes aux CFG.

7.3.2. Apprendre une grammaire hors-contexte probabiliste

Les CFG présentées dans la sous-section précédente, que ce soit Γ_{basic} mais aussi, dans une moindre mesure, $\Gamma_{sibling}$ ont le défaut de permettre une surgénération de séquences d'actes de dialogue, et ainsi autoriser la production d'arbres syntagmatiques incohérents. Dans le but de combattre ce phénomène, nous allons exploiter le fait que nous ayons à notre disposition des corpus relati-



(a) Γ_{basic} (b) $\Gamma_{sibling}$

FIGURE 7.8. – Contexte capturé (en rouge) par les règles (5) des tables 7.3 et 7.4 permettants de produire les sous-arbres des séquences PRO$_2$...STA$_5$

vement grand, même s'ils n'ont pas d'annotations arborescentes du discours.

En particulier, le corpus DATCHAACT est intéressant puisqu'il contient des séquences observées d'actes de dialogue. Ceci permet d'ajouter des probabilités aux règles des grammaires Γ_{basic} et $\Gamma_{sibling}$ en utilisant l'algorithme *Inside-Outside*, construisant ainsi une grammaire hors-contexte probabiliste. L'algorithme Inside-Outside est une instanciation de l'algorithme général Espérance-Maximisation (*Expectation-Maximisation* en anglais) (EM) appliqué aux CFG. Considérons une CFG G et un ensemble de mots $S = \{s_1, \ldots, s_n\}$ pouvant être généré par G, l'algorithme Inside-Outside produit une PCFG G' qui maximise la vraisemblance $P(S)$, c.-à-d. qui fait en sorte que les probabilités des règles maximisent la probabilité de chaque mot. Les règles de G et G' sont identiques mise à part le fait que désormais dans G' chaque règle est associé à une probabilité (pouvant être nulle).

Dans notre cas, les mots s_i correspondent aux séquences d'actes de dialogue qui étiquètent chaque tour de parole des conversations du corpus DATCHAACT. Les probabilités des règles calculées par l'algorithme maximisent donc la vraisemblance de DATCHAACT. Il est cependant important de noter que l'algorithme maximise les probabilités des séquences d'actes de dialogue. L'algorithme ne cherche pas à trouver les bons arbres syntagmatiques pour représenter le discours — les arbres corrects étant inconnus — mais seulement les arbres qui permettent de produire les séquences d'actes de dialogue. Toutefois, en maximisant la vraisemblance sur un grand nombre de données, il y a de fortes chances pour que les règles ayant de fortes probabilités correspondent à des patrons de sous-dialogues très fréquents dans les conversations, alors que les règles avec des probabilités faibles, voire nulles, correspondent à des segments de conversations très inhabituels. Bien entendu, ceci dépend nécessairement de la capacité des grammaires à généraliser : plus une grammaire tend à généraliser, c.-à-d. à ne pas imposer

beaucoup de contraintes, plus il sera difficile pour l'algorithme Inside-Outside d'attribuer des probabilités pertinentes aux règles.

La PCFG ainsi obtenue par l'algorithme permet d'associer une probabilité à chaque arbre construit correspondant à la séquence d'actes de dialogue. Plusieurs arbres pouvant être utilisés pour générer une même séquence d'actes de dialogue, ces probabilités sont alors utilisées afin de ne conserver que les arbres ayant de bonnes probabilités, en supposant que ceux-ci décrivent alors au mieux le discours dans le dialogue.

7.3.3. Induire les arbres discursifs à partir d'une PCFG

La PCFG étant produite, il est désormais possible de prédire les arbres discursifs pour certains dialogues. Pour ce faire, il reste deux étapes :

1. Déterminer les règles de la PCFG à utiliser afin d'obtenir la séquence d'actes de dialogue donnée en entrée. Les règles ainsi obtenues correspondent alors à l'arbre syntagmatique associé à la séquence d'actes de dialogue. Cette étape correspond au problème classique de l'analyse syntaxique où un algorithme tel que CYK est utilisé.
2. Transformer l'arbre syntagmatique ainsi obtenu en arbre en dépendance afin de pouvoir réaliser l'évaluation des arbres.

La première étape consiste à utiliser un analyseur, généralement utilisé pour la syntaxe, se basant sur des PCFG. C'est une version probabiliste de l'algorithme CYK [Coc70 ; You67 ; Kas66] qui est utilisé dans les travaux qui suivent. Cet algorithme repose sur la programmation dynamique et requiert que la PCFG soit sous forme normale de Chomsky. L'idée du fonctionnement de CYK est relativement simple. L'algorithme travaille sur une table P et considère toutes les sous-chaines du mot en entrée. Pour chaque sous-chaine, si la sous-chaine peut être générée à partir d'un non-terminal, alors l'algorithme met à vrai la valeur dans P correspondant à la sous-chaine et au non-terminal. L'algorithme commence par les sous-chaines de taille 1, puis de taille 2 jusqu'à la sous-chaine correspondant au mot entier. Le mot peut être généré par la CFG si la case correspondant au mot entier et au symbole de départ est à vrai dans la table P. Plus formellement, pour une PCFG $G = (N,\Sigma,\Pi,S)$, l'algorithme calcule $P[i,j,X] = \mathbb{1}_{w_i \dots w_j \in L(X)}$, où $X \in N$, i et j sont les positions des mots dans la phrase, $w_j,w_i \in \Sigma$ et $L(X)$ est le langage qui peut être engendré à partir de X. La version probabiliste de l'algorithme est identique à l'exception près qu'au lieu d'insérer des booléens dans la table, les probabilités des sous-chaines sont insérées à la place.

La deuxième étape a pour premier objectif de permettre une comparaison avec les données annotées manuellement dans le corpus DATCHAREL. En effet, les arbres discursifs sont annotés en dépendance et donc il est nécessaire d'appliquer la transformation inverse à celle qui a été effectuée dans la section 7.3.1. Un deuxième objectif de cette deuxième étape est de rendre de nouveau les arbres

non-projectifs. Il est important de rendre les arbres produits non-projectifs étant donné que la présence d'enchevêtrements de sous-dialogues est inhérente aux tchats. Il faut donc être capable d'au moins partiellement gérer ce phénomène si on souhaite correctement représenter le discours conversationnel.

7.3.4. Expérimentations

Comme indiqué en introduction du chapitre, en plus de vouloir construire automatiquement des représentations du discours conversationnel, je souhaite également déterminer s'il est intéressant d'enrichir un corpus de petite taille afin de pouvoir utiliser des approches plus gourmandes en données. Dans la section précédente, j'ai décrit une méthodologie se fondant sur des PCFG pour prédire des arbres discursifs. Avant d'utiliser cette méthodologie pour enrichir le corpus DATCHA, il est nécessaire de valider plusieurs points :

1. Est-ce que Γ_{basic} et $\Gamma_{sibling}$ permettent de produire des arbres de qualités raisonnables ?
2. Quelle quantité de données est nécessaire pour construire les grammaires ?
3. Quels aspects du discours conversationnel sont bien modélisés par les grammaires ? Lesquels le sont mal ?

Dans cette section, je vais donc réaliser des expérimentations permettant de donner des réponses à ces questions.

7.3.4.1. Protocole expérimental

Avant de pouvoir donner les résultats des expérimentations, je vais détailler le protocole mis en place permettant de répondre aux différentes questions posées. Par ailleurs, je vais dans un premier temps m'intéresser en détails aux données utilisées dans les expérimentations.

Les données utilisées sont celles provenant du sous-corpus DATCHAREL. Le découpage en sous-ensembles de développement et d'évaluation est conservé. Toutefois, dans le but de m'intéresser à la quantité des données nécessaire pour induire les grammaires, je travaille sur deux versions du sous-ensemble de développement :

— DATCHAREL82 correspondant à l'entièreté du sous-ensemble de développement ;

— DATCHAREL25 constitué de seulement 25 conversations (798 tours de parole). Cette version du corpus est importante car elle correspond à la quantité de donnée qui m'était initialement disponible. Les grammaires ont donc été originellement conçues pour des corpus de cette taille.

À partir de DATCHAREL25 et DATCHAREL82, il est possible d'induire les CFG en appliquant la méthodologie décrite dans la section 7.3.1. Pour rappel, en utilisant les CFG, il n'est pas possible d'utiliser le contenu des tours de parole, chaque

tour de parole est alors représenté par son acte de dialogue. Cependant, afin de conserver des informations sur les scripteurs de chaque tour de parole, des étiquettes distinctes sont créées pour les actes de dialogue par type de scripteurs (c.-à-d. des actes de dialogue différents pour les téléconseillers et les clients). En effet, les deux types de scripteurs ayant des rôles très différents dans le discours, il parait important de différencier leurs actes de dialogue. De plus, cette différenciation permet de créer davantage de séquences possibles et de permettre ainsi d'avoir des CFG plus précises.

CFG	Nombre de règles binaires	Nombre de règles unaires
$\Gamma^{(25)}_{basic}$	119	103
$\Gamma^{(25)}_{sibling}$	1703	104
$\Gamma^{(82)}_{basic}$	204	173
$\Gamma^{(82)}_{sibling}$	5357	174

TABLE 7.5. – Tailles des CFG Γ_{basic} et $\Gamma_{sibling}$ extraites depuis DATCHAREL

La table 7.5 présente quelques informations sur la taille des CFG ainsi obtenues. On peut y trouver quatre CFG :

— $\Gamma^{(25)}_{basic}$ et $\Gamma^{(25)}_{sibling}$ induites à partir de DATCHAREL25 ;

— $\Gamma^{(82)}_{basic}$ et $\Gamma^{(82)}_{sibling}$ induites à partir de DATCHAREL82.

Dans cette table, on peut bien y voir que le nombre de règles produites par les grammaires $\Gamma_{sibling}$ sont bien plus importantes que celles produites par Γ_{basic}. En particulier, la taille du corpus a une influence plutôt grande sur la taille de la grammaire. Il sera intéressant de voir si le fait d'avoir un nombre beaucoup plus important de règles permet de mieux modéliser le discours, ou si au contraire provoque un phénomène de surgénération.

À partir des CFG, je souhaite désormais obtenir les PCFG. Pour cela j'utilise l'algorithme Inside-Outside à l'aide d'une implémentation préexistante de l'algorithme[4]. Afin de pouvoir entrainer la PCFG, l'algorithme est appliqué sur le sous-ensemble d'entrainement du corpus DATCHAACT. Dans la suite de ce chapitre, je nommerai les PCFG Γ_{basic} et $\Gamma_{sibling}$ afin de ne pas alourdir les notations (leurs versions non probabilistes n'étant plus utilisées).

Afin de réaliser l'analyse discursive à partir des PCFG, j'utilise l'implémentation du *Natural Language Toolkit* [LB02] de l'algorithme CYK.

Afin d'évaluer la qualité d'une PCFG G, deux aspects sont considérés :

1. la capacité de G à générer les séquences d'actes de dialogue qui constituent le corpus ;

4. https ://github.com/jgontrum/PCFG-EM

2. la qualité de l'arbre discursif que G associe à une séquence d'actes de dialogue.

Le premier point est mesuré par la *couverture* de la grammaire. Cette métrique est définie comme suit :

$$\text{couverture} = \frac{\text{Nombre de séquences générées par } G}{\text{Nombre total de séquences}}$$

L'intérêt majeur de cette métrique est qu'elle permet de juger si une grammaire impose trop de contraintes. En effet, un résultat proche de zéro indiquerait que très peu de séquences ont pu être générées par la grammaire. Un tel résultat révèlerait un probable sur-apprentissage où la grammaire serait uniquement capable de générer les séquences rencontrées dans le corpus de développement.

Le second point est lui mesuré par des métriques usuellement utilisées dans les domaines de l'analyse syntaxiques sur des arbres de dépendance : le score de rattachements non-étiquetés (*Unlabeled Attachement Score* en anglais) (UAS) et le score de rattachements étiquetés (*Labeled Attachement Score* en anglais) (LAS). Étant donné que ces métriques n'ont de sens que sur des arbres de dépendance, il est nécessaire de transformer les arbres syntagmatiques prédits automatiquement en arbre de dépendance. Ces métriques sont définies comme suit :

$$\text{UAS} = \frac{\text{Nombre de rattachements corrects}}{\text{Nombre total de tours de parole}}$$

$$\text{LAS} = \frac{\text{Nombre de rattachements corrects avec étiquette correcte}}{\text{Nombre total de tours de parole}}$$

Le UAS permet de déterminer si les arbres discursifs obtenus lient entre eux les bons tours de parole. Cette métrique se concentre donc sur la forme des structures obtenues. Le LAS regarde toujours si les tours de parole sont correctement reliés entre eux, mais prend également en compte les étiquettes des liens entre tours de parole. Le LAS est toujours inférieur ou égal au UAS.

En complément du UAS et du LAS qui évaluent la structure discursive de manière générale, je cherche également à évaluer les prédictions obtenues pour chaque type de relation discursive. Pour ce faire, les métriques de précision et de rappel sont utilisées.

Lors de l'évaluation, deux sous-ensembles différents sont considérés :

— TestComplet correspondant au sous-ensemble d'évaluation complet du corpus DatchaRel, constitué de 100 conversations. L'évaluation sur ce sous-ensemble permet de mesurer la couverture et de voir l'influence qu'elle peut avoir sur les scores de UAS et LAS. Ce sous-ensemble aura donc pour effet de pénaliser les scores obtenus par une grammaire ayant une faible couverture.

— TESTCOUVERTS constitué de 63 conversations issues du sous-ensemble d'évaluation du corpus DATCHAREL. Ces 63 conversations sont celles qui sont couvertes par les deux grammaires Γ_{basic} et $\Gamma_{sibling}$ permettant ainsi de pouvoir comparer la qualité des arbres produits. Dans les faits, ceci correspond aux conversations couvertes par $\Gamma^{(25)}_{sibling}$, étant donné que c'est la grammaire imposant le plus de contraintes.

7.3.4.2. Résultats

PCFG	Couverture (en %)	UAS (en %)	LAS (en %)
TESTCOMPLET : 100 dialogues – ensemble de test de DATCHAREL			
$\Gamma^{(25)}_{basic}$	98	39.9	25.1
$\Gamma^{(25)}_{sibling}$	63	35.5	30.1
$\Gamma^{(82)}_{basic}$	100	38.9	30.2
$\Gamma^{(82)}_{sibling}$	87	48.9	35.8
TESTCOUVERTS : 63 dialogues couverts par Γ_{basic} et $\Gamma_{sibling}$			
$\Gamma^{(25)}_{basic}$	100	36.6	28.0
$\Gamma^{(25)}_{sibling}$	100	59.0	49.9
$\Gamma^{(82)}_{basic}$	100	39.1	30.4
$\Gamma^{(82)}_{sibling}$	100	58.4	42.3

TABLE 7.6. – Évaluation des arbres discursifs inférés par les PCFG sur les ensembles de test

La table 7.6 présente les scores obtenus par les analyseurs discursifs basés sur Γ_{basic} et $\Gamma_{sibling}$ sur les conversations du corpus DATCHAREL. La partie du haut de la table s'intéresse aux performances obtenues sur l'ensemble des conversations TESTCOMPLET alors que la partie du bas s'évalue sur TESTCOUVERTS.

Les scores de couverture permettent de constater que les grammaires Γ_{basic} couvrent presque l'entièreté des conversations alors que les grammaires $\Gamma_{sibling}$ n'en couvrent qu'une partie : 63% avec un corpus de développement de taille 25 et 87% avec un corpus de développement de taille 82. Ceci était un comportement plutôt attendu étant donné que $\Gamma_{sibling}$ impose de nombreuses contraintes afin de pouvoir générer les séquences d'actes de dialogue, contrairement à Γ_{basic} qui n'en impose presque pas. Il était également attendu que plus la taille du corpus de développement était élevée, plus la couverture de la grammaire le serait également.

En observant cette fois-ci les scores de UAS et de LAS, on peut voir que $\Gamma_{sibling}$ permet d'obtenir de meilleurs arbres discursifs que Γ_{basic}. Bien que $\Gamma^{(25)}_{basic}$

et $\Gamma^{(82)}_{basic}$ obtiennent des scores de UAS supérieur à $\Gamma^{(25)}_{sibling}$ d'environ 5 points sur TestComplet, ceci est uniquement dû au fait que $\Gamma^{(25)}_{sibling}$ ne couvre pas toutes les conversations. Ceci est confirmé par le fait que $\Gamma^{(82)}_{sibling}$, qui couvre beaucoup plus de conversations, obtient les meilleurs scores sur TestComplet avec 10 points de plus que les deux grammaires Γ_{basic}. Lorsque l'on compare cette fois-ci les scores sur TestCouverts, on constate que $\Gamma^{(25)}_{sibling}$ obtient un score de UAS bien supérieur avec 20 points de plus que les grammaires Γ_{basic} pour arriver à presque 60%. Sur le score de LAS, ce phénomène est encore plus visible, $\Gamma^{(25)}_{sibling}$ ayant un score de 50% contre les 28% de Γ_{basic}.

En se focalisant sur la taille des corpus de développement utilisés, on peut constater que pour les grammaires $\Gamma_{sibling}$ le fait de passer de 25 à 82 conversations ne permet pas d'améliorer le score de UAS et fait même perdre 8 points de LAS. Ces résultats permettent de voir que le fait d'augmenter la taille du corpus sur lequel les grammaires sont apprises permet certes d'améliorer la couverture de celles-ci, mais ne permet pas d'automatiquement améliorer la qualité des arbres produits. Ce comportement peut être expliqué par le fait que les bénéfices des contraintes ajoutées dans la grammaire $\Gamma_{sibling}$ diminuent avec l'augmentation lorsque le corpus de développement devient trop grand. En effet, le nombre de règles binaires augmentant fortement (voir la table 7.5), on risque d'introduire beaucoup de possibilités pour produire une même séquence d'actes de dialogue, ce qui revient à en partie annuler les contraintes de contexte qui ont été ajoutées.

Il semblerait que l'augmentation de la taille du corpus n'est pas synonyme d'amélioration des résultats. La figure 7.9 permet de visualiser le comportement des grammaires en fonction de la taille du corpus de développement lors de l'induction des CFG. Il est important de noter que les scores de UAS et LAS sont calculés ici à chaque fois uniquement sur les conversations couvertes pour une taille de corpus de développement donnée. Ces courbes ne permettent donc pas de comparer directement la qualité des arbres produits sur un corpus fixé. En revanche, étant donné que les grammaires ne peuvent produire des arbres que pour des séquences d'actes de dialogue qu'elles peuvent générer, ces courbes peuvent nous permettre d'identifier le moment à partir duquel les grammaires deviennent trop ambigües et détériorant la qualité des arbres produits. Les courbes en pointillés correspondent aux scores obtenus par $\Gamma_{sibling}$ et les courbes pleines aux scores de Γ_{basic}.

En s'intéressant tout d'abord aux scores de couverture (bleu et rouge), on peut constater que, sans surprise, la couverture croît à chaque fois que la taille du corpus de développement augmente. Γ_{basic} atteint très rapidement un score proche des 100% avec uniquement 10 conversations. Dans le cas de $\Gamma_{sibling}$, cette croissance est plus lente et atteint au maximum une couverture de 87% avec une croissance plus lente à partir de 25 conversations et une couverture de 63%. Ces résultats correspondent à ce qui avait été observé dans la table 7.6.

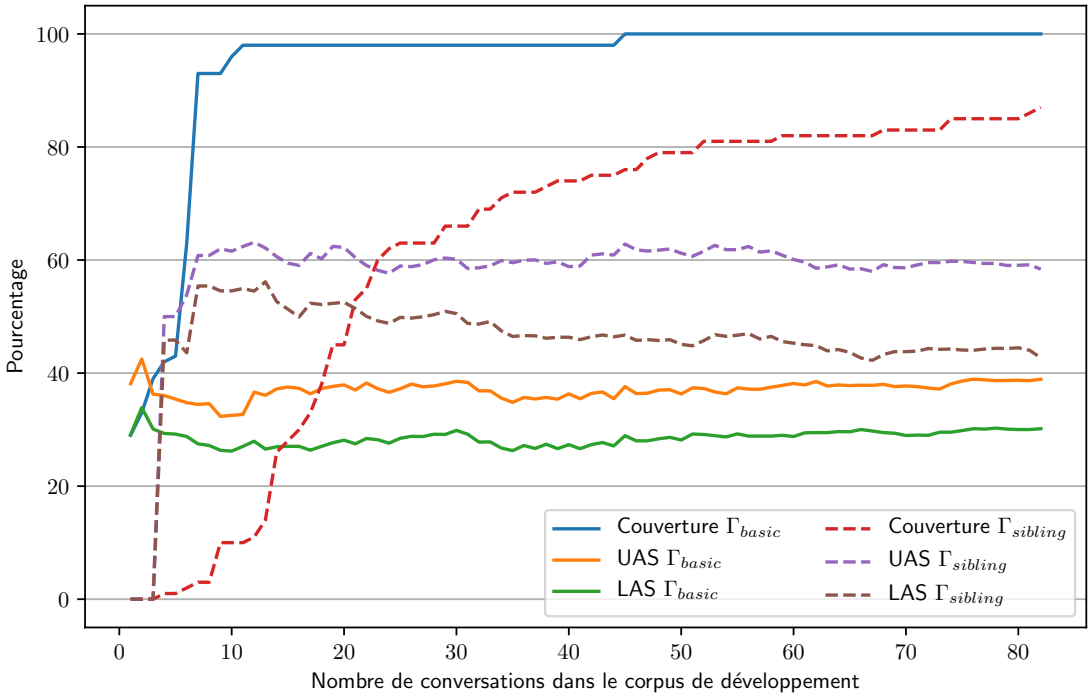


FIGURE 7.9. – Évolution de la couverture et des scores de rattachements en fonction de la taille du corpus de développement. Les courbes continues (bleue, orange et verte) correspondent aux scores obtenues par les PCFG Γ_{basic}. Les courbes en pointillés (rouge, violet et marron) correspondent aux scores attribués aux PCFG $\Gamma_{sibling}$.

En observant cette fois-ci les scores de rattachements, on peut constater des phénomènes différents en fonction de la CFG. Avec Γ_{basic}, les scores de UAS et LAS restent assez stables quelle que soit la taille du corpus de développement. Avec $\Gamma_{sibling}$, le comportement diffère en fonction de la métrique et on observe un phénomène étrange. En considérant le score de UAS, on constate que le score reste stable autour des 60%, contrairement au score de LAS qui est lui impacté par la taille du corpus. En effet, plus la couverture est élevée, plus le score de LAS baisse, démarrant autour des 50% avec un petit corpus pour finir autour des 42% avec le corpus de développement complet. Ceci n'est pas nécessairement attendu, on s'attendrait davantage à ce que les scores restent au moins stables et que UAS et LAS aient des évolutions semblables. Les résultats obtenus permettent de mettre en évidence deux phénomènes :

— le fait d'ajouter des conversations pour induire la CFG détériore constamment la qualité des arbres produits ;

— la qualité de la structure générale des arbres produits est peu influencée, la détérioration concernant surtout les étiquettes prédites des relations.

Ces deux points semblent donc indiquer qu'en ajoutant des conversations, la

grammaire $\Gamma_{sibling}$ devient plus ambigüe. Toutefois, cette ambigüité porte surtout sur les types de relations à utiliser pour générer une séquence d'actes de dialogue.

De manière générale, il semblerait qu'avec les grammaires un choix est à faire entre qualité des arbres produits et couverture de toutes les conversations. Toutefois, ce compromis peut être utilisé à notre avantage afin de ne considérer que des conversations similaires à celles se trouvant dans notre corpus de référence. En effet, on peut partir du principe que si $\Gamma_{sibling}$ n'est pas capable de produire un arbre discursif pour une séquence d'actes de dialogue donnée, alors cela signifie que cette séquence ne ressemble pas à une séquence se trouvant dans le corpus de développement. En outre, un de mes objectifs est d'utiliser les grammaires afin d'enrichir un corpus avec des arbres produits automatiquement. De ce fait, il peut paraitre intéressant de n'enrichir qu'avec les conversations ayant des structures similaires à celles se trouvant dans le corpus de développement DATCHAREL.

7.4. Utiliser davantage de données à l'aide d'annotations automatiques

Une limite importante des PCFG décrites dans la section précédente est qu'elles délimitent strictement les séquences qu'elles peuvent générer, qui sont les langages des grammaires. Les grammaires étant construites à partir d'un petit ensemble de conversations, il est possible de rencontrer des séquences d'actes de dialogue ne pouvant être engendrées par les grammaires induites. Or, parmi les séquences d'actes de dialogue qui ne peuvent pas être engendrées, il est très probable d'y trouver des sous-séquences que la grammaire est capable d'engendrer. Il est donc clair que les grammaires ne sont pas l'approche à considérer dans le but de généraliser la prédiction des arbres discursifs à l'ensemble des conversations.

Toutefois, les grammaires peuvent être un outil intéressant afin d'augmenter automatiquement la quantité de données annotées disponible. Ceci permettrait alors d'entrainer des analyseurs en dépendance usuellement utilisés sur des quantités relativement importantes de données dans le domaine de l'analyse syntaxique. Cette approche s'appuie donc sur l'hypothèse qu'avoir une quantité importante de données bruitées permet d'obtenir un meilleur modèle d'analyse du discours que si on se fondait sur peu de données non bruitées.

À l'opposé, on peut également considérer qu'il est plus important d'améliorer la qualité des représentations données en entrée pour entrainer les modèles d'analyse du discours. Ceci correspond alors à essayer de compenser le peu de données par des représentations (celles des tours de parole par exemple) qui permettent de modéliser le discours conversationnel.

Afin de valider, ou pas, ces deux hypothèses, deux approches sont confrontées dans cette section :

1. Utiliser les PCFG afin d'enrichir automatiquement le corpus DATCHAACT avec des arbres discursifs, permettant ensuite d'entrainer un analyseur en transition sur une quantité importante de données. Cette approche sera décrite dans la section 7.4.1.
2. Utiliser des plongements de tours de parole qui permettent de modéliser le discours conversationnel tels que les plongements SKIP-ACT. Ces plongements seront donnés en entrée d'un analyseur en transition. Cette approche sera décrite dans la section 7.4.2.

Ces deux approches seront également combinées afin de déterminer si elles peuvent être complémentaires.

7.4.1. Résoudre les problèmes de couverture des PCFG

Dans le but d'enrichir le corpus DATCHAACT de manière automatique, j'utilise sur celui-ci les PCFG précédemment construite. Pour rappel, le corpus DATCHAACT est uniquement annoté en actes de dialogue, ce qui permet donc d'utiliser sans traitements additionnels les différentes PCFG. Le corpus obtenu à l'issue de l'utilisation de la PCFG (et du passage à des arbres en dépendance) sur DATCHAACT est nommé DATCHAACT+R. Étant donné que les grammaires ne sont pas capables d'engendrer toutes les séquences imaginables d'actes de dialogue, il est attendu qu'uniquement un sous-ensemble de DATCHAACT soit annoté. La couverture de la plus faible étant de 63% ($\Gamma^{(25)}_{sibling}$), on peut s'attendre à obtenir un taux de couverture semblable sur DATCHAACT. Un autre aspect à prendre en compte est que le corpus DATCHAACT+R aura un grand nombre d'erreurs d'annotation, ce nombre dépendant de la PCFG utilisée.

À partir de DATCHAACT+R, l'idée est alors d'utiliser un analyseur en dépendance en transition. Un tel analyseur offre deux avantages importants par rapport aux PCFG :

— il permet de produire une analyse de (presque) toutes conversations étant donné qu'il ne dépend pas d'une grammaire formelle ;

— il est capable de prendre en compte les informations lexicales qui constituent les tours de parole à l'aide de plongements de phrases.

Pour le premier point, l'hypothèse qui est faite est que l'analyseur en transition sera capable d'obtenir des performances au moins comparables aux PCFG tout en permettant de couvrir l'ensemble des conversations. L'idée est que lorsqu'une PCFG ne parvient pas à engendrer une séquence d'actes de dialogue, cela est probablement uniquement dû à certains éléments de la séquence qui apparaissent à des positions non admises par la grammaire. Les autres sous-séquences auraient

probablement pu être engendrées par la grammaire si ces éléments inhabituels n'étaient pas présents.

Le second point est intéressant car dans les grammaires utilisées, je me base uniquement sur les actes de dialogue (et le scripteur) pour représenter les tours de parole. Or, même si les actes de dialogue portent une information importante sur la fonction de communication du tour et les intentions du scripteur, ils regroupent ensemble une très grande variété de tours de parole qui peuvent avoir des rôles variés avec d'autres tours de parole, et donc induire des relations discursives différentes avec ces derniers. Par exemple, si on considère le sous-dialogue suivant :

— **CL**: Oui le décodeur est redémarré (STA)

— **TC**: Qu'est-ce qui s'affiche sur votre écran ? (INQ)

— **CL**: La TV d'Orange (STA)

— **TC**: Pouvez-vous tout d'abord mettre quelques chaines pour vérifier s'il fonctionne ? (INQ)

— **CL**: Ça fonctionne toujours un peu par saccade par moment (STA)

Dans ce sous-dialogue, on peut y voir deux tours de paroles avec pour acte de dialogue `INQ`, étant donné que ce sont des questions. Toutefois, leur rôle vis-à-vis des autres tours de dialogue n'est pas le même. Dans le cas de la première question, celle-ci est simplement une spécification (`Spe`) du tour précédent. Dans la deuxième question, l'agent demande au client de réaliser une action externe au dialogue (`Imp`) après avoir eu la réponse du client. On peut donc constater qu'en utilisant uniquement l'information sur l'acte du dialogue, on perd certaines informations lexicales qui sont indispensables pour correctement identifier le rôle d'un tour vis-à-vis des autres tours.

La prise en compte du lexique peut donc aider à correctement déterminer les relations dialogiques dans certaines configurations où les actes de dialogue ne suffisent pas.

7.4.2. Prise en compte du lexique par un analyseur en dépendance

L'utilisation d'un analyseur en transition permet d'utiliser des plongements de phrases, et ainsi prendre en compte le lexique dans la prise des décisions de l'analyseur. Toutefois, les plongements de phrases sont généralement utilisés dans le contexte de documents qui ne sont pas des dialogues. On a pu voir dans le chapitre 6 (Représentation vectorielle des tours de parole) que les plongements qui ne prennent pas en compte le discours conversationnel dans leur construction ne sont pas adaptés pour correctement répondre à des tâches d'analyse de surface du discours.

Une question qui peut alors se poser est de savoir si ceci est également vrai dans le cadre d'une analyse profonde du discours conversationnel. Jusqu'à présent, je me suis surtout posé la question de savoir s'il était possible d'améliorer la prédiction d'arbres discursifs en enrichissant le petit corpus d'entrainement avec des exemples bruités. Une autre approche possible est d'améliorer la qualité des représentations du dialogue données en entrée de l'analyseur, en particulier en utilisant des plongements de phrases adaptés à la tâche.

Une manière d'obtenir des plongements adaptés à la tâche est de laisser le modèle d'analyse en dépendance apprendre lui-même les plongements de tours de parole. Un défaut majeur de cette approche est qu'elle nécessite un corpus d'apprentissage très volumineux. De plus, ces approches imposent l'utilisation d'architectures beaucoup plus complexes (et donc beaucoup plus couteuse en ressources de calcul) due à la nature même des données utilisées (un tour de parole est composé de plusieurs mots, et il existe un nombre extrêmement important de combinaisons possibles de mots où l'ordre est important).

Par conséquent, j'ai fait le choix d'utiliser des plongements de phrase pré-entrainés. En outre, l'utilisation de ceux-ci permet d'évaluer si ces plongements modélisent suffisamment d'information afin de pouvoir réaliser des analyses profondes du discours. Dans mes travaux, je m'intéresse à deux types de plongements : des plongements de phrase issus de la moyenne des plongements de mots FASTTEXT [Boj+17] et les plongements SKIP-ACT introduits dans le chapitre 6.

Les premiers permettent d'évaluer si la prise en compte du lexique permet de résoudre le problème de tours ayant les mêmes actes de dialogue ayant des rôles dialogiques différents vis-à-vis des autres tours. Si c'est le cas, on peut alors espérer améliorer la qualité des arbres discursifs produits, en particulier au niveau des types des relations discursives. En effet, on a pu constater dans la section 7.3 que c'est à ce niveau-là que la qualité se dégrade le plus lorsque les grammaires deviennent trop permissives. Le fait d'utiliser ces plongements-ci plutôt que des plongements SKIP-THOUGHT se justifie par le fait qu'ils sont plus faciles à mettre en place et constituent donc une bonne base de référence. En outre, on a pu constater dans le chapitre 6 qu'ils modélisent mieux le discours conversationnel que les plongements SKIP-THOUGHT.

Les plongements SKIP-ACT sont intéressants car ils sont construits de manière à explicitement prendre en compte le discours conversationnel. De ce fait, ils peuvent être vus comme une étape de prétraitement qui a pour but d'extraire les caractéristiques propres au discours conversationnel se trouvant dans les tours de parole. Ceci permet de simplifier la tâche de l'algorithme d'apprentissage en lui mettant en avant des caractéristiques pouvant être intéressantes pour sa tâche.

7.4.3. Expérimentations

Dans les deux sections précédentes, j'ai décrit deux approches — potentiellement complémentaires — qui ont pour but de construire un analyseur en dépendance du discours conversationnel à partir de peu de données.

L'objectif des expérimentations est alors de valider, ou pas, l'utilité de ces approches. La première approche consiste à enrichir le corpus DATCHAREL à l'aide des PCFG décrites dans la section 7.3. La deuxième approche s'intéresse aux représentations des dialogues donnés en entrée de l'analyseur, le but étant alors d'utiliser des représentations adaptées à l'analyse du discours.

Ces expérimentations me permettront de donner des débuts de réponses aux questions suivantes :

1. Est-ce qu'un corpus avec des annotations très bruitées est tout de même intéressant pour réaliser de l'analyse du discours conversationnel ? En d'autres termes, est-il intéressant d'enrichir un petit corpus à l'aide d'annotations automatiques ?
2. Est-ce que le lexique est une caractéristique importante pour répondre à la tâche ? Est-ce que les actes de dialogue, ainsi que leurs contextes de production sont suffisants ?
3. Quelle est l'influence des plongements de phrases sur la modélisation du discours conversationnel par l'analyseur ? Que permettent-ils de mieux modéliser ?

7.4.3.1. Protocole expérimental

Pour toutes les expérimentations qui suivent, j'utilise un analyseur en transition. Le but est alors de couvrir davantage de conversations tout en espérant conserver, voire améliorer, les performances obtenues par les PCFG. L'analyseur en transition est développé en interne dans le laboratoire, se fondant sur les travaux de CHEN et MANNING [CM14]. Les analyseurs en transition sont généralement utilisés pour l'analyse syntaxique. Ils construisent un arbre de dépendance en une seule passe en lisant la phrase de la gauche vers la droite de la séquence de manière gloutonne. L'analyseur commence avec une configuration initiale et à chaque étape de l'algorithme une transition est prédite (créer une relation, passer au mot suivant, terminer l'analyse, etc.). Cette transition permet de passer à une nouvelle configuration menant petit à petit à l'arbre de dépendance final. Les configurations correspondent aux informations accessibles par le classifieur comme les mots à gauche et à droite, les mots qui peuvent encore être liés à d'autres mots et les différentes caractéristiques des mots pris en compte. Le classifieur utilisé par notre analyseur est un simple perceptron multi-couche.

Étant donné que je cherche à faire une analyse du discours conversationnel, l'unité de base n'est plus le mot mais le tour de parole. Les caractéristiques utilisées pour décrire une configuration sont le tour de parole courant mais égale-

ment les tours de parole à sa droite et à sa gauche. Les actes de dialogue, les scripteurs et les relations dialogiques déjà prédites de ces tours sont également présents dans les configurations. Pour davantage de détails, les caractéristiques utilisées pour construire les configurations sont précisément décrites dans l'annexe D.

Les tours de parole, actes de dialogue, scripteurs et relations discursives donnés en entrée dans les configurations sont respectivement représentées par des plongements de tailles 2048, 10, 3 et 20. Comme indiqué dans la section 7.4.2, les plongements de tours de parole sont pré-entrainés et fixés et sont de deux types :

— une moyenne de vecteurs FASTTEXT ;

— des vecteurs SKIP-ACT

Les autres plongements sont eux appris directement par le classifieur lors la phase d'apprentissage.

Étant donné que les meilleurs arbres étaient produits par la PCFG $\Gamma^{(25)}_{sibling}$, cette dernière est utilisée afin d'annoter automatiquement le sous-ensemble d'entrainement du corpus DATCHAACT permettant ainsi d'obtenir le corpus DATCHAACT+R [5]. Celui-ci contient 1 345 conversations avec un total de 37 266 tours de parole.

La table 7.7 présente la distribution des relations dialogiques se trouvant dans le corpus DATCHAACT+R. La distribution dans le corpus DATCHAREL25 est également incluse afin de pouvoir aisément comparer les corpus. On peut constater que les proportions sont relatives semblables, tout en ayant un nombre bien plus important d'exemples pour chaque relation. Il est donc raisonnable de penser que les deux corpus contiennent des conversations similaires, ce qui était attendu étant donné que $\Gamma^{(25)}_{sibling}$ est limitée dans les séquences d'actes de dialogue qu'elle peut générer.

Afin d'entrainer les analyseurs en transition, il est nécessaire d'avoir à la fois un sous-ensemble d'apprentissage et de développement, DATCHAREL82 est de ce fait divisé en deux avec le sous-ensemble d'apprentissage ayant 62 conversations (DATCHAREL62 qui inclue DATCHAREL25) et celui de développement ayant les 20 conversations restantes. Trois configurations d'apprentissage de l'analyseur sont évaluées :

1. Apprentissage uniquement sur DATCHAREL62 (TP-DATCHAREL62). Cette configuration permet de déterminer les scores obtenus par l'analyseur en transition sans avoir réalisé d'enrichissement automatique. Celle-ci permet également de déterminer si les plongements de tours peuvent influencer la qualité des prédictions (en utilisant des plongements FASTTEXT ou SKIP-ACT).
2. Apprentissage uniquement sur DATCHAACT+R (TP-DATCHAACT+R). Cette configuration doit permettre de déterminer les scores obtenus en ayant unique-

5. De premières expériences se fondant sur $\Gamma^{(82)}_{sibling}$ ont été également réalisées mais ont produits de moins bons résultats qu'avec $\Gamma^{(25)}_{sibling}$.

	# DatchaAct+R		# DatchaRel25	
Relation	Distrib.	Prop. (%)	Distrib.	Prop. (%)
Opening	1 811	4,86	39	4,89
Specification	7 160	19,21	130	16,29
Response	10 286	27,60	232	29,07
Imperative	1 142	3,06	15	1,88
Temporisation	589	1,58	18	2,26
Acknowledgement	4 616	12,39	89	11,15
Ending	195	0,52	3	0,36
Closing	1 792	4,81	56	7,02
Opinion	396	1,06	9	1,13
Elaboration	6 670	17,90	143	17,92
Explicit Relation Demand	2 609	7,00	64	8,02

TABLE 7.7. – Distribution des relations dialogiques du corpus DatchaAct+R annoté avec $\Gamma^{(25)}_{sibling}$ comparée à DatchaRel25

ment des conversations avec des annotations bruitées.

3. Apprentissage sur DatchaAct+R combiné à DatchaRel62 (TP-Both). Cette configuration permet de déterminer si le fait d'enrichir un petit corpus permet d'avoir des arbres discursifs de meilleures qualités que si l'entrainement était uniquement réalisé sur le petit corpus.

À chaque fois, les deux variantes de plongements de tour de parole sont évaluées.

Les métriques utilisées afin d'évaluer les arbres discursifs inférés sont identiques à celles utilisées dans la section 7.3.4, c.-à-d. la couverture, le UAS, le LAS. De plus, le rappel et la précision de chaque étiquette sont également calculés afin de pouvoir précisément déterminer les relations qui sont mieux ou moins bien prédites qu'avec les PCFG. Ces différents scores sont calculés à partir d'une moyenne de 5 apprentissages de chaque modèle issus de l'analyseur en transition.

Afin de pouvoir comparer les résultats avec ceux obtenus par les PCFG, les deux sous-ensembles d'évaluation TestComplet et TestCouverts sont de nouveaux utilisés. Pour rappel, TestComplet contient les 100 conversations de l'ensemble de test de DatchaRel et TestCouverts contient les 63 conversations pour lesquelles $\Gamma^{(25)}_{sibling}$ est capable d'engendrer les séquences d'actes de dialogue.

7.4.3.2. Résultats

La table 7.8 présente les résultats obtenus par les différents modèles issus de l'analyseur en transition. Les résultats obtenus par $\Gamma^{(25)}_{sibling}$ y sont reproduits afin

Analyseur	Plongements	UAS (en %)	LAS (en %)
TESTCOMPLET : 100 dialogues – ensemble de test de DATCHAREL			
$\Gamma^{(25)}_{sibling}$	–	35,5	30,1
TP-DATCHAREL62	Moyenne	58,0	41,2
TP-DATCHAACT+R	Moyenne	60,4	49,7
TP-Both	Moyenne	59,8	49,4
TP-DATCHAREL62	Skip-Act	67,2	55,3
TP-DATCHAACT+R	Skip-Act	61,5	52,3
TP-Both	Skip-Act	65,1	56,3
TESTCOUVERTS : 63 dialogues couverts par $\Gamma^{(25)}_{sibling}$			
$\Gamma^{(25)}_{sibling}$	–	59,0	49,9
TP-DATCHAREL62	Moyenne	60,5	43,8
TP-DATCHAACT+R	Moyenne	61,9	51,8
TP-Both	Moyenne	61,7	51,8
TP-DATCHAREL62	Skip-Act	69,2	57,7
TP-DATCHAACT+R	Skip-Act	62,5	53,9
TP-Both	Skip-Act	66,9	58,3

TABLE 7.8. – Évaluation des arbres discursifs inférés par l'ensemble des analyseurs discursifs sur les ensembles de test

d'avoir des points de comparaison. Les scores de couverture n'y sont pas indiqués car ils sont de 100% pour tous les nouveaux modèles.

En s'intéressant dans un premier temps aux résultats sur TESTCOUVERTS, on observe très clairement que les meilleurs résultats sont obtenus en utilisant les plongements SKIP-ACT par les analyseurs TP-DATCHAREL62 (69,2% UAS et 57,7% LAS) et TP-Both (66,9% UAS et 58,3% LAS). En outre, ces résultats sont bien supérieurs aux scores obtenus par $\Gamma^{(25)}_{sibling}$ (59,0% UAS et 49,9% LAS), mais également aux scores obtenus par les variantes d'analyseurs en transition se fondant sur les plongements MOYENNE. On observe donc que les plongements SKIP-ACT sont bien meilleurs que les plongements MOYENNE pour cette tâche, probablement dû à la prise en compte du discours conversationnel dans les premiers.

Dans le détail, il est intéressant de constater une différence majeure de comportement de l'analyseur TP-DATCHAREL62 au niveau du score de LAS. En effet, avec des plongements MOYENNE ce score est de 43,8% (et donc inférieur au score de $\Gamma^{(25)}_{sibling}$) alors qu'il est de 57,7% avec les plongements SKIP-ACT. Ces résultats montrent qu'un corpus ne contenant que 62 conversations n'est pas suffisamment grand si les interactions entre tours de parole ne sont pas prises en compte

par les plongements de phrases.

Les résultats obtenus par TP-DATCHAACT+R avec des plongements MOYENNE confirment cela étant donné que celui-ci parvient à battre de 2 points la PCFG. Cette amélioration peut sembler surprenante étant donné qu'on pourrait s'attendre à ce que l'analyseur en transition reproduisent exactement les mêmes arbres que ceux produits par $\Gamma^{(25)}_{sibling}$. Néanmoins, la prise en compte du contenu lexical des tours par l'analyseur en transition permet probablement de distinguer des tours ayant des actes de dialogue identiques, et ainsi légèrement améliorer la qualité des arbres produits.

Toutefois, les résultats obtenus avec l'analyseur TP-DATCHAACT+R utilisant les vecteurs SKIP-ACT (62,5% UAS et 53,9% LAS) permettent de constater qu'un enrichissement automatique n'est pas complémentaire avec l'utilisation de ces vecteurs. En effet, ces résultats sont moins bons que les résultats obtenus par les deux analyseurs TP-DATCHAREL62 et TP-Both qui ont accès au corpus annoté manuellement DATCHAREL62. Ceci est potentiellement dû au bruit introduit par les annotations automatiques qui pourrait entrer en conflit avec les plongements SKIP-ACT.

En comparant ces mêmes scores mais sur TESTCOMPLET, on observe que contrairement à la PCFG qui s'effondre dû à sa faible couverture, les autres analyseurs obtiennent des performances proches de celles obtenues sur TESTCOUVERTS, avec une légère baisse de 1 ou 2 points de UAS et de LAS. Ceci est particulièrement intéressant car cela montre qu'on est capable de prédire pour l'ensemble des conversations des arbres discursifs de qualités similaires à ceux produits pour les conversations couvertes par $\Gamma^{(25)}_{sibling}$.

Ces différents résultats nous permettent de conclure que les deux approches individuellement ont un intérêt et permettent de prédire sur l'ensemble des conversations des arbres discursifs de qualités au moins aussi bonnes que ceux produits par $\Gamma^{(25)}_{sibling}$. Toutefois, les deux approches entrent en conflits lorsqu'elles sont combinées. L'approche qui consiste à choisir des représentations adaptées aux dialogues (via les vecteurs SKIP-ACT) obtient de bien meilleurs résultats que l'approche qui consiste à annoter automatiquement le corpus DATCHAACT. Les résultats obtenus grâce aux vecteurs SKIP-ACT laissent penser que le contexte de production des tours de parole dans le discours est une caractéristique importante à modéliser, et qu'il n'est pas possible de simplement l'induire à partir de représentations entièrement lexicales.

Un point qu'il est difficile à analyser avec les résultats de la table 7.8 est celui de l'influence des différents modèles sur la prédiction des étiquettes. Le LAS permet d'avoir une vue générale mais ne permet pas de déterminer les résultats par étiquette. Or ceux-ci pourraient être intéressants afin de déterminer les phénomènes dialogiques qui sont mieux modéliser par les différentes approches. Les figures 7.10 et 7.11 comparent les scores précision et de rappel par étiquette pour les analyseurs $\Gamma^{(25)}_{sibling}$, TP-DATCHAACT+R et TP-DATCHAREL62 avec des plongements SKIP-ACT pour ces deux derniers.

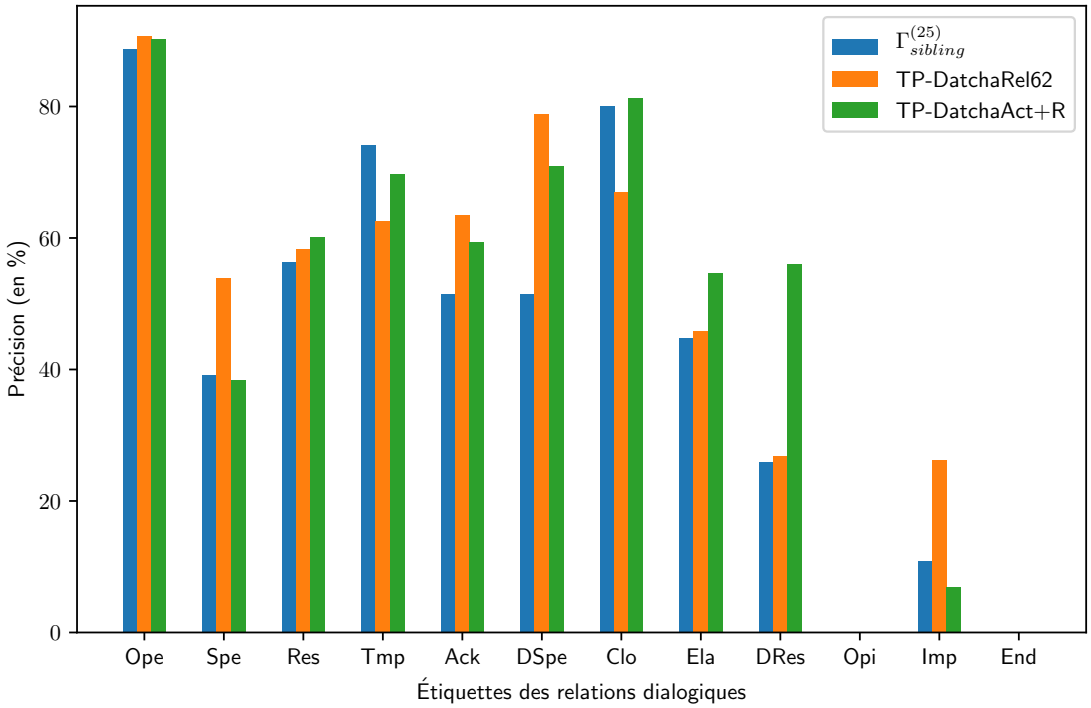


FIGURE 7.10. – Score de précision des différents analyseurs discursifs par type de relation discursive

En observant les deux graphiques, on constate — sans surprise — que certaines étiquettes sont mieux prédites que d'autres. Quel que soit l'analyseur utilisé, les relations protocolaires Ope et Clo obtiennent de bons scores. Au contraire, les relations qui dépendent beaucoup plus de la manière dont se déroule la conversation et qui sont liés à des actes de dialogue peu discriminants (STA ou INQ par exemple) obtiennent des scores faibles ou nuls (Opi, End, DRes et Imp).

Toutefois, on peut observer des comportements différents en fonction des modèles utilisées. Par rapport à $\Gamma^{(25)}_{sibling}$, TP-DATCHAREL62 permet d'obtenir une meilleure précision sur les relations Spe, Ack, DSpe et Imp ($+10$ à $+30$ points). Sur le rappel, l'amélioration porte sur les relations Spe, Res, Ack et Clo ($+10$ à $+20$ points). Par ailleurs, il n'y a que la relation Tmp qui est moins bien prédite.

Ces résultats sont intéressants car on peut y voir que les relations portant directement sur les enjeux dialogiques (c.-à-d. Spe, Imp et Res) ou sur des aspects liés à ces enjeux (DSpe et Ack) sont mieux modélisées qu'avec la grammaire. Ceci est probablement dû au fait que la grammaire n'a qu'accès aux actes de dialogue. Or, les quatre relations sont généralement liées à des actes STA ou INQ et il peut donc être difficile pour la grammaire de déterminer la bonne relation. Les plongements SKIP-ACT permettent de donner une solution à ce problème en donnant une représentation du tour qui prend en compte les mots du tour et de les considérer dans le contexte du discours conversationnel.

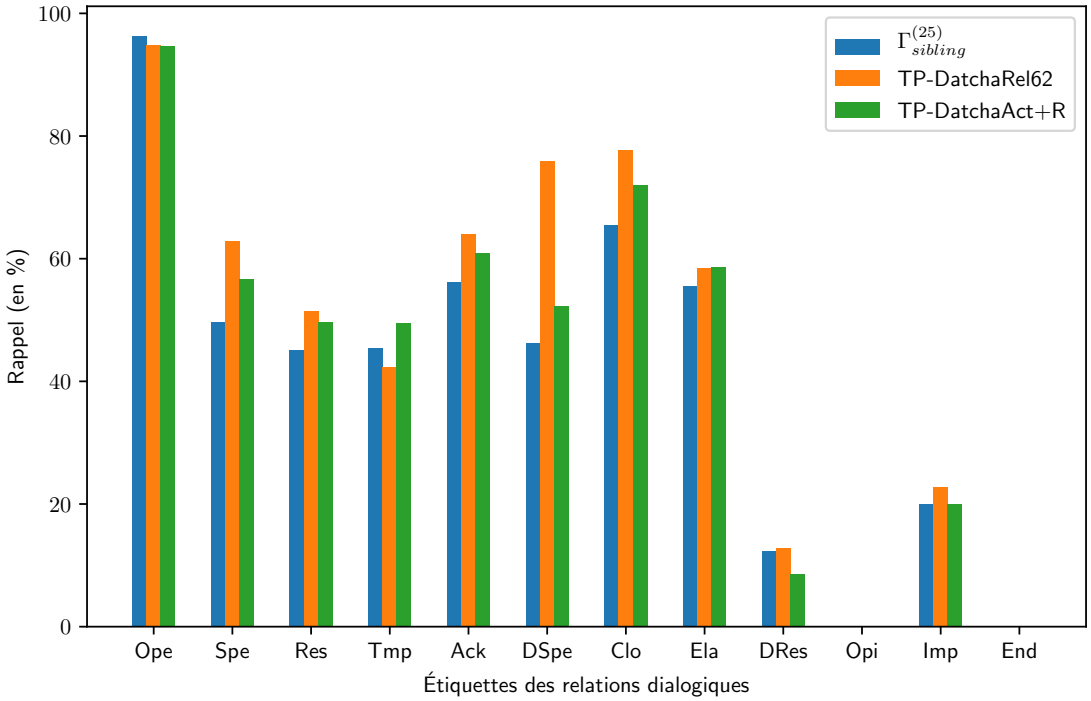


FIGURE 7.11. – Score de rappel des différents analyseurs discursifs par type de relation discursive

L'analyseur TP-DATCHAACT+R ne permet pas une aussi bonne modélisation de ces relations. En effet, même si on peut observer quelques gains sur les précisions de certaines classes (Ela, DRes et Clo principalement), ceux-ci ne se retrouvent pas dans le rappel qui est presque toujours égal ou inférieur à TP-DATCHAREL62. Par ailleurs, sur la précision, certaines classes sont également moins bien prédites, en particulier Spe (-15 points), Imp (-20) et DSpe (-8) qui permettent de définir les différents enjeux du dialogue.

Ces résultats semblent indiquer que l'enrichissement des données ne permet pas de mieux modéliser les interactions dans le dialogue. En revanche, on peut constater une amélioration de la modélisation des relations discursives ou protocolaires. Ceci est peut-être dû au fait que le bruit introduit par l'enrichissement introduit moins d'erreurs sur ces relations, celles-ci ne concernant que les tours de paroles n'ayant pas pour objectif d'interagir avec un interlocuteur.

Étant donné que je m'intéresse principalement aux interactions entre locuteurs, il est important que les modèles utilisés fassent le moins d'erreurs sur les parties du dialogue qui portent directement sur les enjeux dialogiques. Dans l'état actuel des choses et en ayant peu de données, l'enrichissement n'est pas une solution adaptée, contrairement à l'utilisation de plongements de phrases prenant explicitement en compte le discours conversationnel. En effet, même si cette dernière approche ne permet pas d'obtenir des arbres dialogiques parfaits,

celle-ci permet de déterminer les différents niveaux d'enjeux dialogiques d'un dialogue. En déterminant les différents niveaux de sous-dialogues grâce aux relations Spe et Imp et en retirant les sous-arbres induits par les relations DSpe et Ack, on obtient les tours de parole strictement nécessaires à la résolution du problème initial.

7.5. Conclusion

Dans ce chapitre, j'ai présenté plusieurs modèles permettant de réaliser une analyse profonde du discours conversationnel s'appuyant sur peu de données annotées. Deux types d'analyseurs ont été explorés avec dans un cas l'utilisation de grammaires hors-contextes et dans l'autre l'utilisation d'un analyseur en transition reposant sur un réseau de neurones. Un objectif de ce chapitre était de déterminer s'il était possible d'enrichir un corpus de petite taille à l'aide de prédictions automatiques provenant d'un analyseur ayant un apprentissage davantage contrôlé. Cette approche a ensuite été comparée à une autre cherchant non pas à ajouter des données mais à améliorer la manière dont sont représentées les données en entrée des analyseurs.

Pour réaliser l'enrichissement, l'idée a été de construire un premier analyseur à partir de peu de données en utilisant des PCFG. Cet analyseur permet de prédire des arbres discursifs de qualités raisonnables (59% de UAS et 50% de LAS sur les conversations couvertes) sur une tâche difficile. Cependant, une limite majeure de cet analyseur est qu'il ne permet pas de prédire un arbre pour toutes les conversations. Cette limite n'est pas nécessairement un inconvénient car elle permet d'identifier et de ne traiter que les conversations ayant une ressemblance avec l'ensemble des conversations de référence utilisées pour induire la CFG. Les PCFG sont ensuite utilisées afin d'enrichir automatiquement le corpus DATCHAACT avec des annotations d'arbres discursifs.

Une fois le corpus enrichi, les deux approches peuvent être comparées : plusieurs analyseurs en transition sont appris avec plusieurs configurations de corpus (enrichi ou non) et en utilisant deux types de représentations des tours de parole, dont les plongements SKIP-ACT. La comparaison des différents analyseurs permet de mettre en évidence deux résultats :

— enrichir un corpus à l'aide d'une PCFG permet d'entrainer un analyseur qui obtient des performances comparables à cette PCFG mais en ayant l'avantage de pouvoir prédire des arbres pour toutes les conversations (62% de UAS et 52% de LAS) ;

— la pertinence et la qualité des données en entrées est plus importante que la quantité de données mise à disposition pour l'analyse profonde du discours. Le modèle appris sur seulement 62 conversations et avec des plongements SKIP-ACT obtenant de bien meilleurs scores (69% de UAS et 58% de LAS). Par ailleurs, par rapport aux autres modèles étudiés, ce modèle est capable

de mieux prendre en compte les aspects purement dialogiques, ce qui peut permettre de faire ressortir les différents enjeux dialogiques d'une conversation.

Dans des travaux futurs, il serait intéressant de chercher à comprendre pourquoi la combinaison des deux approches n'est pas bénéfique. Une hypothèse est que les arbres produits par la PCFG sont trop bruités (relations erronées entre tours), ce qui a pour conséquence d'entrer en conflit avec les phénomènes modélisés par les plongements SKIP-ACT. Une solution pourrait donc être de mieux sélectionner les arbres produits automatiquement grâce aux PCFG en se fondant sur les probabilités des arbres produits ou le nombre d'arbres possibles pour une conversation donnée par exemple.

Conclusion générale

Tout au long de cette thèse, mon objectif a été de développer diverses approches permettant de représenter le discours conversationnel tout en se fondant sur des méthodologies nécessitant peu de données annotées discursivement. En effet, annoter manuellement l'ensemble des relations discursives et dialogiques dans une conversation est un processus lourd et complexe. Par ailleurs, le mode de communication sous forme de « tchats » permet d'aisément constituer un corpus de conversations très volumineux, ce qui, en règle générale, n'est pas toujours facile à avoir. Dans le but de pouvoir exploiter cette quantité de donnée, j'ai donc proposé différentes approches permettant de limiter le besoin d'annotations discursives.

J'ai apporté trois types de contribution à ce problème :

1. Étudier la possibilité de se fonder sur une tâche support sans lien explicite avec le discours conversationnel pour produire des représentations distributionnelles des conversations.
2. Se fonder sur une tâche support en lien direct avec le discours conversationnel pour produire des plongements de phrases adaptés aux dialogues.
3. Prédire des structures discursives à partir de très peu de dialogues annotés à l'aide de relations discursives et dialogiques.

Par ailleurs, dans le cadre de ces trois contributions, je me suis également intéressé à la question de l'évaluation des représentations produites — en particulier lorsqu'elles sont distributionnelles — afin de déterminer si celles-ci permettent de bien prendre en compte les différentes interactions entre locuteurs.

J'ai commencé par étudier la possibilité de se fonder sur des annotations de la satisfaction des clients à l'issue des dialogues — disponibles à très grande échelle — afin de les utiliser comme tâche support pour produire des représentations du discours conversationnel. La satisfaction client n'a pas un lien explicite avec ce dernier, mais elle peut potentiellement être utilisée pour faire ressortir des indices sur la nature des interactions dans la conversation. Afin de produire des représentations distributionnelles, l'idée a donc été d'utiliser des modèles bout en bout pour prédire les réponses à un questionnaire de satisfaction client. En effet, les questions posées aux clients portent (pour certaines) sur la qualité des interactions ayant eu lieu dans la conversation. On a constaté que la très grande subjectivité de la tâche, mais aussi le fait que la tâche est trop peu corrélée avec le problème initial, rend l'utilisation d'une telle approche très difficile à exploiter pour prendre en compte implicitement ces interactions.

Dans le but de ne plus être confronté à ces différents problèmes, je propose de me fonder sur une annotation en lien direct avec le discours conversationnel sous la forme des actes de dialogue. Cette annotation permet l'utilisation d'un étiqueteur pour produire un très grand corpus annoté automatiquement. Ces annotations (manuelles et automatiques) sont alors intéressantes pour deux utilisations : la définition d'un nouveau cadre d'évaluation de plongements de phrases qui permet de vérifier que ceux-ci prennent en compte la dimension interactive des dialogues ; la production de nouveaux plongements de phrases, les vecteurs SKIP-ACT, qui permettent de prendre en compte explicitement les interactions entre locuteur en s'appuyant sur le contexte de production des actes de dialogue.

Bien que les plongements SKIP-ACT permettent de modéliser certains aspects des interactions entre locuteurs, ces représentations sont difficiles à interpréter. En particulier, il n'est pas possible de déterminer la nature exacte des liens entre tours de parole. Dans une dernière contribution, je me suis fondé sur des annotations manuelles des structures discursives en très faible quantité afin d'apprendre des analyseurs du discours. Deux approches sont confrontées : enrichir automatiquement — à l'aide de grammaires hors-contextes — un corpus n'ayant que des annotations en actes de dialogue ; améliorer la qualité des entrées en utilisant des plongements de tours de parole prenant en compte les interactions entre locuteurs (en utilisant les vecteurs SKIP-ACT). Bien que l'enrichissement d'un corpus a montré une amélioration des prédictions par rapport à un analyseur appris directement sur très peu de conversations, l'utilisation de vecteurs SKIP-ACT s'est révélée bien plus intéressante et permet de sensiblement améliorer les résultats.

De manière générale, mes contributions montrent qu'il est difficile de prendre en compte les interactions d'un dialogue lors de la production de représentations distributionnelles. L'utilisation d'annotations en lien direct avec le discours conversationnel est nécessaire pour que les différents modèles d'apprentissage parviennent à modéliser ces phénomènes. Toutefois, on a pu voir qu'il est suffisant d'avoir à disposition des annotations incomplètes des interactions (actes de dialogue) ou en quantité très limitée (à l'aide d'un enrichissement automatique des données) afin de produire des représentations du discours conversationnel de qualité raisonnable.

Néanmoins, il reste de nombreuses questions en suspens auxquelles j'aimerais pouvoir apporter des réponses dans des travaux futurs. En effet, de manière générale, je me limite à des approches fondées sur des réseaux récurrents (avec des LSTM), avec parfois l'introduction de mécanismes d'attention. Or, ces mécanismes ont très récemment montré leur utilité dans de nombreuses tâches du TAL sous la forme de réseaux de type *transformers*. L'intérêt principal de ces derniers est qu'ils permettent de ne pas nécessairement considérer l'entrée sous forme de séquence ordonnée, et permettent ainsi de mettre en relation différentes parties d'une séquence. Par ailleurs, les *transformers* ont également montré leur utilité pour la production de plongements contextuels (par exemple BERT) qui per-

mettent des améliorations des prédictions dans de nombreuses tâches. Il parait donc indispensable d'évaluer des représentations produites à l'aide de ces plongements afin de déterminer si celles-ci prennent mieux en compte le discours conversationnel, en particulier pour les comparer aux vecteurs SKIP-ACT. En effet, les plongements contextuels permettent de produire des vecteurs différents en fonction du contexte de production d'un mot. Les plongements de tours de parole pourraient donc également profiter de cette meilleure prise en compte du contexte.

Un autre aspect qu'il serait intéressant de développer est celui de la satisfaction client et du lien qu'elle peut avoir avec le discours conversationnel. Dans le chapitre 5, on a pu constater qu'il était difficile de se reposer sur la satisfaction client pour déduire des structures discursives. Toutefois, les expériences réalisées ne permettent pas de valider le fait que la satisfaction client ne puisse pas du tout être utilisée pour identifier des interactions entre locuteurs. À partir des modèles présentés dans les chapitres 6 et 7, on peut se demander s'il serait possible d'avoir une évaluation plus précise d'un éventuel lien entre discours et satisfaction client. En effet, annoter automatiquement à l'aide de structures discursives le corpus DATCHASAT et utiliser des plongements SKIP-ACT pourraient permettre de savoir si la prise en compte des interactions entre locuteurs est utile à la prédiction de la satisfaction client.

L'évaluation des représentations est un autre point à approfondir. En effet, dans mes travaux, j'ai proposé deux types d'évaluation : l'un se fondant sur les interactions très locales à l'aide d'actes de dialogue ; l'autre se fondant sur les liens dialogiques entre tours de parole. Dans la première approche, il est difficile de juger si certaines interactions sont mieux prises en compte que d'autres. La deuxième approche résout en partie cette limite en utilisant explicitement des relations dialogiques. Toutefois, dans les deux cas, l'ensemble des interactions et des tours de parole ont le même niveau d'importance. En particulier, certaines interactions sont plus importantes que d'autres pour faire progresser le discours dans un dialogue (par exemple, une spécification est permet d'introduire les enjeux du dialogue, alors qu'une relation de fermeture est surtout protocolaire).

Afin de prendre en compte ces différences lors de l'évaluation de la qualité de représentations, une solution possible serait d'adapter les métriques utilisées lors de l'évaluation des prédictions d'arbres discursifs. En effet, les profondeurs des nœuds des arbres produits peuvent par exemple être un moyen d'identifier les sous-dialogues importants du dialogue. Par ailleurs, d'autres approches permettant d'identifier ces sous-dialogues, en utilisant des méthodes révélant les questions en discussion (*Questions Under Discussion* en anglais), pourraient être intéressantes afin d'évaluer la qualité des représentations. En effet, les questions en discussion sont une autre méthode utilisée sur les monologues afin d'identifier les différents enjeux évoqués. Cette approche pourrait donc potentiellement

être utilisée sur des dialogues afin d'identifier à quelles questions [1] permettent de répondre chaque tour de parole.

Enfin, les différentes contributions produites dans ma thèse s'appuient sur une famille de dialogues très précises : des tchats. On a pu toutefois constater que ces tchats ont de nombreux points communs avec des conversations téléphoniques. Il serait donc intéressant d'étudier comment se comportent les schémas d'annotation et les modèles étudiés sur de telles données. Une limite importante reste toutefois la disponibilité de telles conversations — un grand intérêt des tchats provient de la simplicité de la collecte des données, ce qui n'est pas le cas avec des conversations téléphoniques à cause de la transcription à réaliser. Sur les tchats, la limite provenait donc uniquement de la quantité d'annotations disponibles alors que sur des productions orales, il serait nécessaire d'utiliser des approches reposant sur peu de conversations.

Une autre particularité des conversations du corpus DATCHA est qu'elles s'inscrivent dans le cadre relativement bien défini de l'assistance clientèle. Celui-ci permet de produire des dialogues ayant un but précis, ce qui permet de bien définir les interactions possibles et la manière dont les locuteurs communiquent (le téléconseiller mène la conversation en essayant de trouver la solution au problème du client). Il serait donc intéressant d'étudier le comportement des approches proposées dans des contextes de production différents. Cela permettrait d'identifier les interactions propres aux données DATCHA et celles communes à d'autres catégories de dialogues.

1. Questions utilisées pour définir des enjeux (quel est le problème principal du client, quelles informations du client sont nécessaires pour démarrer l'assistance, etc.), à ne pas confondre avec les questions effectivement posées par les locuteurs

Bibliographie

[ACT03] Anne Abeillé, Lionel Clément et François Toussenel. « Building a Treebank for French ». In : *Treebanks*. Springer, 2003, p. 165-187 (cf. p. 97).

[Afa+15] Stergos Afantenos, Eric Kow, Nicholas Asher et Jérémy Perret. « Discourse Parsing for Multi-Party Chat Dialogues ». In : Association for Computational Linguistics (ACL), 2015 (cf. p. 32, 52, 53, 168, 169).

[AS12] Apoorv Agarwal et Jasneet Sabharwal. « End-to-End Sentiment Analysis of Twitter Data ». In : *Proceedings of the Workshop on Information Extraction and Entity Analytics on Social Media Data*. 2012, p. 39-44 (cf. p. 132).

[Agi+09] Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca et Aitor Soroa. « A Study on Similarity and Relatedness Using Distributional and Wordnet-Based Approaches ». In : (2009) (cf. p. 72).

[AMR95] Jan Alexandersson, Elisabeth Maier et Norbert Reithinger. « A Robust and Efficient Three-Layered Dialogue Component for a Speech-to-Speech Translation System ». In : *Seventh Conference of the European Chapter of the Association for Computational Linguistics*. 1995 (cf. p. 38, 45).

[All+95] James F. Allen, Lenhart K. Schubert, George Ferguson, Peter Heeman, Chung Hee Hwang, Tsuneaki Kato, Marc Light, Nathaniel Martin, Bradford Miller et Massimo Poesio. « The TRAINS Project : A Case Study in Building a Conversational Planning Agent ». In : *Journal of Experimental & Theoretical Artificial Intelligence* 7.1 (1995), p. 7-48 (cf. p. 39).

[ANA92] Jens Allwood, Joakim Nivre et Elisabeth Ahlsén. « On the Semantics and Pragmatics of Linguistic Feedback ». In : *Journal of semantics* 9.1 (1992), p. 1-26 (cf. p. 33).

[And+17] Atsushi Ando, Ryo Masumura, Hosana Kamiyama, Satoshi Kobashikawa et Yushi Aono. « Hierarchical LSTMs with Joint Learning for Estimating Customer Satisfaction from Contact Center Calls. » In : *INTERSPEECH*. 2017, p. 1716-1720 (cf. p. 117).

[Ash+16] Nicholas Asher, Julie Hunter, Mathieu Morey, Farah Benamara et Stergos Afantenos. « Discourse Structure and Dialogue Acts in Multiparty Dialogue : The STAC Corpus ». In : (2016) (cf. p. 52, 80).

[AL03] Nicholas Asher et Alex Lascarides. *Logics of Conversation*. Cambridge University Press, 2003 (cf. p. 51, 52).

[Aug+19a] Jeremy Auguste, Frédéric Béchet, Geraldine Damnati et Delphine Charlet. « Skip Act Vectors : Integrating Dialogue Context into Sentence Embeddings ». In : *Proceedings of the Tenth International Workshop on Spoken Dialogue Systems (IWSDS)*. Syracuse, Italy, 2019 (cf. p. 25).

[Aug+19b] Jeremy Auguste, Delphine Charlet, Geraldine Damnati, Frédéric Béchet et Benoit Favre. « Can We Predict Self-Reported Customer Satisfaction from Interactions ? » In : *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Brighton, United Kingdom, 2019 (cf. p. 25).

[Aug+18] Jeremy Auguste, Delphine Charlet, Géraldine Damnati, Benoit Favre et Frédéric Bechet. « Customer Satisfaction Prediction with Attention-Based RNNs from a Chat Contact Center Corpus ». In : *25e Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN)*. Rennes, France, 2018 (cf. p. 25).

[ARF17] Jeremy Auguste, Arnaud Rey et Benoit Favre. « Evaluation of Word Embeddings against Cognitive Processes : Primed Reaction Times in Lexical Decision and Naming Tasks ». In : *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*. Copenhagen, Denmark, 2017 (cf. p. 25, 73).

[Aus62] John Langshaw Austin. *How to Do Things with Words*. Oxford university press, 1962 (cf. p. 35, 36, 45).

[BRK14] Simon Baker, Roi Reichart et Anna Korhonen. « An Unsupervised Model for Instance Level Subcategorization Acquisition ». In : *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014, p. 278-289 (cf. p. 72).

[BDS08] Srinivas Bangalore, Giuseppe Di Fabbrizio et Amanda Stent. « Learning the Structure of Task-Driven Human–Human Dialogs ». In : *IEEE Transactions on Audio, Speech, and Language Processing* 16.7 (2008), p. 1249-1259 (cf. p. 55).

[Ben+03] Yoshua Bengio, Réjean Ducharme, Pascal Vincent et Christian Jauvin. « A Neural Probabilistic Language Model ». In : *Journal of machine learning research* 3.Feb (2003), p. 1137-1155 (cf. p. 63).

[BNJ03] David M. Blei, Andrew Y. Ng et Michael I. Jordan. « Latent Dirichlet Allocation ». In : *Journal of machine Learning research* 3.Jan (2003), p. 993-1022 (cf. p. 63).

[Boc+17] Joseph Bockhorst, Shi Yu, Luisa Polania et Glenn Fung. « Predicting Self-Reported Customer Satisfaction of Interactions with a Corporate Call Center ». In : *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, p. 179-190 (cf. p. 112, 117).

[Boj+17] Piotr Bojanowski, Edouard Grave, Armand Joulin et Tomas Mikolov. « Enriching Word Vectors with Subword Information ». In : *Transactions of the Association of Computational Linguistics* 5.1 (2017), p. 135-146 (cf. p. 66, 75, 110, 147, 198).

[BSL16] Mohammad Hadi Bokaei, Hossein Sameti et Yang Liu. « Extractive Summarization of Multi-Party Meetings through Discourse Segmentation ». In : *Natural Language Engineering* 22.1 (2016), p. 41-72 (cf. p. 32, 56).

[Bot+18] Chandrakant Bothe, Cornelius Weber, Sven Magg et Stefan Wermter. « A Context-Based Approach for Dialogue Act Recognition Using Simple Recurrent Neural Networks ». In : *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*. 2018 (cf. p. 44).

[Bow+15] Samuel R. Bowman, Gabor Angeli, Christopher Potts et Christopher D. Manning. « The SNLI Corpus ». In : (2015) (cf. p. 77).

[BF95] Roberto Brunelli et Daniele Falavigna. « Person Identification Using Multiple Cues ». In : *IEEE transactions on pattern analysis and machine intelligence* 17.10 (1995), p. 955-966 (cf. p. 32).

[Bru74] J. S. Bruner. « From Communication to Language—a Psychological Perspective ». In : *Cognition* 3.3 (jan. 1974), p. 255-287 (cf. p. 37).

[Bru+12] Elia Bruni, Gemma Boleda, Marco Baroni et Nam-Khanh Tran. « Distributional Semantics in Technicolor ». In : *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics : Long Papers-Volume 1*. Association for Computational Linguistics, 2012, p. 136-145 (cf. p. 72).

[Bun06] Harry Bunt. « Dimensions in Dialogue Act Annotation. » In : *LREC*. 2006, p. 919-924 (cf. p. 42).

[Bun09] Harry Bunt. « The DIT++ Taxonomy for Functional Dialogue Markup ». In : *AAMAS 2009 Workshop, Towards a Standard Markup Language for Embodied Dialogue Acts*. 2009, p. 13-24 (cf. p. 43).

[Car+05] Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij et Melissa Kronenthal. « The AMI Meeting Corpus : A Pre-Announcement ». In : *International Workshop on Machine Learning for Multimodal Interaction*. Springer, 2005, p. 28-39 (cf. p. 56).

[CM14] Danqi Chen et Christopher D. Manning. « A Fast and Accurate Dependency Parser Using Neural Networks ». In : *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014, p. 740-750 (cf. p. 199).

[Che+18] Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai et Xiaofei He. « Dialogue Act Recognition via Crf-Attentive Structured Network ». In : *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 2018, p. 225-234 (cf. p. 44).

[Cho+14] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau et Yoshua Bengio. « On the Properties of Neural Machine Translation : Encoder-Decoder Approaches ». In : *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)* (2014) (cf. p. 103).

[Cho57] Noam Chomsky. *Syntactic Structures*. English. The Hague : Mouton, 1957 (cf. p. 177).

[CSR16] Shammur Absar Chowdhury, Evgeny A. Stepanov et Giuseppe Riccardi. « Predicting User Satisfaction from Turn-Taking in Spoken Conversations. » In : *Interspeech*. 2016, p. 2910-2914 (cf. p. 117).

[Coc70] John Cocke. « Programming Languages and Their Compilers : Preliminary Notes ». In : (1970) (cf. p. 188).

[CW08] Ronan Collobert et Jason Weston. « A Unified Architecture for Natural Language Processing : Deep Neural Networks with Multitask Learning ». In : *Proceedings of the 25th International Conference on Machine Learning*. ACM, 2008, p. 160-167 (cf. p. 64).

[CK18] Alexis Conneau et Douwe Kiela. « SentEval : An Evaluation Toolkit for Universal Sentence Representations ». In : *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan, 2018 (cf. p. 77).

[Con+17] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault et Antoine Bordes. « Supervised Learning of Universal Sentence Representations from Natural Language Inference Data ». In : *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017, p. 670-680 (cf. p. 69).

[Con+18] Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault et Marco Baroni. « What You Can Cram into a Single \backslash$& !#* Vector : Probing Sentence Embeddings for Linguistic Properties ». In : *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*. Melbourne, Australia : Association for Computational Linguistics, 2018, p. 2126-2136 (cf. p. 76).

[Con+11] Mathieu Constant, Isabelle Tellier, Denys Duchier, Yoann Dupont, Anthony Sigogne et Sylvie Billot. « Intégrer Des Connaissances Linguistiques Dans Un CRF : Application à l'apprentissage d'un Segmenteur-Étiqueteur Du Français ». In : *TALN*. T. 1. 2011, p. 321 (cf. p. 97).

[CA97] Mark G. Core et James Allen. « Coding Dialogs with the DAMSL Annotation Scheme ». In : *AAAI Fall Symposium on Communicative Action in Humans and Machines*. T. 56. Boston, MA, 1997 (cf. p. 40).

[Dam64] Fred J. Damerau. « A Technique for Computer Detection and Correction of Spelling Errors ». In : *Communications of the ACM* 7.3 (mar. 1964), p. 171-176 (cf. p. 101).

[Dam+18] Géraldine Damnati, Jeremy Auguste, Alexis Nasr, Delphine Charlet, Johannes Heinecke et Frédéric Bechet. « Handling Normalization Issues for Part-of-Speech Tagging of Online Conversational Text ». In : *Eleventh International Conference on Language Resources and Evaluation (LREC)*. Miyazaki, Japan, 2018 (cf. p. 25).

[DGC16] Géraldine Damnati, Aleksandra Guerraz et Delphine Charlet. « Web Chat Conversations from Contact Centers : A Descriptive Study ». In : *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. 2016, p. 2017-2021 (cf. p. 85, 86, 88).

[de +17] Maira Gatti de Bayser, Paulo Cavalin, Renan Souza, Alan Braz, Heloisa Candello, Claudio Pinhanez et Jean-Pierre Briot. « A Hybrid Architecture for Multi-Party Conversational Systems ». In : *arXiv :1705.01214 [cs]* (mai 2017). arXiv : 1705.01214 [cs] (cf. p. 32).

[de 16] Ferdinand de Saussure. *Cours de Linguistique Générale*. 1916 (cf. p. 61).

[Dee+90] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer et Richard Harshman. « Indexing by Latent Semantic Analysis ». In : *Journal of the American society for information science* 41.6 (1990), p. 391-407 (cf. p. 62).

[DS09] Pascal Denis et Benoît Sagot. « Coupling an Annotated Corpus and a Morphosyntactic Lexicon for State-of-the-Art POS Tagging with Less Human Effort ». In : *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*. 2009 (cf. p. 97).

[Dev+19] Jacob Devlin, Ming-Wei Chang, Kenton Lee et Kristina Toutanova. « BERT : Pre-Training of Deep Bidirectional Transformers for Language Understanding ». In : *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, p. 4171-4186 (cf. p. 68).

[EC10] Micha Elsner et Eugene Charniak. « Disentangling Chat ». In : *Computational Linguistics* 36.3 (2010), p. 389-409 (cf. p. 47).

[FD14] Manaal Faruqui et Chris Dyer. « Improving Vector Space Word Representations Using Multilingual Correlation ». In : Association for Computational Linguistics, 2014 (cf. p. 75).

[Far+16] Manaal Faruqui, Yulia Tsvetkov, Pushpendre Rastogi et Chris Dyer. « Problems with Evaluation of Word Embeddings Using Word Similarity Tasks ». In : *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*. 2016, p. 30-35 (cf. p. 71, 72).

[Fil71] Charles J. Fillmore. « Some Problems for Case Grammar ». In : *Monograph Series on Languages and Linguistics* (1971) (cf. p. 37).

[Fin+01] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman et Eytan Ruppin. « Placing Search in Context : The Concept Revisited ». In : *Proceedings of the 10th International Conference on World Wide Web*. 2001, p. 406-414 (cf. p. 72).

[Fir57] John R. Firth. « A Synopsis of Linguistic Theory, 1930-1955 ». In : *Studies in linguistic analysis* (1957) (cf. p. 62).

[For+03] Katherine Forbes, Eleni Miltsakaki, Rashmi Prasad, Anoop Sarkar, Aravind Joshi et Bonnie Webber. « D-LTAG System : Discourse Parsing with a Lexicalized Tree-Adjoining Grammar ». In : *Journal of Logic, Language and Information* 12.3 (2003), p. 261-279 (cf. p. 52).

[Gai65] Haim Gaifman. « Dependency Systems and Phrase-Structure Systems ». In : *Information and control* 8.3 (1965), p. 304-337 (cf. p. 181).

[Ger+16] Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart et Anna Korhonen. « SimVerb-3500 : A Large-Scale Evaluation Set of Verb Similarity ». In : *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016, p. 2173-2182 (cf. p. 72).

[GHM92] John J. Godfrey, Edward C. Holliman et Jane McDaniel. « SWITCHBOARD : Telephone Speech Corpus for Research and Development ». In : *[Proceedings] ICASSP-92 : 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*. T. 1. IEEE, 1992, p. 517-520 (cf. p. 42, 44, 80, 88).

[Hak09] Dilek Hakkani-Tur. « Towards Automatic Argument Diagramming of Multiparity Meetings ». In : *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, p. 4753-4756 (cf. p. 56).

[Hal+12] Guy Halawi, Gideon Dror, Evgeniy Gabrilovich et Yehuda Koren. « Large-Scale Learning of Word Relatedness with Constraints ». In : *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2012, p. 1406-1414 (cf. p. 72).

[Han79] Michael Hancher. « The Classification of Cooperative Illocutionary Acts ». en. In : *Language in Society* 8.1 (avr. 1979), p. 1-14 (cf. p. 36).

[Har54] Zellig S. Harris. « Distributional Structure ». In : *Word* 10.2-3 (1954), p. 146-162 (cf. p. 62).

[Hay64] David G. Hays. « Dependency Theory : A Formalism and Some Observations ». In : *Language* 40.4 (1964), p. 511-525 (cf. p. 181).

[HLA19] Shizhu He, Kang Liu et Weiting An. « Learning to Align Question and Answer Utterances in Customer Service Conversation with Recurrent Pointer Networks ». In : (2019) (cf. p. 47).

[HCK16] Felix Hill, Kyunghyun Cho et Anna Korhonen. « Learning Distributed Representations of Sentences from Unlabelled Data ». In : *Proceedings of NAACL-HLT*. 2016, p. 1367-1377 (cf. p. 69, 74, 76).

[HRK16] Felix Hill, Roi Reichart et Anna Korhonen. « Simlex-999 : Evaluating Semantic Models with (Genuine) Similarity Estimation ». In : *Computational Linguistics* (2016) (cf. p. 72).

[HFF15] Jennifer Hill, W. Randolph Ford et Ingrid G. Farreras. « Real Conversations with Artificial Intelligence : A Comparison between Human–Human Online Conversations and Human–Chatbot Conversations ». In : *Computers in Human Behavior* 49 (2015), p. 245-250 (cf. p. 31).

[HS97] Sepp Hochreiter et Jürgen Schmidhuber. « Long Short-Term Memory ». In : *Neural computation* 9.8 (1997), p. 1735-1780 (cf. p. 65).

[HW16] Geoff Hollis et Chris Westbury. « The Principals of Meaning : Extracting Semantic Dimensions from Co-Occurrence Models of Semantics ». In : *Psychonomic bulletin & review* 23.6 (2016), p. 1744-1756 (cf. p. 73).

[Hua+12] Eric H. Huang, Richard Socher, Christopher D. Manning et Andrew Y. Ng. « Improving Word Representations via Global Context and Multiple Word Prototypes ». In : *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics : Long Papers-Volume 1*. Association for Computational Linguistics, 2012, p. 873-882 (cf. p. 67).

[Hut+13] Keith A. Hutchison, David A. Balota, James H. Neely, Michael J. Cortese, Emily R. Cohen-Shikora, Chi-Shing Tse, Melvin J. Yap, Jesse J. Bengson, Dale Niemeyer et Erin Buchanan. « The Semantic Priming Project ». In : *Behavior Research Methods* 45.4 (2013), p. 1099-1114 (cf. p. 74).

[Jek+95] Susanne Jekat, Alexandra Klein, Elisabeth Maier, Ilona Maleck, Marion Mast et J. Joachim Quantz. « Dialogue Acts in VERBMOBIL ». In : (1995) (cf. p. 38).

[Jia+18] Jyun-Yu Jiang, Francine Chen, Yan-Ying Chen et Wei Wang. « Learning to Disentangle Interleaved Conversational Threads with a Siamese Hierarchical Network and Similarity Ranking ». In : *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*. 2018, p. 1812-1822 (cf. p. 47).

[Jo04] Natasa Jovanovic et Rieks op den Akker. « Towards Automatic Addressee Identification in Multi-Party Dialogues ». In : *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*. 2004, p. 89-92 (cf. p. 32).

[JSB97] D. Jurafsky, E. Shriberg et D. Biasca. « Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual (1997) ». In : (1997) (cf. p. 42, 44).

[Kas66] Tadao Kasami. « An Efficient Recognition and Syntax-Analysis Algorithm for Context-Free Languages ». In : *Coordinated Science Laboratory Report no. R-257* (1966) (cf. p. 188).

[Kay92] Martin Kay. *Verbmobil : A Translation System for Face-to-Face Dialog*. Sous la dir. de Peter Norvig et Mark Gawron. Chicago, IL, USA : University of Chicago Press, 1992 (cf. p. 38).

[Kha+18] Chandra Khatri, Rahul Goel, Behnam Hedayatnia, Angeliki Metanillou, Anushree Venkatesh, Raefer Gabriel et Arindam Mandal. « Contextual Topic Modeling For Dialog Systems ». In : *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, p. 892-899 (cf. p. 45).

[Kim14] Yoon Kim. « Convolutional Neural Networks for Sentence Classification ». In : *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (août 2014), p. 1746-1751 (cf. p. 119).

[Kim+16] Yoon Kim, Yacine Jernite, David Sontag et Alexander M. Rush. « Character-Aware Neural Language Models ». In : *Thirtieth AAAI Conference on Artificial Intelligence*. 2016 (cf. p. 65).

[KB15] Diederik P. Kingma et Jimmy Ba. « Adam : A Method for Stochastic Optimization ». In : *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA, 2015 (cf. p. 103).

[Kir+15] Ryan Kiros, Yukun Zhu, Ruslan R. Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba et Sanja Fidler. « Skip-Thought Vectors ». In : *Advances in Neural Information Processing Systems*. 2015, p. 3294-3302 (cf. p. 69, 76, 147).

[Kum+18] Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta et Sachindra Joshi. « Dialogue Act Sequence Labeling Using Hierarchical Encoder with Crf ». In : *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018 (cf. p. 44).

[Kum+08] Vivek Kumar, Rangarajan Sridhar, Shrikanth Narayanan et Srinivas Bangalore. « Enriching Spoken Language Translation with Dialog Acts ». In : *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies : Short Papers*. Association for Computational Linguistics, 2008, p. 225-228 (cf. p. 45).

[KSY89] Masako Kume, Gayle K. Sato et Kei Yoshimoto. « A Descriptive Framework for Translating Speaker's Meaning : Towards a Dialogue Translation System Between Japanese and English ». In : *Proceedings of the Fourth Conference on European Chapter of the Association for Computational Linguistics*. EACL '89. Stroudsburg, PA, USA : Association for Computational Linguistics, 1989, p. 264-271 (cf. p. 38).

[LM14] Quoc Le et Tomas Mikolov. « Distributed Representations of Sentences and Documents ». In : *International Conference on Machine Learning*. 2014, p. 1188-1196 (cf. p. 70).

[LG14a] Omer Levy et Yoav Goldberg. « Dependency-Based Word Embeddings. » In : *ACL (2)*. Citeseer, 2014, p. 302-308 (cf. p. 75).

[LG14b] Omer Levy et Yoav Goldberg. « Linguistic Regularities in Sparse and Explicit Word Representations ». In : *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*. 2014, p. 171-180 (cf. p. 73).

[Lin+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár et C. Lawrence Zitnick. « Microsoft Coco : Common Objects in Context ». In : *European Conference on Computer Vision*. Springer, 2014, p. 740-755 (cf. p. 77).

[Lin+15] Wang Ling, Chris Dyer, Alan W. Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luis Marujo et Tiago Luis. « Finding Function in Form : Compositional Character Models for Open Vocabulary Word Representation ». In : *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015, p. 1520-1530 (cf. p. 65).

[LRH05] Jackson Liscombe, Giuseppe Riccardi et Dilek Hakkani-Tur. « Using Context to Improve Emotion Detection in Spoken Dialog Systems ». In : *Proceedings of Eurospeech'05*. 2005 (cf. p. 45).

[Loc98] Karen E. Lochbaum. « A Collaborative Planning Model of Intentional Structure ». In : *Computational linguistics* 24.4 (1998), p. 525-572 (cf. p. 55).

[LB02] Edward Loper et Steven Bird. « NLTK : The Natural Language Toolkit ». In : *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*. ETMTNLP '02. Stroudsburg, PA, USA : Association for Computational Linguistics, 2002, p. 63-70 (cf. p. 190).

[Low+15] Ryan Lowe, Nissan Pow, Iulian Vlad Serban et Joelle Pineau. « The Ubuntu Dialogue Corpus : A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems ». In : *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 2015, p. 285-294 (cf. p. 80).

[LB96] Kevin Lund et Curt Burgess. « Producing High-Dimensional Semantic Spaces from Lexical Co-Occurrence ». In : *Behavior research methods, instruments, & computers* 28.2 (1996), p. 203-208 (cf. p. 63).

[LSM13] Thang Luong, Richard Socher et Christopher D. Manning. « Better Word Representations with Recursive Neural Networks for Morphology. » In : *CoNLL*. 2013, p. 104-113 (cf. p. 72).

[Luq+17] Jordi Luque, Carlos Segura, Ariadna Sánchez, Martí Umbert et Luis Angel Galindo. « The Role of Linguistic and Prosodic Cues on the Prediction of Self-Reported Satisfaction in Contact Centre Phone Calls. » In : *INTERSPEECH*. 2017, p. 2346-2350 (cf. p. 112, 117).

[MT88] William C. Mann et Sandra A. Thompson. « Rhetorical Structure Theory : Toward a Functional Theory of Text Organization ». In : *Text-Interdisciplinary Journal for the Study of Discourse* 8.3 (1988), p. 243-281 (cf. p. 50).

[MSM93] Mitchell Marcus, Beatrice Santorini et Mary Ann Marcinkiewicz. « Building a Large Annotated Corpus of English : The Penn Treebank ». In : (1993) (cf. p. 51).

[MAR12] Elijah Mayfield, David Adamson et Carolyn Penstein Rosé. « Hierarchical Conversation Structure Prediction in Multi-Party Chat ». In : *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, 2012, p. 60-69 (cf. p. 47).

[McC+17] Bryan McCann, James Bradbury, Caiming Xiong et Richard Socher. « Learned in Translation : Contextualized Word Vectors ». In : *Advances in Neural Information Processing Systems*. 2017, p. 6294-6305 (cf. p. 68).

[MP06] Ryan McDonald et Fernando Pereira. « Online Learning of Approximate Dependency Parsing Algorithms ». In : *11th Conference of the European Chapter of the Association for Computational Linguistics*. 2006 (cf. p. 177).

[MSR07] Kathleen McKeown, Lokesh Shrestha et Owen Rambow. « Using Question-Answer Pairs in Extractive Summarization of Email Conversations ». In : *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 2007, p. 542-550 (cf. p. 47).

[MHM09] T. Daniel Midgley, Shelly Harrison et Cara MacNish. « Empirical Verification of Adjacency Pairs Using Dialogue Segmentation ». In : *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*. Association for Computational Linguistics, 2009, p. 104-108 (cf. p. 47).

[Mik+13] Tomas Mikolov, Kai Chen, Greg Corrado et Jeffrey Dean. « Efficient Estimation of Word Representations in Vector Space ». In : *In Proceedings of Workshop at ICLR* (2013) (cf. p. 63, 75, 102, 147).

[MYZ13] Tomas Mikolov, Wen-tau Yih et Geoffrey Zweig. « Linguistic Regularities in Continuous Space Word Representations ». In : *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*. 2013, p. 746-751 (cf. p. 72).

[Mil95] George A. Miller. « WordNet : A Lexical Database for English ». In : *Communications of the ACM* 38.11 (1995), p. 39-41 (cf. p. 62).

[MC91] George A. Miller et Walter G. Charles. « Contextual Correlates of Semantic Similarity ». In : *Language and cognitive processes* 6.1 (1991), p. 1-28 (cf. p. 72).

[MX17] Yi Mou et Kun Xu. « The Media Inequality : Comparing the Initial Human-Human and Human-AI Social Interactions ». In : *Computers in Human Behavior* 72 (2017), p. 432-440 (cf. p. 31).

[Mul+12] Philippe Muller, Stergos Afantenos, Pascal Denis et Nicholas Asher. « Constrained Decoding for Text-Level Discourse Parsing ». In : *Proceedings of COLING 2012*. 2012, p. 1883-1900 (cf. p. 53).

[NM94] Masaaki Nagata et Tsuyoshi Morimoto. « First Steps towards Statistical Modeling of Dialogue to Predict the Speech Act Type of the next Utterance ». en. In : *Speech Communication* 15.3-4 (déc. 1994), p. 193-203 (cf. p. 38).

[Nas+16] Alexis Nasr, Geraldine Damnati, Aleksandra Guerraz et Frederic Bechet. « Syntactic Parsing of Chat Language in Contact Center Conversation Corpus ». In : *17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. 2016, p. 175 (cf. p. 84, 98).

[NN05] Joakim Nivre et Jens Nilsson. « Pseudo-Projective Dependency Parsing ». In : *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2005, p. 99-106 (cf. p. 181).

[Ohm71] Richard Ohmann. « Speech Acts and the Definition of Literature ». en. In : *Philosophy & Rhetoric* 4.1 (1971), p. 1-19 (cf. p. 37).

[Ped+11] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss et Vincent Dubourg. « Scikit-Learn : Machine Learning in Python ». In : *Journal of machine learning research* 12.Oct (2011), p. 2825-2830 (cf. p. 119).

[PSM14] Jeffrey Pennington, Richard Socher et Christopher D Manning. « Glove : Global Vectors for Word Representation. » In : *EMNLP*. T. 14. 2014, p. 1532-1543 (cf. p. 63, 75).

[Per+16] Francisco Pereira, Samuel Gershman, Samuel Ritter et Matthew Botvinick. « A Comparative Evaluation of Off-the-Shelf Distributed Semantic Representations for Modelling Behavioural Data ». In : *Cognitive neuropsychology* 33.3-4 (2016), p. 175-190 (cf. p. 73).

[PW17] Luis Perez et Jason Wang. « The Effectiveness of Data Augmentation in Image Classification Using Deep Learning ». In : *arXiv preprint arXiv :1712.04621* (2017) (cf. p. 167).

[PNA18] Robin Perrotin, Alexis Nasr et Jeremy Auguste. « Dialog Acts Annotations for Online Chats ». In : *25e Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN)*. Rennes, France, 2018 (cf. p. 25, 94).

[Pet+18] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee et Luke Zettlemoyer. « Deep Contextualized Word Representations ». In : *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*. T. 1. 2018, p. 2227-2237 (cf. p. 68).

[Pol+04] Livia Polanyi, Chris Culy, Martin Van Den Berg, Gian Lorenzo Thione et David Ahn. « A Rule Based Approach to Discourse Parsing ». In : *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*. 2004 (cf. p. 52).

[Pra+18] Louisa Pragst, Niklas Rach, Wolfgang Minker et Stefan Ultes. « On the Vector Representation of Utterances in Dialogue Context. » In : *LREC*. 2018 (cf. p. 145).

[Pra+08] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K. Joshi et Bonnie L. Webber. « The Penn Discourse TreeBank 2.0. » In : *LREC*. Citeseer, 2008 (cf. p. 51, 55).

[QWK17] Kechen Qin, Lu Wang et Joseph Kim. « Joint Modeling of Content and Discourse Relations in Dialogues ». In : *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*. 2017, p. 974-984 (cf. p. 56).

[Rad+11] Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich et Shaul Markovitch. « A Word at a Time : Computing Word Relatedness Using Temporal Semantic Analysis ». In : *Proceedings of the 20th International Conference on World Wide Web*. 2011, p. 337-346 (cf. p. 72).

[Rei03] Frederick F. Reichheld. « The One Number You Need to Grow ». In : *Harvard business review* 81.12 (2003), p. 46-55 (cf. p. 115).

[RM10] Joseph Reisinger et Raymond J. Mooney. « Multi-Prototype Vector-Space Models of Word Meaning ». In : *Human Language Technologies : The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, p. 109-117 (cf. p. 67).

[Rei+96] Norbert Reithinger, Ralf Engel, Michael Kipp et Martin Klesen. « Predicting Dialogue Acts for a Speech-to-Speech Translation System ». In : *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*. T. 2. IEEE, 1996, p. 654-657 (cf. p. 45).

[RHv05] Rutger Rienks, Dirk Heylen et Erik van der Weijden. « Argument Diagramming of Meeting Conversations ». In : *Multimodal Multiparty Meeting Processing, Workshop at the 7th International Conference on Multimodal Interfaces*. 2005, p. 85-92 (cf. p. 56).

[Roy+16] Shourya Roy, Ragunathan Mariappan, Sandipan Dandapat, Saurabh Srivastava, Sainyam Galhotra et Balaji Peddamuthu. « Qa Rt : A System for Real-Time Holistic Quality Assurance for Contact Center Dialogues ». In : *Thirtieth AAAI Conference on Artificial Intelligence*. 2016 (cf. p. 117).

[RG65] Herbert Rubenstein et John B. Goodenough. « Contextual Correlates of Synonymy ». In : *Communications of the ACM* 8.10 (1965), p. 627-633 (cf. p. 72).

[SSJ78] Harvey Sacks, Emanuel A. Schegloff et Gail Jefferson. « A Simplest Systematics for the Organization of Turn Taking for Conversation ». In : *Studies in the Organization of Conversational Interaction*. Elsevier, 1978, p. 7-55 (cf. p. 32).

[Sad74] Jerrold M. Sadock. *Toward a Linguistic Theory of Speech Acts*. en. New York : Academic Press, 1974 (cf. p. 37).

[Sag10] Benoît Sagot. « The Lefff, a Freely Available and Large-Coverage Morphological and Syntactic Lexicon for French ». In : *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*. 2010 (cf. p. 103).

[SS73] Emanuel A. Schegloff et Harvey Sacks. « Opening up Closings ». In : *Semiotica* 8.4 (1973), p. 289-327 (cf. p. 46).

[Sea75] John R. Searle. « A Taxonomy of Illocutionary Acts ». In : (1975) (cf. p. 36, 38, 39, 43, 45).

[SHB16] Rico Sennrich, Barry Haddow et Alexandra Birch. « Neural Machine Translation of Rare Words with Subword Units ». In : *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*. 2016, p. 1715-1725 (cf. p. 66, 110).

[SA07] Bayan Abu Shawar et Eric Atwell. « Chatbots : Are They Really Useful ? » In : *Ldv Forum*. T. 22. 2007, p. 29-49 (cf. p. 31).

[SK19] Connor Shorten et Taghi M. Khoshgoftaar. « A Survey on Image Data Augmentation for Deep Learning ». In : *Journal of Big Data* 6.1 (2019), p. 60 (cf. p. 167).

[Shr+04] Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang et Hannah Carvey. « The ICSI Meeting Recorder Dialog Act (MRDA) Corpus ». In : *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*. 2004, p. 97-100 (cf. p. 44).

[Søg16] Anders Søgaard. « Evaluating Word Embeddings with fMRI and Eye-Tracking ». In : *ACL 2016* (2016), p. 116 (cf. p. 73).

[Tay+98] Paul Taylor, Simon King, Stephen Isard et Helen Wright. « Intonation and Dialog Context as Constraints for Speech Recognition ». In : *Language and Speech* 41.3-4 (1998), p. 493-512 (cf. p. 33, 45).

[Tho+19] Catherine Thompson, Nicholas Asher, Philippe Muller et Jeremy Auguste. « Weakly Supervised Dialog Act Analysis ». In : *Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN - PFIA 2019)*. Toulouse, France, 2019 (cf. p. 25).

[TML15] Andrew Trask, Phil Michalak et John Liu. « Sense2vec-a Fast and Accurate Method for Word Sense Disambiguation in Neural Word Embeddings ». In : *arXiv preprint arXiv :1511.06388* (2015) (cf. p. 67).

[Tra04] David Traum. « Issues in Multiparty Dialogues ». In : *Advances in Agent Communication*. Sous la dir. de Gerhard Goos, Juris Hartmanis, Jan van Leeuwen et Frank Dignum. T. 2922. Berlin, Heidelberg : Springer Berlin Heidelberg, 2004, p. 201-211 (cf. p. 31).

[TH92] David R. Traum et Elizabeth A. Hinkelman. « Conversation Acts in Task-Oriented Spoken Dialogue ». In : *Computational intelligence* 8.3 (1992), p. 575-599 (cf. p. 48).

[TR02] David Traum et Jeff Rickel. « Embodied Agents for Multi-Party Dialogue in Immersive Virtual Worlds ». In : *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems : Part 2*. ACM, 2002, p. 766-773 (cf. p. 32).

[Tur+08] Gokhan Tur, Andreas Stolcke, Lynn Voss, John Dowding, Benoit Favre, Raquel Fernández, Matthew Frampton, Michael Frandsen, Clint Frederickson et Martin Graciarena. « The CALO Meeting Speech Recognition and Understanding System ». In : *2008 IEEE Spoken Language Technology Workshop*. IEEE, 2008, p. 69-72 (cf. p. 32).

[VRH06] Daan Verbree, Rutger Rienks et Dirk Heylen. « First Steps towards the Automatic Construction of Argument-Diagrams from Real Discussions ». In : *Frontiers in Artificial Intelligence and Applications* 144 (2006), p. 183 (cf. p. 56).

[VVR16] Soroush Vosoughi, Prashanth Vijayaraghavan et Deb Roy. « Tweet2vec : Learning Tweet Embeddings Using Character-Level Cnn-Lstm Encoder-Decoder ». In : *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2016, p. 1041-1044 (cf. p. 70).

[Wei66] Joseph Weizenbaum. « ELIZA—a Computer Program for the Study of Natural Language Communication between Man and Machine ». In : *Communications of the ACM* 9.1 (1966), p. 36-45 (cf. p. 17).

[Wie+16] John Wieting, Mohit Bansal, Kevin Gimpel et Karen Livescu. « Charagram : Embedding Words and Sentences via Character n-Grams ». In : *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016, p. 1504-1515 (cf. p. 66).

[WNB18] Adina Williams, Nikita Nangia et Samuel Bowman. « A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference ». In : *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*. 2018, p. 1112-1122 (cf. p. 77).

[WG05] Florian Wolf et Edward Gibson. « Representing Discourse Coherence : A Corpus-Based Study ». In : *Computational linguistics* 31.2 (2005), p. 249-287 (cf. p. 52).

[Wri98] Helen Wright. « Automatic Utterance Type Detection Using Suprasegmental Features ». In : *Proceedings of International Conference on Spoken Language Processing*. 1998 (cf. p. 47).

[Wu+17] Yu Wu, Wei Wu, Chen Xing, Ming Zhou et Zhoujun Li. « Sequential Matching Network : A New Architecture for Multi-Turn Response Selection in Retrieval-Based Chatbots ». In : *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*. 2017, p. 496-505 (cf. p. 19).

[XSJ16] Nianwen Xue, Qishen Su et Sooyoung Jeong. « Annotating the Discourse and Dialogue Structure of SMS Message Conversations ». In : *Proceedings of the 10th Linguistic Annotation Workshop Held in Conjunction with ACL 2016 (LAW-X 2016)*. 2016, p. 180-187 (cf. p. 54, 168).

[YM03] Hiroyasu Yamada et Yuji Matsumoto. « Statistical Dependency Analysis with Support Vector Machines ». In : *Proceedings of the Eighth International Conference on Parsing Technologies*. 2003, p. 195-206 (cf. p. 177).

[YP06] Dongqiang Yang et David Martin Powers. *Verb Similarity on the Taxonomy of WordNet*. Masaryk University, 2006 (cf. p. 72).

[Yan+19] Zhilin YANG, Zihang DAI, Yiming YANG, Jaime CARBONELL, Ruslan SALAKHUTDINOV et Quoc V. LE. « XLNet : Generalized Autoregressive Pretraining for Language Understanding ». In : *arXiv preprint arXiv :1906.08237* (2019) (cf. p. 68).

[You67] Daniel H. YOUNGER. « Recognition and Parsing of Context-Free Languages in Time N3 ». In : *Information and control* 10.2 (1967), p. 189-208 (cf. p. 188).

Index

Acknowledgement (acte de dialogue) 90
Acknowledgement (relation discursive) 172
Acte de dialogue 37, 45
Acte de la conversation 48
Acte illocutoire 35
Acte locutoire 35
Acte perlocutoire 35
Algorithme de Cocke-Younger-Kasami 188
Algorithme Inside-Outside 187
Allocuteur 29
Analyse du discours 28
Analyse sémantique latente 62
Analyse syntagmatique 177
Arbre syntagmatique 177
Association sémantique 72

Bruit (corpus DATCHA) 82

Clarification question (acte de dialogue) 91
Closing (acte de dialogue) 89
Closing (relation discursive) 174
Constativité 35
Conversation 29
Couverture 191

Détracteur (NPS) 116
Destinataire 29
Dialogue 29
Dimension (acte de dialogue) 40

Elaboration (relation discursive) 174
Enchevêtrement de sous-dialogues 30
Encodage à un bit non nul discriminant 60
Encodage par pairs d'octets 66
Ending (relation discursive) 172
Énoncé 29
Erreur absolue moyenne 122

F-mesure 126
Fonction de communication prospective 40
Fonction de communication rétrospective 40
Force illocutoire 38
Forme normale de Chomsky 179

GloVe 63
Gouverneur (analyse du discours) 171
Grammaire formelle 178
Grammaire hors-contexte 179
Grammaire hors-contexte probabiliste 187

Imperative (relation discursive) 171
Information question (acte de dialogue) 91
Interlocuteur 29

Locuteur 29

MacroF1 126
Message 29
Modèle thématique 62
Mot inconnu 64

Net Promoter Score 115

Opening (acte de dialogue) 89
Opening (relation discursive) 173
Opinion (relation discursive) 172

Paires adjacentes 46
Partie du discours 67
Passif (NPS) 116
Performativité 35
Plan proposal (acte de dialogue) 91
plongement de mots contextuel 68
Plongements de mots 62
Problem description (acte de dialogue) 91
Projectif (arbre de dépendance) 180
Promoteur (NPS) 116
Propriété (plongements) 70

Réseau de neurones convolutifs 65
Réseau Long-Short Term Memory 65
Raisonnement par analogie 72
Relation dialogique 170
Relation discursive 169

Relation entre tours de parole 30
Relation prospective 53
Relation rétrospective 53
Représentation distributionnelle 62
Response (relation discursive) 171

Score de rattachement étiqueté 191
Score de rattachement non étiqueté 191
Scripteur 29
Silhouette (dialogue) 86
Similarité cosinus 71
Similarité sémantique 72
Sous-dialogue 29
Specification (relation discursive) 171
Statement (acte de dialogue) 90
Structure de surface du discours 34

Tâche de décision lexicale (processus cognitif) 74
Tâche de dénomination (processus cognitif) 74
Taux d'erreurs mots 98
Taux d'erreurs sérieuses 125
Taux de messages avec substitution 98
Temporisation (acte de dialogue) 90
Temporisation (relation discursive) 172
Théorie des actes de langage 35
Tour de parole 29

Unité du discours 50

Vecteur Skip-Act 154
Vecteur Skip-Thought 69

Word2Vec 63

ANNEXES

Sommaire

A Exemple de dialogue avec annotations 233
B Lexique LEX5 234
C Lexique LEX6 235
D Caractéristiques utilisées par l'analyseur en transition 236

A. Exemple de dialogue avec annotations

	G.[1]	Rel.	Acte	Loc.	Tour de parole
1.	0	Ope	OPE	TC :	Bonjour, je suis TC1, que puis-je pour vous ?
2.	0	Spe	PRO	CL :	Bonjour TC1, voila depuis samedi j ai changer toute mon instalation live box plus decodeur mais quand on regarde la tv sa s arrete souvent et apres sa repart
3.	2	Spe	INQ	TC :	Comment avez-vous branché votre tv au décodeur ? Avec un cable péritél or hdmi ?
4.	3	Res	STA	CL :	hdmi
5.	2	Imp	STA	TC :	Pouvez-vous tester avec avec un cable péritel ?
6.	5	Res	STA	CL :	pas de peritel sur deco
7.	6	Ela	STA	CL :	ni sur tv
8.	2	Res	PPR	TC :	Je viens de lancer une mise à jour sur votre décodeur.
9.	8	Spe	INQ	CL :	dois je couper la tv
10.	9	Res	STA	TC :	Ce n'est pas nécessaire, seulement le décodeur.
11.	8	Spe	INQ	TC :	Qu'affiche votre décodeur ?
12.	11	Tmp	TMP	CL :	je vais le rallumer vous m'aviez dit de l'éteindre
13.	12	Ack	TMP	TC :	Merci, je patiente.
14.	2	Spe	INQ	CL :	un technicien était passé il y a quelques mois et nous avait dis que sa venait surment du debit vous ne pouvez pas l'augmenter ?
15.	14	Tmp	TMP	TC :	Merci de patienter afin que je puisse effectuer quelques tests.
16.	15	Ack	ACK	CL :	oui merci
17.	16	Ack	STA	TC :	Merci d'avoir patienté.
18.	2	Res	PPR	TC :	En vérifiant de mon coté, je vois que votre ligne présente une instabilité.
19.	18	Spe	INQ	CL :	et alors que pouvons-nous faire ?
20.	19	Res	STA	TC :	Je lance une action afin de corriger ce souci.
21.	20	Ela	STA	TC :	Par contre cette correction va interrompre l'accès internet quelques secondes et notre conversation va être coupée.
22.	20	Ack	ACK	CL :	d'accord
23.	20	Opi	STA	CL :	j'espere que le soucis sera réglé apres
24.	2	Res	PPR	TC :	Je vous invite donc à redémarrer votre décodeur dans 30 minutes.
25.	0	DSpe	INQ	TC :	Avez-vous d'autres demandes ?
26.	25	Res	STA	CL :	non pas pour le moment
27.	0	Clo	CLO	CL :	aurevoir
28.	0	Clo	CLO	TC :	Orange vous remercie de votre confiance. Je vous souhaite une excellente journée.

1. Indique le gouverneur du tour de parole (en utilisant le numéro du tour)

B. Lexique lex5

attente
bonjour
et
entrez
position
l’
conseiller
pour
numtel
mobile
des
bientôt
sont
pouvez
avoir
vous
un
demande
merci
par
:
de
service
est
en
accord
dans
que
êtes
me
cela
journée
il
prix
les
bien
j’
avez
c’
ce
ne
à
tout
ligne
du
charge
puis
sur
remercie
suis
d’
faire
conversation
a
nous
une
pas
plus
qui
orange
_tc1_
file
n’
svp
.
ok
bonne
?
oui
avec
va
au
non
prendre
si
ou
mon
la
je
numéro
s’
votre
le
sosh
-
,
client
nombre
bienvenue
ai
adresse
mais

C. Lexique lex6

a
bien
nombre
-
sur
pour
nous
le
les
êtes
de
oui
il
en
.
une
,
la
d’
est
demande
votre
merci
ce
l’
et
j’
avoir
avec
que
me
un
conversation
c’
je
?
ai
pas
dans
vous
mobile
à

D. Caractéristiques utilisées par l'analyseur en transition

Caractéristiques utilisées par l'analyseur en dépendance en transitions. La colonne origine décrit si la caractéristique est tirée de la pile ou du tampon. Dans la pile, la position 0 correspond à la tête de la pile. Dans le tampon, 0 correspond au mot courant, -1 au mot précédent et 1 au mot suivant. Lorsqu'il y a plusieurs positions, cela indique que la caractéristique est tirée à chaque position.

Caractéristique	Origine	Position	Élément considéré
Tour de parole, Acte de dialogue, Scripteur, Relation dialogique	Pile	0 à 2	–
Tour de parole, Acte de dialogue, Scripteur, Relation dialogique	Pile	0 et 1	Dépendant le plus à droite (dep. droit)
Tour de parole, Acte de dialogue, Scripteur, Relation dialogique	Pile	0 et 1	Dep. droit du dep. droit
Tour de parole, Acte de dialogue, Scripteur, Relation dialogique	Tampon	-2 à 2	–
Distance	Pile	0	Distance avec le mot en position 1 dans la pile
Distance	Tampon	1	Distance avec le mot en position 0 dans la pile
Valence	Pile	0 et 1	Nombre de relations