



HAL
open science

'Nudges' dans l'interaction homme-machine : analyse et modélisation d'un agent capable de nudges personnalisés

Hugues Ali Mehenni

► To cite this version:

Hugues Ali Mehenni. 'Nudges' dans l'interaction homme-machine : analyse et modélisation d'un agent capable de nudges personnalisés. Intelligence artificielle [cs.AI]. Université Paris-Saclay, 2023. Français. NNT : 2023UPASG043 . tel-04382223

HAL Id: tel-04382223

<https://theses.hal.science/tel-04382223v1>

Submitted on 9 Jan 2024

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.

‘Nudges’ dans l’interaction
homme-machine : analyse et
modélisation d’un agent capable de
nudges personnalisés
*Nudges in human-machine interaction: analyse and
modelisation of an agent able of personalized
nudging*

Thèse de doctorat de l’université Paris-Saclay

École doctorale n° 580, Sciences et Technologies de l’Information et de la
Communication (STIC)

Spécialité de doctorat : Informatique

Graduate School : Informatique et sciences du numérique. Référent : Faculté des
sciences d’Orsay

Thèse préparée dans le **Laboratoire interdisciplinaire des sciences du numérique**
(Université Paris-Saclay, CNRS), sous la direction de **Laurence DEVILLERS**,
Professeure, le co-encadrement de **Ioana VASILESCU**, Directrice de recherche

Thèse soutenue à Paris-Saclay, le 20 juin 2023, par

Hugues ALI MEHENNI

Composition du jury

Membres du jury avec voix délibérative

Anne VILNAT Professeure, LISN/CNRS, Université Paris-Saclay	Présidente
Thierry ARTIERES Professeur, ECM/LIS - AMU - CNRS, Université Aix- Marseille	Rapporteur & Examineur
Fabrice LEFEVRE Professeur, CERI/LIA, Université d’Avignon	Rapporteur & Examineur
Bassam JABAIAN Maître de conférences, CERI/LIA, Université d’Avignon	Examineur
Fabrice LE GUEL Professeur, RITM, Université Paris-Saclay	Examineur

Titre : 'Nudges' dans l'interaction homme-machine : analyse et modélisation d'un agent capable de nudges personnalisés

Mots clés : Système de Dialogue Oral, Apprentissage par renforcement, Nudges linguistiques, Apprentissage Few-shot, Interaction Homme-Machine

Résumé : Devant la multiplication et la généralisation d'assistants personnels vocaux (comme Siri ou Alexa), personnifiés sous forme de robots ou non (Pepper, Reeti, enceintes vocales, applications), ainsi que les succès très récents de chatbots tels que ChatGPT, des questions peuvent se poser sur le type et l'étendue de l'influence de ces agents sur leurs utilisateurs. Dans ce travail, nous nous intéressons à l'utilisation des nudges dans le dialogue oral et proposons pour cela la modélisation d'un agent capable de nudges linguistiques et personnalisés grâce à un apprentissage par renforcement.

Les nudges sont des incitations 'douces' dans le design visant à modifier le comportement ou la prise de décision sans apporter cependant de restrictions dans le choix. Introduits puis étudiés à l'origine en économie comportementale, ils reposent souvent sur des biais cognitifs (attention, mémoire, etc.) mais leur utilisation n'est pas encore réglementée et pourrait être amplifiée par un agent oral.

L'originalité de cette thèse est alors d'observer l'impact de stratégies

de nudges linguistiques dans une approche d'apprentissage automatique à travers la construction d'un système de dialogue orienté-tâche. Pour évaluer leur influence dans un tel environnement, nous introduisons des métriques et protocoles et détaillons les résultats de 2 expériences de collecte de données sur les nudges. La première est réalisée dans une école primaire sur une population potentiellement sensible et la deuxième est développée en ligne auprès d'internautes dans une approche automatique continue. Ces expériences permettent de montrer une influence plus grande de la machine (robot ou enceinte) par rapport à un humain et l'efficacité de stratégies de nudges linguistiques orales dès qu'elles sont présentes. Nous montrons enfin comment un système de dialogue oral modulaire, prenant compte de l'émotion de l'utilisateur, peut être entraîné à l'aide de peu de données pour adapter de telles stratégies de nudges à un interlocuteur par apprentissage par renforcement. Ces premiers résultats permettront de nourrir une discussion éthique et de futures pistes de recherche.

Title: Nudges in human-machine interaction: analyse and modelisation of an agent able of personalized nudging

Keywords: Spoken dialogue system, Reinforcement learning, Linguistic nudge, Few-shot learning, Human-machine interaction

Abstract: In front of the multiplication and generalization of vocal personal assistants (such as Siri or Alexa), personified in the form of robots or not (Pepper, Reeti, voice speakers, applications), as well as the very recent success of chatbots such as ChatGPT, questions can be asked about the type and the extent of the influence of these agents on their users. In this work, we are interested in the use of nudges in spoken dialogue and propose the modeling of a conversational agent capable of linguistic and personalized nudges through reinforcement learning.

Nudges are 'soft' incentives in the design aiming at modifying behavior or decision making without bringing any restriction in the choice. Originally introduced and studied in behavioral economics, they often rely on cognitive biases (attention, memory, etc.) but their use is not yet well regulated and could be amplified by an oral agent.

The originality of this thesis is then to observe the impact of linguistic nudges strategies in a machine learning approach through the construction of a task-oriented dialogue system. To evaluate their influence in such an environment, we introduce metrics and protocols and detail the results of two nudge data collection experiments. The first one is conducted in an elementary school on a potentially sensitive population and the second one is developed online with Internet users in a continuous automatic approach. These experiments show a greater influence of the machine (robot or speaker) compared to a human and the effectiveness of oral linguistic nudges strategies as soon as they are present. We finally show how a modular spoken dialogue system can be trained with an emotional detection module using little data to adapt such nudging strategies to an interlocutor. These initial results will feed an ethical discussion on their use in oral chatbots.

Remerciements

Je remercie tout d'abord Laurence Devillers, sans qui ce projet n'aurait pu exister, ainsi que Ioana Vasilescu pour leurs conseils et leur recul sur mon travail.

Je remercie également les personnes avec qui j'ai eu la chance d'interagir au laboratoire LISN et qui m'ont aidé de près ou de loin dans le développement des interfaces et expériences réalisées lors de cette thèse, ainsi qu'à naviguer la vie quotidienne et administrative de la recherche, notamment Eric Bilinski, Sophie Pageau-Maurice, l'équipe AMIC et bien d'autres.

Un grand merci aussi à tous les participants aux expériences, ceux qui ont permis de les organiser et ceux qui ont pris le temps de les annoter.

Pour finir, je tiens à remercier tous les doctorants et stagiaires avec qui j'ai pu partager ces moments de convivialité au laboratoire et dont les conversations ont nourri mes recherches et ma curiosité. Ils ont contribué à créer une ambiance chaleureuse et dynamique et je les remercie pour leur soutien et conseils de rédaction, les échanges culturels, les lectures d'articles et les pauses café... Une pensée particulière pour mon voisin de bureau Paul, toujours à l'écoute, et pour mes compagnons de salle, notamment Natalia pour sa joie de vivre inépuisable et contagieuse. Un dernier mot pour mes amis qui ont suivi de près dans cette entreprise, Hyoseok, Léa-Marie, Loic, Quentin, sans oublier bien sûr ma famille.

Table des matières

Table des figures	6
Liste des tableaux	8
Glossary	10
Introduction	13
1 Introduction	13
1.1 Contexte	13
1.2 Motivations	14
1.3 Questions de recherche	15
1.4 Organisation	18
I Etat de l'art	19
2 Théorie du nudge	21
2.1 Contexte et définition	21
2.2 Cadre et enjeux éthiques	23
2.3 Etude des nudges	24
2.3.1 Systèmes de persuasion	25
2.3.2 Biais cognitifs et émotionnels	26
2.3.3 Influence du support de l'interaction	27
2.3.4 Influences des nudges dans les interactions Homme-machine	28
3 Détection d'émotions et d'états mentaux	35
3.1 Cadre théorique des émotions et états mentaux	35
3.1.1 Théorie des émotions	35
3.1.2 Etats mentaux	38
3.2 Représentations des émotions	39

3.2.1	Représentation discrète	39
3.2.2	Représentation continue	41
3.3	Détection des émotions	42
3.3.1	Différence entre les données actées et naturelles	42
3.3.2	Annotation	42
3.3.3	Représentation des paramètres	45
3.3.4	Algorithmes classiques pour la détection d'émotions	48
3.3.5	Apprentissage Few-shot	53
4	Structure classique d'un système de dialogue oral	56
4.1	Historique et types de systèmes	56
4.2	Structure pipeline	57
4.2.1	Reconnaissance Automatique de la Parole : ASR	58
4.2.2	Compréhension de la Parole : SLU	59
4.2.3	Dialogue Manager : DM	61
4.2.4	Génération en langue naturelle et synthèse vocale	64
4.2.5	Challenges de l'approche pipeline et approches end-to-end	65
4.3	Evaluation	66
4.3.1	Evaluation unitaire	66
4.3.2	Evaluation jointe	67
5	Méthodes d'apprentissage d'un système de dialogue	69
5.1	Apprentissage par renforcement	69
5.1.1	Paradigme général, Modélisation en problème de RL	69
5.1.2	Challenges pour l'apprentissage du DM	76
5.1.3	Apprentissage en ligne efficace	78
5.2	Les modèles de langue Transformers	80
5.2.1	Motivations	80
5.2.2	Principe	80
5.2.3	Modèles de langage préentraînés et applications au dialogue orienté tâche	84
5.2.4	Avancées récentes et alignement de large modèles de langue (LLM)	85
5.3	Systèmes de dialogues personnalisés	86
5.3.1	Types de personnalisation	86
5.3.2	Modélisation du contexte personnel dans les systèmes de dialogue	87
6	Conclusions sur l'état de l'art	90

II	Expériences pour l'analyse de nudges linguistiques	93
7	Conception des nudges dans le dialogue	95
7.1	Objectif 1 : Expérience sur le terrain montrant l'efficacité des nudges des machines auprès d'enfants	95
7.1.1	Préliminaires	96
7.1.2	Protocole	98
7.1.3	Corpus	101
7.1.4	Résultats	101
7.2	Objectif 2 : Expérience automatisée en continue auprès d'inter-nauts montrant l'influence de stratégies de nudges linguistiques oraux, réalisée sur la plateforme du système de dialogue développé	112
7.2.1	Préliminaires	112
7.2.2	Protocole	115
7.2.3	Corpus	119
7.2.4	Résultats	120
8	Structure du système de dialogue développé	125
8.1	Structure du système dans son ensemble	125
8.2	Traitement de l'entrée audio	126
8.2.1	Preprocessing audio	127
8.2.2	Reconnaissance automatique de la parole : ASR	128
8.3	Interprétation émotionnelle et sémantique	128
8.3.1	Module de détection émotionnelle	128
8.3.2	Compréhension sémantique (SLU)	128
8.4	Suivi de l'état du dialogue	130
8.4.1	Informations contextuelles	130
8.4.2	Fusion	130
8.4.3	Représentation probabiliste de l'état	130
8.5	Dialogue Manager	131
8.6	Génération et synthèse de la réplique du système	131
8.7	Entraînement et évaluation du système	132
9	Mise en place d'un modèle d'analyse émotionnelle	133
9.1	Etats mentaux retenus pour la prédiction de nudges	133
9.1.1	Accord Inter-annotateur	134
9.2	Test perceptif	135
9.2.1	Objectifs	135
9.2.2	Protocole	135
9.2.3	Résultats	137
9.3	Module de détection d'émotions avec apprentissage Few-shot	140

9.3.1	Choix du modèle	140
9.3.2	Préentraînement	141
9.3.3	Entraînement en ligne	143
10	Apprentissage par renforcement	147
10.1	Modélisation du problème	147
10.1.1	Modélisation de la récompense	147
10.1.2	Modélisation de l'utilisateur et de l'environnement	148
10.2	Evaluation sur un modèle utilisateur pour l'apprentissage en ligne	152
10.2.1	Simulation de l'apprentissage du DM en ligne	152
	Conclusion	156
11	Conclusions	156
11.1	Réponses aux questions de recherche	156
11.2	Bilan sur les nudges	159
11.2.1	Vérification des hypothèses de travail	159
11.2.2	Limites des nudges	160
11.2.3	Utilisation éthique des nudges	160
11.3	Perspectives et Travaux futurs	162
11.3.1	Perspectives à court terme	162
11.3.2	Poursuite de l'obtention de données	163
11.3.3	Adaptation à d'autres domaines	163
11.3.4	Axes d'étude à long terme	164
11.3.5	Applications réelles	165
	Bibliographie	166

Table des figures

2.1	Codex des biais cognitifs	27
2.2	Représentation des 23 nudges de l'étude [Caraban et al., 2019] positionnés le long des axes de transparence et de réflexion-automatisme ("Figure 3" extraite de l'étude)	30
3.1	Roue des émotions de Plutchik	40
3.2	Représentation d'un circomplexe bi-dimensionnel, avec les dimensions de valence (horizontale) et d'activation (verticale) . . .	41
3.3	Illustration d'un hyperplan h maximisant la marge dans un cas linéairement séparable en dimension 2. L'hyperplan est alors perpendiculaire au segment de droite le plus court séparant un vecteur support de l'hyperplan h et dont la longueur vaut $\frac{1}{\ w\ }$ (en normalisant w et w_0)	49
4.1	Architecture d'un système de dialogue (extrait de [Ferreira, 2015])	58
5.1	Architecture Transformers (extrait de [Vaswani et al., 2017]) . . .	82
5.2	Les 2 mécanismes attentionnels des transformers (extrait de [Vaswani et al., 2017])	83
7.1	Enfant participant à l'expérience avec le robot Pepper	98
7.2	Exemple de déroulé de l'expérience du Dictateur	99
7.3	Proportion des enfants ayant changé leur choix pour chaque examinateur suite à chaque Nudge.	102
7.4	Annotations des états de l'utilisateur (états mentaux, affect burst, registre de langue) avec Elan	105
7.5	Visualisation du site en train d'enregistrer la réponse de l'utilisateur	116
7.6	Engagement des participants	120
8.1	Architecture du système	126

9.1	Visualisation du test perceptif proposé (lorsque la transcription est disponible)	136
9.2	Evolution de la f1-score en simulant l'accès progressif de données en ligne avec le nombre d'utilisateurs	146
10.1	Matrice de corrélation des principaux paramètres d'un tour de parole	150
10.2	Importance des paramètres pour prédire le score de nudges au tour de parole	151
10.3	Evolution des récompenses cumulées pour 2 algorithmes (moyennée sur 10 simulations)	154

Liste des tableaux

5.1	Principaux algorithmes de RL pour l'apprentissage de systèmes de dialogue	77
5.2	Cadre pour catégoriser les paramètres de personnalisation (traduit de [den Hengst et al., 2020], table 1)	88
7.1	Données du corpus collecté	101
7.2	Données collectées durant le jeu du Dictateur	103
7.3	Annotations des états mentaux, affect bursts et registres de langue en classes et nombre de segments audio par classe (sur 91 dialogues)	106
7.4	Caractéristiques au niveau du tour de paroles (valeurs moyennes, durées en s)	107
7.5	Catégorisation des caractéristiques utilisées	108
7.6	Résultats de classification : score f1 obtenus par validation croisée (10-fold) avec les sets 1, 2 et 3	110
7.7	Caractéristiques importantes en % pour chaque set et chaque nudge	110
7.8	Caractéristiques du corpus collecté	119
7.9	Occurrences et succès des stratégies dans l'ensemble du corpus	123
9.1	Scores F1 moyens pour les classificateurs d'états mentaux, d'affect bursts et de registres de langue (<i>10-fold nested cross-validation</i>) avec les caractéristiques du set <i>emobase2010</i> d'OpenSMILE (1582 caractéristiques)	134
9.2	Catégories d'étiquettes macro et micro testées	137
9.3	Accord inter-annotateurs (alpha de Krippendorff)	137
9.4	Accord intra-annotateurs (kappa de Cohen moyen)	138
9.5	Corpus utilisés pour l'entraînement de modules de reconnaissance émotionnelle	141
9.6	Répartition des corpus pour le préentraînement, l'entraînement et l'évaluation	143

9.7	Données émotionnelles annotées de l'Expérience 2 disponibles pour l'entraînement et l'évaluation du modèle <i>feed-forward</i> . . .	144
9.8	Résultats (f1-score moyennée sur 10 runs) des modèles entraînés et évalués sur les données de l'expérience 2 (table 9.7) . . .	145

Glossaire

Acoustique

FFT *Fast Fourier Transform* (Transformée de Fourier rapide) 46

LPC *Linear Predictive Coefficients* (Coefficients de Prédiction Linéaire) 46

MFCC *Mel-Frequency Cepstrum Coefficients* (Coefficients Cepstraux de Fréquence Mel) 46

Analyse de données

CER *Concept Error Rate* (taux d'erreur de concepts) 67

PCA *Principal Component Analysis* (Analyse en Composantes Principales) 45

SER *Slot Error Rate* (taux d'erreur de slots) 67

WER *Word Error Rate* (taux d'erreur de mots) 67

Apprentissage machine

A2C *Advantage Actor-Critic* 74

ACER *Actor-Critic with Experience Replay* 78

CNN *Convolutional Neural Network* (Réseau Neuronal Convolutif) 47

CRF *Conditional Random Fields* (Champ aléatoire conditionnel) 60

DQN *Deep Q Network* (Réseau profond modélisant la fonction Q) 73

FNN *Feedforward Network* (Réseau neuronal à propagation avant) 140, 144

GP *Gaussian Process* (Processus Gaussien) 79

GRN *Gated Recurrent Network* (Réseau récurrent à portes) 80

HIS *Hidden Information State* (Etat d'Informations cachées) 62

HMM *Hidden Markov Model* (Modèle de Markov Caché) 51, 58

IRL *Inverse Reinforcement Learning* (Apprentissage par Renforcement Inversé) 75

kNN *k-Nearest Neighbors* (k-Plus proches voisins) 51

KTD *Kalman Temporal Difference* (Différences temporelles par filtre de Kalman) 73, 79

LLM *Large Language Model* (Large modèle de Langue) 85

LSPI *Least Squares Policy Iteration* 78

LSTM *Long Short Term Memory* (Réseau à mémoire longue et court terme) 47, 80

MDP *Markov Decision Process* (Processus de Décision Markovien) 62
POMDP *Partially Observable Markov Decision Process* (MDP Partiellement Observable) 62, 78
RL *Reinforcement Learning* (Apprentissage par Renforcement) 14, 69
RLHF *Reinforcement Learning from Human Feedback* (RL à partir de feedbacks humains) 75, 86
RNN *Recurrent Neural Network* (Réseau de neurones récurrent) 80
RS *Reward Shaping* (Ajout de récompenses supplémentaires durant le dialogue) 77
SARSA *State-Action-Reward-State-Action* (Etat-Action-Récompense-Etat-Action) 73
SGD *Stochastic Gradient Descent* (Algorithme du gradient stochastique) 53
SVM *Support Vector Machine* (Machine à Vecteurs de Support) 49
TD *Temporal Differences* (Différences Temporelles) 72

Dialogue

ASR *Automatic Speech Recognition* (Reconnaissance automatique de la Parole) 58
DM *Dialog Manager* (Gestion du Dialogue) 58, 61
 DP *Dialog Policy* (Stratégie de Dialogue) 61
 DST *Dialog State Tracking* (Gestion du Dialogue) 61
NLG *Natural Language Generation* (Génération en Langage Naturel) 58, 64
SDS *Speech Dialog System* (Système de Dialogue Oral) 57
SLU *Spoken Language Understanding* (Compréhension du Langage Parlé) 58, 59
SS *Speech Synthesis* (Synthèse Vocale) 58

Paramètres pour l'apprentissage

A ensemble des actions (pour un MDP) 62
 P Fonction de transition entre états 62
 Q Fonction Q (action-valeur) 71
 R Fonction de récompense 62
 S ensemble des états 62
 V Fonction de valeur 71
 α Taux d'apprentissage (*learning rate*) 53
 ϵ Probabilité de choisir une action aléatoire dans le cas d'une exploration ϵ -greedy 74, 153
 γ facteur d'actualisation, $\gamma \in [0, 1]$ 70
 π politique (*policy*) 70

Introduction

Chapitre 1

Introduction

1.1 Contexte

Depuis Turing aux succès récents de chatbots comme ChatGPT, le dialogue avec une machine a longtemps fasciné l'Homme. Aujourd'hui, les chatbots et assistants personnels vocaux sont de plus en plus présents au quotidien (Siri d'Apple, Google Home, Amazon Echo, etc.), commercialisés en général sous forme d'enceintes mais aussi récemment de robots miniatures (Astro d'Amazon, Misa, Buddy de Blue Frog Robotics, etc.). Qu'ils soient matérialisés ou non, ces agents conversationnels ont tendance à être personnifiés (persona, avatars, robots) cherchant ainsi à maximiser l'engagement de leurs utilisateurs, jouant pour cela sur le biais naturel de l'Homme d'anthropomorphisme. Souvent à destination de la sphère privée, la généralisation de ces chatbots suscitent alors certaines questions sur le type et l'étendue de l'influence qu'ils pourraient avoir sur leurs utilisateurs.

Tout d'abord, un biais ou une différence de comportement envers un interlocuteur machine plutôt qu'humain peut être observé. En outre, dans la plupart des décisions du quotidien, de nombreux biais décisionnels, souvent peu conscients, ont pu être mis en évidence comme la pression sociale, l'attention, l'aversion pour la perte, etc. Ainsi, les utilisateurs de ces assistants vocaux risqueraient même de ne pas réaliser les différents effets et incitations auxquels ceux-ci pourraient les soumettre. Dans cette étude, nous nous intéressons tout particulièrement à un certain type d'influence de plus en plus présent dans de nombreux domaines : les *nudges*.

Les *nudges* sont des incitations 'douces' dans le design visant à modifier le comportement ou la prise de décision sans apporter cependant de restrictions dans le choix. Introduits puis étudiés à l'origine en économie comportementale,

ils reposent souvent sur des biais cognitifs (attention, mémoire, etc.) mais leur utilisation n'est pas encore bien réglementée.

Depuis ces dernières années, ils ont connu des applications dans de multiples domaines : politique, finance, éducation, santé, écologie, etc. Leur succès vient notamment du fait qu'ils peuvent représenter des solutions efficaces et peu coûteuses tout en préservant le choix des individus. Par exemple, des règles par défaut ou l'utilisation de normes sociales sont parfois plus efficaces qu'une incitation économique. Très libres de forme, ils peuvent faire appel aux 5 sens et être par exemple graphiques (comme les avertissements dissuasifs sur les paquets de cigarette) ou encore audio (tel un escalier à Stockholm transformé en piano pour encourager les passants à prendre les marches au lieu de l'escalator).

En dehors de leurs utilisations commerciales massives pour la prise de décision sur internet (prix barré, avertissement du nombre de places restantes, etc.), les nudges se retrouvent souvent sous forme de *gamification*, qu'elle soit urbaine, éducationnelle ou pour la santé (comme les nutri-scores).

Les nudges apparaissent donc comme un outil fréquent dont la transparence et les implications sont encore mal réglementées. Dans cette étude, nous nous intéresserons surtout à leur efficacité en tant que stratégies d'influence à l'oral et en particulier lorsqu'ils sont employés dans un système de dialogue oral.

Les systèmes de dialogue (ou agents conversationnels) ont connu de nombreux développements depuis les débuts de la recherche en intelligence artificielle. Ils sont passés de système à base de règles, comme le célèbre ELIZA (Weizenbaum, 1966) qui simulait un psychothérapeute en transformant en questions les répliques précédentes, à des modèles statistiques mettant à profit l'émergence des *big data* et ils tendent à présent à utiliser du deep learning. Les stratégies de dialogue du système sont alors souvent présentées sous la forme d'un problème d'apprentissage par renforcement ou *Reinforcement Learning* (RL) et cette approche nous a paru adaptée à notre problématique de modéliser un agent capable d'influencer par des nudges une personne. Dans ce sens, notre objectif se rapproche du nombre croissant de systèmes de dialogue personnalisés, c'est-à-dire cherchant à s'adapter à l'utilisateur individuellement.

1.2 Motivations

Ce travail s'inscrit dans le cadre du projet interdisciplinaire *Bad Nudge Bad Robot* de l'Institut DATAIA rassemblant des informaticiens, linguistes, économistes et juristes et soulèvent des enjeux de recherche tels que :

- la mise en œuvre de stratégies de nudging avec des agents conversationnels embarqués sur différentes plateformes (robots, enceinte vocale)
- le développement d'outils d'évaluation de l'impact de nudges

Il fait également partie de la chaire HUMAINE : HUman-MACHine Affective Interaction & Ethics dirigée par Laurence Devillers. Les enjeux finaux auxquels participent ce manuscrit sont ainsi de permettre une réflexion sur les enjeux technologiques et éthiques des usages positifs ou négatifs des nudges dans les interactions sociales et plus généralement sur la confiance accordée aux agents conversationnels.

1.3 Questions de recherche

L'originalité de ma thèse est de proposer une approche d'apprentissage automatique des nudges dans un système de dialogue oral pour influencer un utilisateur. Si le nudge est un concept bien étudié en économie comportementale, il l'est encore assez peu en linguistique et sa modélisation avec des méthodes de machine learning est encore peu analysée. Si les nudges sont des mécanismes pouvant être utilisés de manière systématique, indifféremment de la personne, nous nous intéressons ici également aux nudges personnalisés et appartenant à la catégorie *AI-enhanced* (augmentés par l'Intelligence Artificielle).

Cette approche nous a amenés tout d'abord à observer l'impact de stratégies de nudge linguistique dans le dialogue et collecter des données de comportements associés (émotions, profil, état mental de l'utilisateur) afin d'entraîner des modèles permettant d'extraire les informations contextuelles et paralinguistiques pertinentes pour une prédiction de nudges. Nous proposons alors d'utiliser des méthodes d'apprentissage par renforcement pour adapter à l'utilisateur des stratégies de nudges à l'oral dans un système avec une architecture modulaire. Une autre originalité de mes travaux a été de chercher une tâche qui pouvait engager les utilisateurs dans une interaction parlée. Mettre en œuvre ce système pose alors un grand nombre de questions de recherche.

Comment étudier les nudges linguistiques et leur personnalisation dans le dialogue ?

Nos hypothèses de travail sur les nudges (énoncées section 2.3.4) sont les suivantes :

1. Nous supposons que le type d'agent (humain, robot ou enceinte) peut avoir une influence sur l'efficacité des nudges.

2. Nous faisons l'hypothèse qu'un système de dialogue utilisant des questions de nudges oraux pourrait changer significativement le choix d'un utilisateur pendant le dialogue comparativement à des questions sans nudges.
3. Nous supposons qu'une adaptation personnelle de stratégies de nudges avec RL est significativement plus efficace pour influencer une personne.
4. Pour estimer l'état actuel du dialogue ainsi que l'état mental et émotionnel de la personne, nous choisissons d'extraire certaines informations émotionnelles, paralinguistiques et contextuelles que nous supposons pertinentes pour prédire les nudges.

Comment tester l'impact de la représentation de l'agent dans l'interaction sur l'efficacité des nudges ?

Afin de creuser davantage l'hypothèse 1 sur les nudges, nous avons testé différentes représentations d'un agent oral automatisé : robot humanoïde, enceinte vocale (non humanoïde) et robot imaginaire, figuré par une image. Les différences d'efficacité des nudges relevées dans chacun de ces scénarii permettront alors d'apporter des premiers éléments de réponses.

Comment créer un environnement de recherche pour tester les nudges linguistiques ?

Afin de pouvoir évaluer concrètement l'impact de stratégies de nudges dans le dialogue, nous nous sommes focalisés sur leur apprentissage dans un système de dialogue orienté tâche. Cette tâche a été choisie de manière à permettre une évaluation numérique des nudges sur un domaine engageant les participants, ici des questions d'écologie.

L'influence d'un nudge pouvant dépendre de son implémentation, nous avons essayé de la contraindre au maximum pour limiter le nombre de biais associés. Nous avons ainsi utilisé pour notre système une génération de réponses déterministes (retrieval-based), afin de tester uniquement l'impact du choix de la stratégie de nudge.

Sur quelles données entraîner le système ?

S'il existe de plus en plus de corpus disponibles pour l'entraînement de systèmes de dialogue, ceux-ci ne correspondent hélas pas à notre sujet d'étude. Cela implique une collecte de données spécifiques aux nudges qui complique l'évaluation finale du système. Pour cela, nous détaillons les résultats de 2 expériences de collecte réalisées sur les nudges, une dans une école auprès d'enfants et l'autre en ligne auprès d'internautes majoritairement adultes. En outre,

nous présentons l'interface et protocole mis en place pour réaliser une collecte automatique en ligne en continu ainsi que l'évaluation finale du système.

Quel apprentissage est possible avec peu de données ?

Ne disposant que peu de données spécifiques au nudge oral, une contrainte de cette thèse est également d'apprendre des stratégies de nudge adaptées avec peu de données. Cela pose un certain nombre de challenges pour faire converger rapidement notre système. Nous nous sommes donc concentrés sur des approches efficaces par échantillon et du few-shot learning.

Comment évaluer l'influence du nudge ?

Les nudges visant à inciter une modification du comportement ou de la prise de décision, leur évaluation nécessite une comparaison à un état initial ou à une première prise de position. Afin de mesurer numériquement et quantitativement l'impact de ce changement, nous avons utilisé des questions faisant intervenir le choix explicite d'un score, dont nous pouvons analyser l'évolution en introduisant des métriques simples. Le changement émotionnel de l'utilisateur suite à un nudge est également pris en compte dans l'état du dialogue et sous forme de faibles récompenses implicites dans l'apprentissage par renforcement.

Comment évaluer les systèmes de nudges linguistiques ?

L'évaluation du système consiste alors à mesurer l'influence de nudges sur la durée de plusieurs interactions orales. Dans ce travail, nous mesurons l'efficacité d'une stratégie de nudge à travers ce que nous appelons une "question de nudge". L'idée est alors de comparer l'influence de plusieurs stratégies de nudges sur une même question.

Le nudge étant par design une "incitation douce", son utilisation dans le dialogue se révèle relativement brève. Afin d'évaluer l'éventuelle évolution des réactions d'une personne au cours du dialogue et pour augmenter les données d'étude, nous avons choisi d'enchaîner plusieurs "questions de nudges" pour chaque participant. L'évaluation du système se fait ainsi sur l'ensemble des questions de nudges.

Avec un apprentissage par renforcement, le succès et la rapidité d'adaptation du système se reflète alors dans la récompense en fin de dialogue. A chaque tour de parole, une récompense est obtenue en fonction du changement de comportement suite au nudge ainsi que de la brièveté du dialogue. L'évolution de cette récompense ainsi que ses valeurs en fin d'apprentissage permettent alors de conclure sur l'efficacité du système de nudges.

Est-ce éthique de faire des systèmes qui nudgent ?

L'utilisation des nudges dans les systèmes de dialogue soulève un certain

nombre de points éthiques, que nous développerons dans les sections 2.2 et 11.2.3. De manière générale, la question de l'influence des systèmes numériques sur l'être humain, leur transparence, ainsi que le respect des données personnelles acquises par ceux-ci s'inscrit dans le domaine de l'éthique du numérique ¹, étudiés par des organismes comme le comité national pilote d'éthique numérique. L'emploi des nudges doit ainsi adhérer aux règles de RGPD sur le droit des données et veiller à respecter des principes de dignité, justice, bienveillance et liberté de choix. Cette contrainte éthique doit idéalement être pensée dès la conception du système, selon l'approche *ethics by design* par exemple.

Bien qu'encore non réglementés, les nudges peuvent cependant être un outil bénéfique et efficace lorsqu'ils respectent des règles de transparence ainsi que les valeurs souhaitées de la personne soumises à ceux-ci. De nombreux domaines ont ainsi pu profiter de leurs applications. Dans un cadre numérique, nous pouvons notamment citer des domaines comme l'écologie (impression recto-verso par défaut), la santé (application de sport, nutri-score) ou l'éducation. Plus précisément, plusieurs groupes comme IEEE et CEN-CENELEC sont en train de construire des standards autour du nudge pour l'encadrement du déploiement de ces stratégies dans des systèmes intelligents et autonomes (*IEEE P7008- Standard for Ethically Driven Nudging for Robotic, Intelligent and Autonomous Systems* ² et *AFNOR - JTC21/CEN-CENELEC/WG4 : AI-enhanced nudge*).

1.4 Organisation

Ce manuscrit est structuré en 2 parties. La première section s'attache à établir le cadre théorique de l'étude en rappelant l'état de l'art sur le nudge, les émotions et les systèmes de dialogue notamment ceux adaptatifs et avec apprentissage par renforcement avec peu de données. La deuxième partie présente nos résultats expérimentaux, l'architecture du système développé, ses modules et leurs apprentissages. Nous concluons enfin sur les hypothèses soulevées durant cette étude et ouvrons sur les perspectives futures d'analyse du nudges dans le dialogue.

¹https://www.ccne-ethique.fr/sites/default/files/2022-02/manifeste_cnpen.pdf

²<https://standards.ieee.org/ieee/7008/7095/>

Première partie

Etat de l'art

Cette section permet de poser le cadre théorique nécessaire à la compréhension de cette thèse. Le Chapitre 2 explique ainsi le nudge, ses enjeux éthiques et les hypothèses que nous formulons sur l'efficacité et le fonctionnement de ses nudges durant notre étude. Le Chapitre 3 détaille les recherches réalisées pour représenter et classifier des émotions. Le Chapitre 4 rappelle la structure classique d'un système de dialogue modulaire et comment il peut être évalué. Enfin, le dernier Chapitre se concentre sur l'apprentissage dans les systèmes de dialogue et décrit les modélisations possibles de ceux-ci en problème d'apprentissage par renforcement, les avancées récentes permises par l'utilisation de modèles de langue et comment cet apprentissage peut être rendu adaptatif et personnalisé.

Chapitre 2

Théorie du nudge

Si le nudge est un concept apparu pour la première fois en économie comportementale [[Thaler and Sunstein, 2008](#)], il repose néanmoins sur des modèles et principes de psychologie. Dans ce chapitre, nous allons expliquer son origine, le cadre dans lequel nous pouvons l'étudier ainsi que les études qui ont pu être réalisées à son sujet.

2.1 Contexte et définition

La notion de nudge a été introduite pour la première fois en 2008, en économie comportementale, par Richard Thaler et Cass Sunstein (prix Nobel d'économie en 2017) dans leur livre *Nudge* [[Thaler and Sunstein, 2008](#)]. Thaler et Sunstein partent du constat que l'hypothèse courante en économie de l'homo economicus, c'est-à-dire l'idée que nous sommes des êtres maximisateurs agissant parfaitement rationnellement, est en réalité très erronée. En effet, cette représentation théorique du comportement de l'être humain, à la base du modèle néoclassique en économie, ne prend pas en compte l'ensemble des biais et limitations humains et individuels que nous avons. Pour expliquer ces biais, Thaler et Sunstein font appel à un modèle utilisé en psychologie et neuroscience : celui des "Deux Systèmes" (ou Système 1 et Système 2 dans la littérature en psychologie de Kahneman [[Kahneman and Egan, 2011](#)]). L'idée est que notre cerveau aurait deux modes de pensée ou fonctionnement. L'un serait automatique (Système 1), instinctif, rapide, ne demanderait pas beaucoup d'effort et permettrait de faire appel à nos automatismes et compétences déjà bien apprises. L'autre serait réfléchi (Système 2), conscient, plus lent, demanderait plus d'effort et ferait appel à notre sens critique. Ces 2 systèmes coopèrent au quotidien. Néanmoins le Système 2 nécessitant un effort d'attention beaucoup plus grand, nous aurions tendance à lui préférer le Système 1 si possible, ce qui per-

mettrait d'expliquer certaines de nos décisions irrationnelles prises rapidement. Ainsi, une grande partie de nos décisions du quotidien serait faite et gérée automatiquement, leurs buts pouvant même rester inconscients parfois [Bargh et al., 2001]. Nos décisions s'appuient alors souvent sur des heuristiques (c'est-à-dire des règles simples de comportements donnant généralement une solution rapide et satisfaisante).

Thaler et Sunstein s'inspirent notamment des travaux de Tversky et Kahneman [Tversky and Kahneman, 1974]. Ceux-ci identifient en particulier 3 heuristiques pour des prises de décisions en présence d'incertitude ainsi que les biais associés à chacune d'elles, ces biais émergeant de l'alternance entre ces Systèmes 1 et 2. Tversky et Kahneman s'intéressent en effet à notre façon subjective au quotidien d'estimer la probabilité ou vraisemblance d'un événement incertain (élection, culpabilité d'un suspect, etc.). Ils présentent alors ces 3 heuristiques comme des "raccourcis de jugements" ("judgmental shortcuts") permettant une décision rapide mais visiblement biaisée. La première est l'heuristique de "représentativité" : la façon dont nous rapprochons un événement incertain d'une représentation mentale connue, potentiellement stéréotypée. La seconde est l'heuristique de "disponibilité" : notre tendance à utiliser les informations qui nous viennent rapidement et facilement à l'esprit pour prendre des décisions concernant l'avenir (e.g. : une publicité ou un reportage sur des gagnants au loto peut nous rendre confiant sur nos propres chances de gagner). La dernière est l'heuristique "d'ancrage" : lorsqu'on se fie de manière forte à une information préexistante ou à la première information trouvée pour prendre une décision.

Ces constatations vont ainsi servir de point de départ à Thaler et Sunstein pour identifier les différents biais cognitifs relatifs à nos prises de décisions. Ce qu'ils proposent alors est de mettre à profit cette connaissance des biais cognitifs pour aider à influencer positivement les gens dans leurs choix à l'aide de nudges. Dans leur ouvrage [Thaler and Sunstein, 2008], ils définissent un nudge comme : "tout aspect de l'architecture de choix qui modifie le comportement des personnes de manière prévisible sans interdire aucune option ni modifier de manière significative leurs motivations économiques".

Conscients que nous ne prenons pas toujours les meilleures décisions dans notre propre intérêt, leur idée est de nous inciter à mieux choisir à travers les nudges sans restreindre en aucune manière le choix initial. Les auteurs identifient leur démarche comme une forme de paternalisme libertaire. Le but des nudges est ainsi de préserver la liberté de décision mais d'utiliser les biais sociaux, attentionnels et mémoriels par exemple pour orienter positivement nos choix dans des domaines comme la santé, l'éducation, etc.

Pour Sunstein, cette définition du concept de nudges peut englober de très nombreux aspects de la vie quotidienne et se révéler étonnamment efficace. Dans [Sunstein, 2014], il explique ainsi que des nudges comme l'utilisation de règles par défaut (cocher une option par défaut) ou l'utilisation de normes sociales peuvent parfois être plus influents qu'une forte incitation économique. Par exemple, selon lui, aux Etats-Unis, les exigences en matière de divulgation et l'utilisation de règles par défaut ont protégé les consommateurs contre de graves préjudices économiques, économisant ainsi des millions de dollars. Un autre exemple est le fait qu'informer les gens sur leur consommation d'électricité, et comment elle se compare à celle de leurs voisins, peut produire la même augmentation de conservation qu'une hausse importante du coût de l'électricité.

Le nudge apparaît ainsi comme un outil efficace dont l'utilisation soulève certains enjeux éthiques, tout particulièrement lorsqu'il est employé dans des interactions homme-machine.

2.2 Cadre et enjeux éthiques

Face au nombre toujours croissant d'agents conversationnels, certaines questions éthiques se posent sur la confiance accordée aux machines [Rahwan et al., 2019] et sur les biais qui pourraient leur être associés. Dans [Hidalgo, 2021], Hidalgo explique notamment que les gens jugent différemment les actions d'une machine et celles d'un humain. Selon cette étude, une machine est davantage jugée sur son résultat tandis qu'un humain est davantage jugé sur ses intentions. Par exemple, un accident sera plus facilement pardonné à un humain qu'à une machine. A l'opposé, une erreur perçue comme intentionnelle sera plus facilement excusée si elle vient d'une machine.

Thaler et Sunstein nous mettent justement en garde sur l'utilisation éthique des nudges et surtout des *sludges* [Thaler and Sunstein, 2019], c'est-à-dire des nudges qui encouragent des prises de décisions à l'encontre des souhaits de la personne. De manière générale, la règle éthique pour qu'un nudge soit légitime est la suivante : il faut que le comportement qu'il tente de générer chez l'interlocuteur soit considéré comme bénéfique pour la personne et qu'il ait ainsi toujours le choix. Ce concept peut alors être rapproché à une notion de "paternalisme libertaire" [Heilmann, 2014]. Le système cognitif "réflexif" de la personne accepte alors en principe le comportement suggéré mais son système cognitif "automatique" ne le fait pas. C'est là qu'intervient alors le nudge qui va jouer sur les différents biais cognitifs de la personne et agir ainsi sur son système cognitif "automatique". Pour que le nudge reste libertaire, la personne doit aussi avoir la possibilité de changer son choix fait sous l'influence de celui-ci, selon

[Heilmann, 2014]. Bien que cet aspect soit pris en compte dans notre démarche (à travers notamment une phase de sensibilisation après une expérience sur le nudge), il n'est pas toujours explicite pendant nos expériences pour éviter trop de redondances.

Un autre point éthique important est la question de la *transparence*. Dans [Sunstein, 2014], Sunstein précise que de nombreuses nations se sont intéressées récemment aux nudges en raison de leur efficacité pour des coûts généralement faibles mais aussi car ils permettent d'obtenir des résultats sans imposer de règles ou de lois puisqu'ils maintiennent une forme de choix. Il indique ainsi que les Etats-Unis et le Royaume-Uni (pour n'en citer que 2 parmi tant d'autres) ont créé chacun une équipe spécialisée sur l'analyse comportementale. Au Royaume-Uni, l'équipe *Behavioural Insights Team* est notamment appelée la "Nudge Unit". Sunstein insiste alors sur l'importance de la transparence, argumentant qu'elle peut aider à obtenir des résultats de nudge efficaces (comme par exemple en utilisant des méthodes de simplification sur certains documents administratifs).

A ce sujet, notre objectif étant d'étudier l'efficacité de stratégies de nudges pouvant potentiellement être employées à l'insu de l'utilisateur, nous ne précisons pas le focus exact de nos expériences lors de leur déroulement mais procédons à une phase de sensibilisation ensuite. En effet, la tromperie, en particulier lorsqu'elle est répétée, peut engendrer un biais de méfiance, qui peut affecter les modes de raisonnement de la personne Scher [2016]. L'objectif précis de nos expériences n'est donc révélé qu'une fois l'ensemble des expériences terminées, lors d'une phase de sensibilisation. L'aspect éthique de nos expériences est cependant pris en compte dès leur conception (en faisant appel à un comité d'éthique par exemple, pour l'expérience réalisée en école primaire).

Des études récentes s'intéressent également à l'aspect moral des nudges *AI-enhanced* (augmentés par l'IA).

2.3 Etude des nudges

L'étude des comportements humains dans les interactions Homme-machine a fait l'objet de nombreuses recherches. Cette thèse se concentre surtout sur l'efficacité des stratégies de nudges, en particulier au travers d'une machine.

2.3.1 Systèmes de persuasion

Bien que le nudging puisse se rapprocher d'une forme de persuasion, il possède cependant des contraintes éthiques et de construction plus fortes. Le nudge, dans des conditions éthiques, ne sert pas à persuader en soi mais plutôt à aiguiller un comportement dont la personne a conscience du bénéfique et ce, sans restreindre sa liberté de choix. Ainsi, le but du nudging peut être davantage d'aider à maintenir une décision déjà prise par une personne (comme par exemple l'aider à se concentrer sur un sujet d'éducation qu'il souhaite apprendre) plutôt que de la "persuader" de cette décision.

Néanmoins, la similarité entre les nudges et les systèmes de persuasion offre une référence intéressante. Les émotions jouent notamment un rôle important dans ces deux concepts. La persuasion repose sur 3 méthodes principales selon Aristote : le *λόγος* (logos : la logique, l'art de la parole), l' *ἦθος* (ethos : la manière d'être, l'attitude de la personne) et le *πάθος* (pathos : l'appel à l'émotion). De nombreuses recherches ont alors prouvé le rôle des émotions dans l'efficacité perçue des arguments de persuasion [DeSteno et al., 2004][Kleef et al., 2015][Wang et al., 2015]. Dans [DeSteno et al., 2004], les auteurs montrent notamment que formuler un message persuasif avec une connotation émotionnelle similaire à celle du destinataire le rend plus efficace et que cet état émotionnel du destinataire peut alors être induit en amont, par exemple, par la lecture d'un article de magazine chargé émotionnellement (de tristesse ou de colère dans leur étude).

Certaines études ont également mis en avant le fait que l'efficacité des stratégies de persuasion semble varier selon l'individu [Kaptein et al., 2010]. La théorie ELM (*Elaboration Likelihood Model*) [Petty and Cacioppo, 1986] indique notamment qu'une personne sera davantage sensible à un message persuasif si elle a la motivation et la capacité de traiter l'information contenue dans celui-ci. Différents systèmes de persuasion "personnalisés" ont alors été proposés.

L'adaptation à la personne peut se faire tout d'abord en un seul tour de dialogue, comme dans [Lukin et al., 2017] où les auteurs montrent l'influence de certains traits de personnalité sur la tendance à changer d'opinion à propos de sujets politiques ou sociaux. Elle peut également se faire tout au long d'un dialogue entretenu avec un système persuasif personnalisé, de manière dynamique et permettant ainsi de tester l'enchaînement de stratégies. Weber et al. [2020] présente un agent virtuel persuasif capable d'adapter son comportement multimodal (gestes, expressions faciales, voix synthétique) à l'utilisateur en fonction de son feedback explicite. Wang et al. [2019] ont cherché, quant à eux, à relier l'efficacité de certaines stratégies de persuasion au profil démographique et psychologique de l'individu dans le but de construire un système de dialogue persuasif personnalisé. Ces différents travaux nous confortent ainsi

dans notre approche d'adapter notre agent capable de nudges à l'individu.

Nous ne citons ici que des résultats de recherches relatifs à la persuasion mais l'adaptation à l'individu dans les systèmes de dialogue peut se faire dans de nombreux domaines et est développée au Chapitre 5.

Pour conclure, si nudges et persuasion jouent sur des mécanismes similaires, la structure argumentative est cependant moins présente chez le nudge qui repose le plus souvent sur des biais cognitifs pour inciter plutôt que convaincre.

2.3.2 Biais cognitifs et émotionnels

Les nudges sont en général définis par rapport à des "biais cognitifs", c'est-à-dire une distorsion de l'information lorsqu'elle entre ou sort de notre système cognitif. Terme introduit à l'origine par Kahneman et Tversky, les biais cognitifs font appel à de nombreux aspects mentaux et sensoriels différents : la mémoire, l'attention, la logique, les relations sociales, etc. Ils peuvent notamment être regroupés en 4 catégories selon le codex de B. Benson (cf Figure 2.1 ci-dessous). Ces 4 catégories regroupent respectivement les biais causées par :

- trop d'information
- une information n'ayant pas assez de sens pour la personne
- un besoin d'agir vite
- des limites mémorielles.

Le site *The decision lab* ¹ recense également la plupart des biais cognitifs pertinents en économie comportementale.

La liste des biais cognitifs et nudges pouvant être longue, il a alors fallu choisir les nudges à étudier en particulier dans le cadre de cette thèse. Si de nombreux nudges visuels peuvent déjà se retrouver au quotidien (parcours fléché à Ikea, nombre de kcal dépensés sur les marches d'escalier, etc.), cette thèse a pour objectif d'étudier les nudges à travers un système de dialogue oral, et donc d'essayer de limiter la présence de nudges visuels.

A ces biais cognitifs peuvent s'ajouter des biais davantage "émotionnels" lorsque la décision ou les connaissances sur un sujet sont modifiées en raison d'un processus émotionnel. En effet, selon nos groupes sociaux et notre environnement, nous construisons des liens affectifs et des croyances sur une multitude de sujets, ce qui peut biaiser naturellement certaines études. Nos

¹<https://thedeisionlab.com/biases>

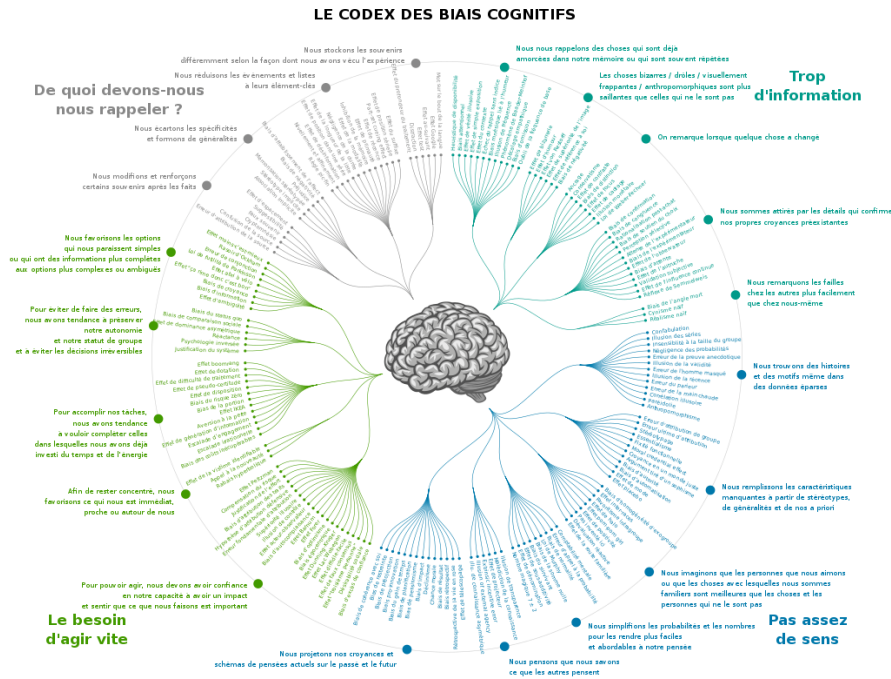


FIGURE 2.1 : Codex des biais cognitifs

émotions influencent également la mémoire d'un événement. Par exemple, la valence émotionnelle positive de mots faciliterait des tâches de mémoire chez l'enfant [Syssau and Monnier, 2012].

2.3.3 Influence du support de l'interaction

Les nudges sont des mécanismes pouvant être déployés sur de nombreux supports à la fois physiques et digitaux. Nous nous intéressons ici principalement à leur utilisation en présence de machines.

Les nudges digitaux en particulier se sont démultipliés ces dernières années et interviennent à de nombreux niveaux :

- dans la présentation et l'architecture de l'interface choisie (nudges d'ancrage par exemple)
- dans le contenu utilisé (options par défaut dans un questionnaire, *gamification* d'applications éducatives, introduisant un système de points et de classement entre les utilisateurs, comme l'application d'apprentissage de langues DuoLingo)

- dans le suivi dynamique de l'utilisateur (rappels encourageants, alertes et notifications d'avancement)

L'impact de nudges d'interface pour la délibération en ligne a par exemple fait l'objet d'une étude [Menon et al., 2020] dans laquelle les auteurs évaluent différentes techniques d'ancrage et de partitionnement pour réduire la charge cognitive. Ils montrent notamment que préciser le nombre de mots comme ancrage (c'est-à-dire en amont) aide significativement.

Etant donné les biais existants envers la façon dont sont perçues les actions des machines [Hidalgo, 2021], une question peut se poser sur l'impact d'un agent robotisé sur les nudges, que cet agent soit physique ou non. L'influence de nudges en présence d'un agent automatique a par exemple déjà pu être observée dans le cadre d'une étude cherchant à pousser les touristes d'un hôtel vers des intentions de comportements pro-environnementaux [Tussyadiah and Miller, 2019]. Selon cette étude, des feedbacks sociaux indiquant la présence ou le regard même non palpable d'un agent virtuel étaient efficaces comme stratégie de nudge. Par ailleurs, les auteurs ne relèvent pas de différence particulière entre l'influence d'un agent virtuel et celle d'un robot.

2.3.4 Influences des nudges dans les interactions Homme-machine

Bien qu'il n'existe pas à notre connaissance de travaux utilisant un chatbot oral avec apprentissage pour étudier les nudges, de nombreuses études ont déjà été réalisées sur l'influence de nudges, notamment dans l'Interaction Homme-Machine. Une analyse [Caraban et al., 2019] réalisée sur 71 articles a notamment cherché à identifier les différents mécanismes de nudges développés dans ce domaine. Elle en identifie alors 23 qu'elle regroupe en 6 catégories :

- *Faciliter* : Il s'agit de favoriser un comportement en diminuant l'effort mental ou physique associé. Les nudges de cette catégorie reposent sur le biais du status-quo, décrivant notre préférence à l'inertie et au moindre effort. Ils incluent notamment l'option par défaut (*default rule*) consistant à proposer le comportement à nudger par défaut [Egebark and Ekström, 2016] ou d'autres biais d'ancrage cherchant à rendre l'option désirée plus accessible, comme le *positioning* [Kammerer and Gerjets, 2014].
- *Confronter* : Ces nudges cherchent à dissuader un comportement généralement non-réfléchi d'un individu en générant un doute pour celui-ci. En effet, selon le biais d'aversion au regret (regret aversion bias), une

personne devient plus prudente lorsqu'elle perçoit un risque. Cela peut être accompli à l'aide d'avertissements plus ou moins explicites ou même d'une simple pause volontaire [Wang et al., 2014].

- *Tromper* : Cette catégorie regroupe des mécanismes trompeurs changeant la perception du comportement à induire ou de ses alternatives. Elle englobe notamment des techniques d'ancrage (illusions visuelles, *decoy effect* : ajouter simultanément une alternative inférieure) ou mémorielles (*peak-end rule* : le moment le plus intense et le dernier moment d'une expérience passée sont les mieux retenus [Cockburn et al., 2015])
- *Influence sociale* : Les stratégies dans cette catégorie jouent en général sur le désir des gens de se conformer à ce qu'ils pensent être attendus d'eux. Elles s'appuient ainsi sur des biais de réciprocité ou d'instinct de groupe (*herd instinct*) [Eckles et al., 2009].
- *Peur* : Ces nudges reposent sur les sentiments d'incertitude et de peur de perte (*loss-aversion*) de l'individu. Par exemple, présenter l'option à choisir comme rare ou limitée (*scarcity bias*) poussera l'individu à la choisir davantage par "peur de manquer" [Cialdini, 1987].
- *Renforcer* : Il s'agit de renforcer un comportement en augmentant l'attention de l'individu dessus. Cela peut être réalisé à l'aide de messages à intervalles adaptés, de feedbacks ambiants ou à travers de l'empathie (heuristique affective [Slovic et al., 2007]).

Reprenant la classification de [Hansen and Jespersen, 2013], les auteurs de l'étude ont également représenté les 23 nudges identifiés sur 2 dimensions : le mode de pensée engagé (automatique ou réfléchi) et la transparence du nudge (si l'utilisateur peut percevoir les intentions et les méthodes liées à celui-ci)(cf Figure 2.2).

Ils remarquent ainsi que seulement 16 % des 74 exemples de nudges rencontrés dans leurs analyses sont "non-transparents" et "automatiques" (quadrant bas-gauche) et correspondent alors à une forme de manipulation du comportement (behaviour manipulation). Selon les auteurs, ces nudges pourraient ainsi soulever des questions éthiques même s'ils concluent que leur analyse ne montre pas en général de conflit entre transparence et nudging. La majorité des nudges rencontrés dans leur étude encouragent plutôt à un choix "réfléchi" et sont en général "transparents" (quadrant haut-droite).

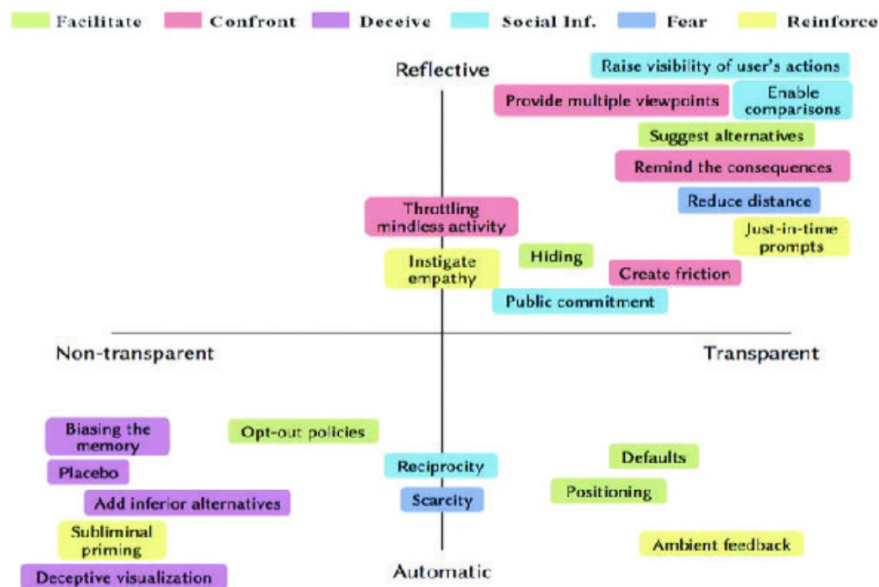


FIGURE 2.2 : Représentation des 23 nudges de l'étude [Caraban et al., 2019] positionnés le long des axes de transparence et de réflexion-automatisme ("Figure 3" extraite de l'étude)

Efficacité des nudges

Parmi le large choix de nudges existants, il a fallu sélectionner les nudges audio à étudier spécifiquement pour cette thèse. Pour des soucis de performance, nous avons recherché des exemples de nudges ayant prouvé être "efficaces". Une méta-analyse (96 expérimentations) en 2019 avait notamment cherché à mesurer l'efficacité de nudges poussant vers une alimentation plus saine [Cadorio and Chandon, 2019]. Selon elle, les nudges axés sur la "réflexion" (ex : un nutriscore) semblent en moyenne moins influents que les nudges "émotionnels" (ex : insistant sur le plaisir de manger), qui semblent eux-mêmes moins efficaces que les nudges de "comportement" (ex : réduire la taille des contenants pour servir la nourriture).

Sunstein propose lui-même une liste de "10 nudges importants" dans un court guide sur le nudge [Sunstein, 2014] avec certaines remarques pour maximiser leurs efficacités. Les 10 nudges qu'il propose sont les suivants :

- *les règles par défaut* (par exemple, l'inscription automatique à des programmes, notamment d'éducation, de santé ou d'épargne)
- *la simplification*
- *l'utilisation des normes sociales* (comme le *peer-effect*, en mettant l'accent

sur ce que font la plupart des gens, comme par exemple : "la plupart des gens prévoient de voter")

- *l'augmentation de la facilité et de la commodité* (par exemple, en rendant visibles les options à faible coût ou les aliments sains)
- *l'ajout d'information ou disclosure* (par exemple, les coûts économiques ou environnementaux associés à l'utilisation de l'énergie)
- *les avertissements* graphiques ou sonores (comme sur le paquet des cigarettes)
- *les stratégies de pré-engagement* (par lesquelles les gens s'engagent à suivre une certaine ligne de conduite d'action)
- *les rappels* (par exemple : par courrier électronique comme pour les rendez-vous à venir)
- *susciter des intentions de mise en œuvre* ("Avez-vous l'intention de voter?")
- *informer les gens de la nature et des conséquences de leurs propres choix passés* (par exemple : le "smart disclosure" aux États-Unis et le "midata project" au Royaume-Uni)

Nous nous sommes ainsi inspirés de cette littérature pour adapter des nudges considérés comme efficaces à deux protocoles d'expériences par la suite (partie II). Pour ce faire, nous avons notamment déployé une plateforme en ligne pour étudier les nudges dans le dialogue (cf section 7.2). Si des projets similaires existent pour étudier en ligne des jeux économiques de théorie des jeux par exemple (comme VeconLab²), il s'agit du premier projet de ce type pour les nudges a priori.

Limites des nudges

Si les nudges montrent des succès étonnants, différentes raisons expliquant leurs échecs dans certaines situations ont été proposées. Sunstein y consacre notamment un article [Sunstein, 2016], investiguant en particulier l'inefficacité des nudges par défaut (default rules). Selon lui, les deux raisons principales sont l'existence préalable de fortes préférences individuelles et la présence de "contre-nudges" déjouant les efforts du nudge considéré, notamment lorsque

²<http://veconlab.econ.virginia.edu/>

d'autres acteurs économiques sont impliqués. Il donne ainsi l'exemple de banques contrecarrant une règle par défaut, destinée à protéger les consommateurs contre les frais élevés de découvert bancaire, à l'aide de marketing actif, d'incitations économiques, et de nudges de simplification ou de decoy effect, exploitant la croyance populaire qu'il est coûteux de ne pas être enrôlé dans leur programme.

Bien qu'ils estiment le succès des nudges étudiés dans leur analyse à 66%, [Caraban et al., 2019] étudie également la raison de l'échec de certains nudges rencontrés. Selon eux, l'efficacité ne dépend pas du mécanisme du nudge en lui-même mais davantage de l'implémentation particulière choisie dans un certain contexte. D'après les résultats de ces études [Sunstein, 2016][Caraban et al., 2019], les principales causes à l'origine du manque de réussite de certains nudges seraient finalement :

- *des effets non durables* : Avec l'accoutumance, certains nudges peuvent perdre de leurs effets. Le nudge peut également fonctionner par automatisme sans véritables effets éducationnels. De manière générale, les études sur le nudge investiguent peu les effets à long-terme de ceux-ci : 14% des articles analysés dans [Caraban et al., 2019]. Seulement 35% des nudges efficaces rencontrés lors de cette même étude, avaient des effets durant plus d'un jour.
- *de fortes préférences individuelles et habitudes établies* : Le passé d'un individu (préférences alimentaires, etc.) est un biais important qui devrait être pris en compte selon le domaine d'application du nudge. Néanmoins, Sunstein précise que le manque d'impact d'un nudge sur certaines personnes et selon certains contextes est la preuve que le nudge préserve une certaine forme de liberté individuelle [Sunstein, 2016].
- *un manque de compréhension de l'intention des designers du nudge* : Le nudge peut dans certains cas donné lieu à une interprétation non envisagée par ses designers. Le timing et le contexte sont généralement des facteurs importants à examiner pour éviter toute confusion ou influence extérieure sur le nudge.
- *de la réactance* : Bien que rares selon [Sunstein, 2016], certains nudges peuvent provoquer la réaction inverse de celle souhaitée, lorsque l'individu sent son autonomie ou sa liberté affectée négativement. L'autorité perçue comme l'origine du nudge peut en être un facteur. Certains nudges jouant sur une légère friction avec la personne cible (dans la catégorie "Confronter") peuvent aussi avoir ce genre d'effets [Caraban et al., 2019].

- *une réaction de compensation venant annuler les effets du nudge* : Par exemple, si le fait de pousser les gens à faire plus de sport les incite à manger davantage de nourriture peu saine en contrepartie.

L'efficacité d'un nudge est donc dépendante de sa mise en œuvre, ce qui motivera ainsi notre prise en compte de facteurs individuels dans nos expériences suivantes.

Hypothèses

Les nudges sont donc un outil relativement efficace pour inciter un comportement sans contraindre une personne et dont les applications sont très larges. Dans le cadre de cette thèse, nous nous concentrons sur des nudges *linguistiques* (c'est-à-dire reposant sur le langage) et utilisés *à l'oral par un agent en interaction avec un utilisateur*. Nous formulons alors plusieurs hypothèses de travail sur les nudges :

1. Nous supposons que le type d'interlocuteur (humain, robot ou enceinte) peut avoir une influence sur l'efficacité des nudges. Des travaux précédents ont en effet pu montrer différents biais envers les machines [Hidalgo, 2021]. Une étude de jeu coopératif de théorie des jeux (dilemme du prisonnier) [Ishowo-Oloko et al., 2019] indique néanmoins un compromis entre performance et transparence sur l'identité de l'interlocuteur lorsque celui-ci est une machine utilisant un algorithme (d'apprentissage par renforcement). Il s'agit cependant d'un contexte bien précis de négociation qui ne correspond pas à l'architecture de choix des nudges.
2. Nous faisons l'hypothèse qu'un système de dialogue utilisant des questions de nudges oraux pourrait changer significativement le choix d'un utilisateur pendant le dialogue comparativement à des questions sans nudges.
3. Nous supposons qu'une adaptation personnelle de stratégies de nudges avec RL est également plus efficace pour influencer une personne que des stratégies générales non individuelles.
4. Pour estimer l'état actuel du dialogue ainsi que l'état mental et émotionnel de la personne, nous choisissons d'extraire certaines informations émotionnelles, paralinguistiques et contextuelles que nous supposons pertinentes pour prédire les nudges. Nous nous inspirons pour cela des travaux précédents de la littérature et des résultats de notre première expérience.

Nous avons ainsi vu dans cette partie l'origine et les enjeux des nudges numériques. Nous pouvons notamment remarquer que certains nudges semblent être liés à un état émotionnel, ce qui motive notre étude des émotions et l'utilisation de modèles de détection associées par la suite.

Chapitre 3

Détection d'émotions et d'états mentaux

Une partie importante de nos processus de décision est dépendante de nos émotions et nous avons choisi de les prendre en compte tout particulièrement dans notre modèle. En effet, les émotions influent sur notre jugement [Angie et al., 2011] et sur nos raisonnements [Damasio, 1994]. Elles ont ainsi un lien étroit avec la prise de décision, l'attention et la mémoire, qui sont des paramètres clés des nudges.

3.1 Cadre théorique des émotions et états mentaux

3.1.1 Théorie des émotions

Bien que les émotions soient courantes au quotidien, leur fonctionnement et nature exacts sont en réalité difficiles à définir. Leurs mécanismes, notamment d'un point de vue cognitif, sont encore le lieu de nombreuses questions de recherche. Quelles sont leurs causes ? Qu'est-ce qui est inné dans les émotions ? Comment expliquer des réactions émotionnelles différentes d'individus dans des situations similaires ? De nombreuses théories dans des domaines variés ont cependant tenté de les expliquer. Nous présentons ici les principales théories avancées.

Théorie biologique

Ayant fasciné depuis l'Antiquité bien des philosophes (Aristote, Descartes, etc.), les émotions ont également été étudiées sous une perspective évolutive et

biologique avec [Darwin \[1872\]](#). Darwin distingue alors certaines émotions primaires qui résulteraient des adaptations successives de l'organisme à son environnement. Selon lui, les émotions seraient innées, acquises avec l'évolution et universelles. Pour une même situation, elles déclencheraient une série de réponses biologiques (hormones, modification du rythme cardiaque, etc.) sélectionnées par l'évolution, qui se retrouveraient exprimées d'une manière similaire indépendamment des cultures et des espèces. Cette première théorie scientifique inspira ensuite d'autres travaux comme la théorie des émotions de base de [Ekman \[1999\]](#) mentionnée par la suite.

Théories physiologiques

Si le lien entre émotions et réactions physiologiques (réactions liées à l'organisme humain) semble naturel, une question peut se poser sur l'ordre d'apparition de ces 2 phénomènes.

[Jammes \[1884\]](#) et [Lange \[1885\]](#) défendent alors que les modifications physiologiques du corps précèdent les émotions et que c'est la prise de conscience de ces changements qui crée l'émotion. Leur théorie est ainsi nommée "périphéraliste". Par exemple, une personne aurait peur parce qu'elle se rend compte qu'elle tremble.

Pour [Cannon \[1927\]](#) et [Bard \[1928\]](#) cependant, l'émotion est plutôt déterminée par le traitement d'un stimulus au niveau du système nerveux "central" (cerveau et moelle épinière), en particulier au niveau du thalamus. Leur théorie "centraliste" critique notamment l'approche de James-Lange en précisant que les modifications physiologiques sont parfois imperceptibles, non réellement spécifiques à une émotion particulière et seraient donc davantage des conséquences plutôt que les causes des émotions. [Schachter and Singer \[1962\]](#) propose alors une théorie "bi-factorielle", s'inspirant de ces 2 théories. Selon lui, l'émotion serait issue de l'interaction de 2 éléments : une activation physiologique diffuse d'une certaine intensité mais non identifiée et une interprétation cognitive déclenchée par cette activation qui permettrait d'obtenir la nature subjective de l'émotion. Par exemple, dans le cas où une personne rencontre un ours, celle-ci a des tremblements. Le cerveau sollicité par cette réaction analyse alors la situation pour en comprendre l'origine, remarque la présence de l'ours et interprète ainsi ces tremblements comme une réaction de peur.

Bien que légèrement contre-intuitive, la théorie de James-Lange a été revisitée à la fin du XXème siècle par [Damasio \[1994\]](#) notamment qui introduit la notion de "marqueurs somatiques" présentés comme des réactions physiologiques associées à des événements passés émotionnels. Ces marqueurs seraient activés lors de nouvelles situations influençant ainsi nos prises de déci-

sions.

Théories psychologiques et cognitives

Parmi les théories psychologiques et cognitives récentes, 3 grandes théories se distinguent en particulier.

La première est celle des **émotions de base**. Pour Ekman par exemple, les émotions sont discrètes et mesurables car comportant les mêmes réactions physiologiques universelles peu importe l'individu [Ekman, 1999]. Il identifie alors 6 émotions primaires : la peur, la colère, le dégoût, la joie, la surprise et la tristesse. Il distingue ces émotions d'autres émotions plus morales ou sociales. Elles ont notamment comme caractéristiques d'être innées, spontanées, de courte durée et de faire intervenir de manière systématique les mêmes expressions faciales. [Plutchik, 1984] a également proposé une manière de combiner différentes émotions primaires en les représentant sur une roue (cf Figure 3.1).

Une autre approche est celle des **théories dimensionnelles**. Pour [Russel and Feldman-Barrett \[1999\]](#), si les émotions sont bien issues de l'évolution, l'utilisation de catégories ne permet pas de bien les représenter. Ils proposent alors de combiner plusieurs axes ou dimensions élémentaires comme l'activation (passive-active) et la valence (positif-négatif) permettant ainsi de rendre compte de manière continue des nuances subjectives d'une émotion pour un individu (cf Figure 3.2). Cependant, deux dimensions ne semblent pas suffisantes pour différencier de manière précise les différentes émotions et d'autres dimensions ont ainsi été proposées comme la dominance [Russel, 1997]. Néanmoins, il ne semble pas exister de réel consensus sur le choix des dimensions élémentaires à sélectionner (en particulier au-delà de 2 dimensions).

La dernière approche, qui est la théorie majoritaire aujourd'hui, est celle de **l'évaluation cognitive**. Selon cette dernière, l'émotion ressentie dans une situation dépend de notre évaluation cognitive (*appraisal*) des stimuli perçus dans notre environnement et de l'interprétation personnelle que nous faisons de ceux-ci. Cette théorie s'est développée à partir des travaux d'Arnold en 1960 et de Lazarus en 1966. Cherchant à déterminer les facteurs déterminants de cette évaluation, Lazarus [1968] en identifie alors 3 composantes :

- l'évaluation primaire : la personne cherche à estimer les implications d'un stimulus donné sur son bien-être

- l'évaluation secondaire : la personne analyse ses propres capacités et ressources pour faire face à cette situation
- la réévaluation : réévaluation en continue des évaluations primaires et secondaires permettant une adaptation progressive

Scherer and Peper [2001] propose plus tard le modèle des “processus composants” pour expliquer l'origine d'une émotion. Il présente alors l'évaluation cognitive d'un événement comme une succession d'étapes très rapides traitant les stimuli détectés et appelées “SEC” (*Stimulus Evaluation Checks*). Il identifie ainsi différentes dimensions permettant de caractériser l'évaluation d'un événement de manière séquentielle :

1. la *pertinence* : la nouveauté (soudaineté, familiarité, prédictibilité), l'agrément intrinsèque, la pertinence par rapport aux buts et besoins de la personne
2. les *implications* : les causalité de l'agent et du motif, la probabilité des conséquences, les différences avec les attentes, l'opportunité et l'urgence
3. le *potentiel de maîtrise* : contrôle, puissance, ajustement
4. la *signification normative* : compatibilité avec les standards internes et externes

Scherer insiste sur le fait que l'émotion est multidimensionnelle (cognitive, physiologique, motrice avec des expressions faciales et autres postures, motivationnelle et subjective) et qu'il s'agit d'un processus dynamique, mis à jour continuellement.

Néanmoins, prendre en compte l'aspect dynamique d'une émotion est difficile à réaliser en pratique. En général, l'émotion est plutôt estimée pour une certaine temporalité (souvent des segments audio de quelques secondes) en utilisant soit une représentation discrète (ou catégorielle), soit continue (ou dimensionnelle).

3.1.2 Etats mentaux

De manière plus générale, l'état actuel d'un interlocuteur peut être représenté par un état mental, qui dépasse ainsi le cadre purement émotionnel. Le fondement d'une telle représentation s'appuie sur le concept neuropsychologique de la théorie de l'esprit (*Theory of mind*), qui désigne la capacité cognitive d'attribuer une ou plusieurs représentations mentales à soi et aux autres. Ces états

mentaux sont alors inférés à partir des expressions émotionnelles, comportementales, attitudes et connaissances supposées de la personne. Le terme a été introduit pour la première fois en 1978 dans l'étude : *Does the chimpanzee have a theory of mind ?* [Premack and Woodruff, 1978]. Cette théorie peut cependant être référencée de différentes manières dans la littérature et se rapproche notamment d'une forme d'empathie ou de compréhension sociale (*social understanding*)[Sullivan and Ruffman, 2004].

Les états mentaux sont généralement séparés en 2 types de représentation :

- une représentation cognitive, concernant plutôt l'état épistémique d'une personne (ses connaissances sur le monde)
- une représentation affective, permettant notamment d'interpréter l'intention d'une personne et englobant des émotions complexes et sociales

Dans cette thèse, nous nous sommes alors intéressés aux indices pertinents pour représenter l'état mental d'une personne susceptible d'être nudgée, ce qui a motivé la prise en compte d'informations contextuelles individuelles ainsi qu'une variété d'états émotionnels.

3.2 Représentations des émotions

3.2.1 Représentation discrète

L'approche la plus répandue est de représenter les émotions de manière discrète en sélectionnant un nombre minimum d'émotions primaires. Le nombre et la nature de ces émotions varient dans la littérature, nous citons ici quelques exemples :

- 5 émotions pour Oatley (1989) : bonheur, colère, dégoût, inquiétude, tristesse
- 6 émotions pour Ekman (1982) : colère, dégoût, joie, peur, surprise et la tristesse,
- 8 émotions pour Plutchik (1980) : colère, joie, dégoût, tristesse, surprise, peur, anticipation, confiance
- 11 émotions pour Arnold (1960) : amour, aversion, colère, courage, découragement, désespoir, désir, espoir, haine, peur, tristesse

Pour choisir des catégories émotionnelles, nous nous sommes principalement inspirés de la roue des émotions de [Plutchik, 1984] (cf Figure 3.1) représentant sur un cercle les relations entre les différentes catégories d'émotions, leurs intensités et leurs combinaisons. Nous avons cependant été amenés à simplifier nos catégories ensuite, pour optimiser les performances de nos modèles.

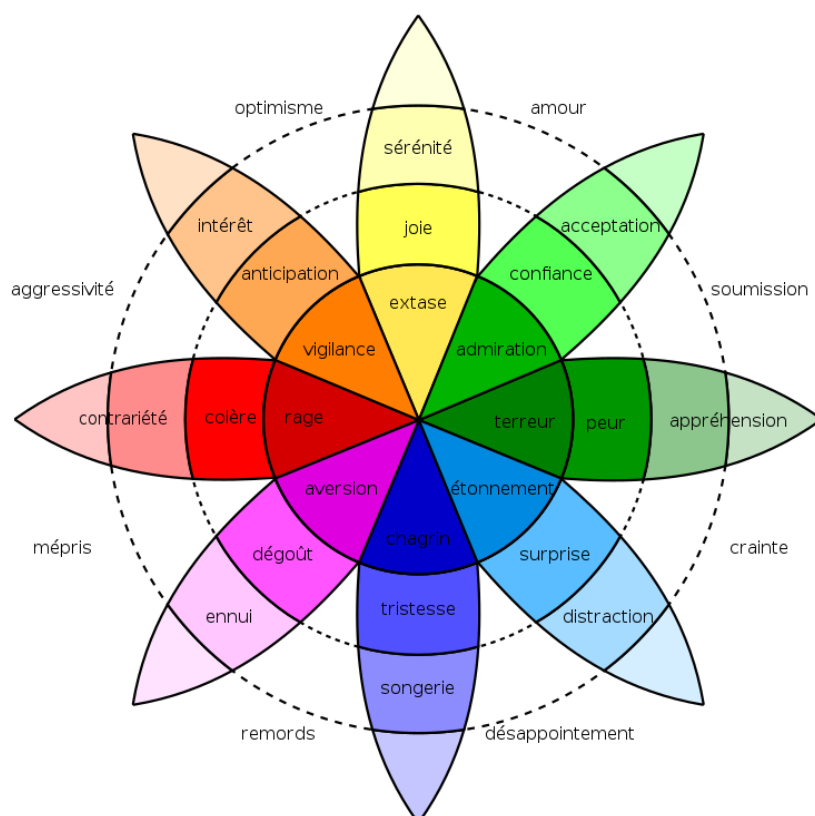


FIGURE 3.1 : Roue des émotions de Plutchik

Si les théories sur les émotions primaires sont une première référence pour choisir une représentation, de nombreuses catégories peuvent être utilisées pour caractériser la dimension affective orale en fonction de la temporalité sélectionnée (durée courte ou longue) et des aspects multimodaux considérés.

Scherer [1994] introduit par exemple la notion d'affect bursts. Ils sont définis comme : "des expressions affectives non-verbales, très brèves, discrètes, présentes dans la voix et le visage, déclenchées par des événements très clairement identifiables". Ils comprennent notamment les rires, souffles, hésitations

et raclements de gorge. Ils représentent donc des marqueurs réalistes, pouvant dénoter d'émotions fortes et constituent ainsi des indices importants pour estimer l'état mental actuel d'une personne.

L'approche catégorielle a l'avantage de présenter chaque émotion comme un état unique, bien différentiable ce qui est nécessaire pour pouvoir effectuer une tâche de classification ensuite. Dans ce cas-là, une catégorie "neutre" est généralement ajoutée, permettant de distinguer les événements affectifs remarquables du reste de l'interaction, souvent peu marquée émotionnellement.

3.2.2 Représentation continue

Un des intérêts de l'approche dimensionnelle est de pouvoir mesurer l'expérience subjective émotionnelle d'un individu. Son aspect continu offre notamment plus de nuances et la possibilité d'appliquer des modèles de régression.

La figure 3.2 illustre la représentation du circomplexe [Russel and Feldman-Barrett \[1999\]](#) représentant 2 dimensions souvent utilisées : la valence (désagréable-agréable) et l'activation (faible-forte).

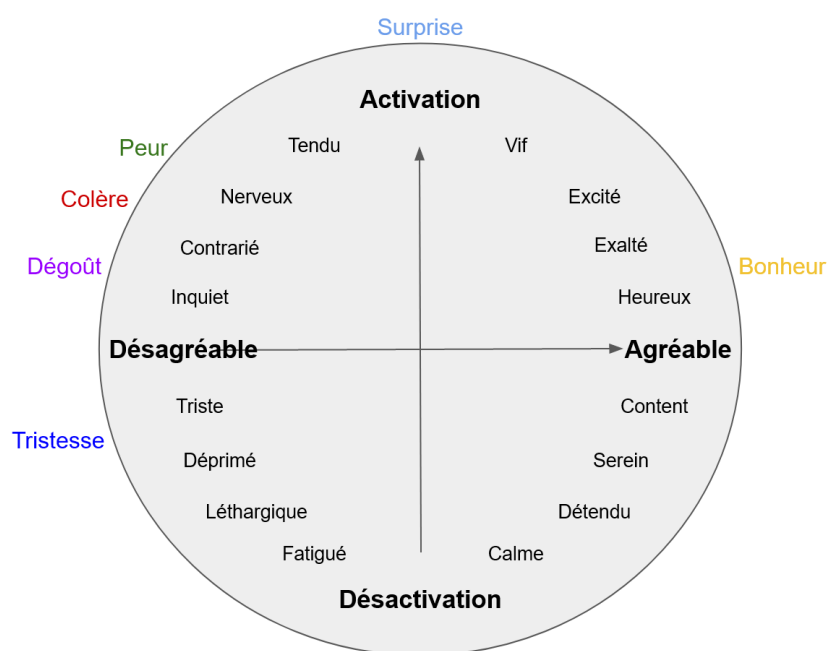


FIGURE 3.2 : Représentation d'un circomplexe bi-dimensionnel, avec les dimensions de valence (horizontale) et d'activation (verticale)

Une critique de ce modèle cependant est que ces deux dimensions seules

ne permettent pas réellement de différencier certaines catégories d'émotions. Par exemple, la colère et la peur sont toutes deux marquées par une activation forte et une forte valence négative dans cette représentation et peuvent être ainsi difficile à identifier. Des recherches ont notamment montré que des émotions de même valence (comme la colère et la peur) influençaient de manière différente le jugement et la prise de décision [Angie et al., 2011]. Néanmoins, nos corpus utilisés ne comportant pas de labellisation de peur ou de dégoût, ces deux axes fourniront une première estimation suffisante dans nos analyses.

3.3 Détection des émotions

3.3.1 Différence entre les données actées et naturelles

Lors du recueil d'un corpus oral comportant des émotions, deux types d'interactions peuvent avoir lieu. Les données collectées peuvent être actées (de manière scriptée ou non) ou naturelles (c'est-à-dire générées spontanément lors de l'interaction par l'individu). Les émotions actées sont plus faciles à obtenir mais ne sont pas toujours représentatives d'une émotion naturelle. Elles ont également l'avantage de permettre une comparaison de paramètres acoustiques plus fine lorsque l'énoncé linguistique est le même. Les émotions spontanées, quant à elles, peuvent demander davantage de traitement, en particulier en termes d'annotations (cf section suivante). Leur distribution dans le corpus est aussi très souvent déséquilibrée, avec généralement une surreprésentation d'émotions neutres.

Différentes études ont pu montrer la difficulté de généraliser les résultats obtenus sur des données actées à des données naturelles. Batliner et al. [2003] ont notamment comparé des expériences réalisées sur des données actées et réelles (en interaction Homme-machine) et ont conclu que leurs modèles performants entraînés sur des données actées obtenaient de mauvais résultats sur des données réelles. Vogt and Andre [2005] ont confirmé ces résultats en montrant que les paramètres pertinents pour ces deux types de données seraient différents et que les modèles entraînés sur des données actées auraient de meilleures performances. Nous devons donc tenir compte de cette difficulté dans le choix de nos corpus d'apprentissage pour la détection d'émotions, en particulier pour de l'apprentissage avec peu de données.

3.3.2 Annotation

Pour exploiter les données collectées lors de cette thèse, un certain nombre d'annotations en particulier sur les émotions ont été réalisées. Ces annotations

sont notamment nécessaires pour pouvoir faire de l'apprentissage supervisé sur des données spontanées.

Unités de dialogue

Bien qu'une annotation en continue soit possible, les annotations d'émotions dans un dialogue se font le plus souvent de manière segmentale. Le premier enjeu lors de l'annotation de données naturelles est alors le choix d'une unité de dialogue [Batliner et al., 1998]. La littérature emploie dans ce cas différentes tailles de segments :

- un mot : la plus petite unité, la plus flexible et donc la meilleure selon Batliner et al. [2003].
- un chunk : unité intermédiaire situationnelle de quelques mots moins longue qu'un tour de parole
- un groupe de souffle : unité entre 2 souffles
- une durée arbitraire courte et fixe pour l'ensemble des émotions
- un tour de parole : la parole d'un locuteur sans interruption dans le dialogue (entre 2 prises de parole d'un autre locuteur)
- un acte de dialogue [Batliner et al., 2003]

La durée des segments est généralement courte pour éviter un trop gros mélange d'émotions. Une question peut néanmoins se poser sur la taille minimum d'un segment pour permettre une bonne reconnaissance des émotions. Une étude réalisée sur le corpus JEMO (utilisé ici section 9) et CINEMO montre notamment qu'une durée d'au moins 1s est nécessaire pour une reconnaissance satisfaisante [Schuller et al., 2010].

Etiquetage et mélange d'émotions

Une fois l'intervalle d'annotation décidé, il convient de choisir les étiquettes discrètes ou continues à attribuer. L'annotation émotionnelle est alors séparée par locuteur et peut se faire sur plusieurs axes d'annotations, comportant dans certains cas des sous-étiquettes. Par exemple, des émotions primaires peuvent être regroupées dans des macro-classes négatif/neutre/positif, en particulier lorsque ces émotions primaires sont peu représentées.

Un point important à souligner est qu'une émotion réelle est rarement "pure" et qu'il s'agit plutôt d'un mélange d'émotions généralement Devillers et al. [2005].

Par exemple, une personne peut être à la fois en colère et amusée. [Plutchik, 1984] utilise ainsi un modèle de combinaison d'émotions et décrit ces émotions mélangées comme des "états mixtes" de 2 ou 3 émotions primaires. Selon Ekman [1999] et lui, ces émotions mélangées sont cependant bien discernables à l'intérieur du mélange. Pour prendre en compte cet aspect, un couple d'annotations émotionnelles majeure (émotion principale) et mineure (émotion secondaire) pour le segment considéré est généralement proposé Devillers et al. [2005].

Accord Inter-annotateur

Lors de l'acquisition de données spontanées, des annotations perceptives avec plusieurs annotateurs sont généralement réalisées pour prendre en compte l'aspect subjectif des émotions détectées. Différentes mesures sont alors utilisées pour estimer l'accord de ses annotateurs entre eux et obtenir ainsi une approximation de la difficulté à détecter les émotions du corpus en question. Une métrique généralement employée est le coefficient κ de Cohen [Cohen, 1960] (pour des variables catégorielles). Il est défini dans la formule suivante où P_a est la proportion d'accord obtenue et P_r la proportion d'accord pour une annotation aléatoire :

$$\kappa = \frac{P_a - P_r}{1 - P_r}$$

Le κ peut avoir une valeur maximale de 1 dans le cas d'un accord parfait entre 2 annotateurs et vaut 0 si l'accord n'est pas meilleur que le hasard. L'accord est considéré fort pour une valeur supérieure à 0.6 et presque parfait lorsqu'il est plus élevé que 0.8. Pour plus de 2 annotateurs, cette formule peut être généralisée avec le κ de Fleiss.

D'autres métriques existent également pour de multiples annotateurs comme le coefficient α de Krippendorff [Krippendorff, 2011] qui calcule plutôt le désaccord entre annotateurs, une mesure potentiellement plus pertinente. Il offre notamment les avantages d'être applicable à des données variées (nominales, ordinales, intervalles, etc.) et de différentes tailles, ainsi que d'ignorer les données manquantes. Il s'exprime sous une forme similaire au κ donnée ci-dessous, où D_o est le désaccord mesuré et D_e le désaccord attendu pour une annotation aléatoire :

$$\alpha = 1 - \frac{D_o}{D_e}$$

Comme pour le kappa, le coefficient α de Krippendorff prend une valeur maximale de 1 pour un accord parfait et 0 si l'accord n'est pas meilleur que le hasard. Krippendorff indique qu'une valeur supérieure à 0.8 est recommandée. Si le coefficient α de Krippendorff est plus flexible et général que le κ de Fleiss, dans la pratique, ces deux mesures donnent en général des scores similaires [Gwet, 2016].

3.3.3 Représentation des paramètres

Pour entraîner un modèle de *machine learning*, il convient alors de choisir la paramétrisation des segments audio qui serviront ensuite d'entrée aux modèles.

Les caractéristiques recherchées pour un paramètre, pour une classification notamment, incluent :

- *le caractère discriminant* : Le but des descripteurs est en particulier de permettre de maximiser les différences interclasses et minimiser celles intra-classes.
- *la robustesse* : Robustesse au bruit, aux variations d'interlocuteur et d'échelles. Un des objectifs d'un ensemble de descripteurs est ainsi de capturer l'ensemble des contraintes pertinentes à l'apprentissage. Ces contraintes sont souvent liées au type de corpus utilisé ensuite. Par exemple, pour la reconnaissance d'émotions, il peut être intéressant de considérer le type de voix (enfant-personnes âgées, homme/femme), les bruits d'environnement liés au support d'enregistrement, etc. En général, utiliser des corpus avec des domaines variés peut aider à limiter les biais de représentation. Certaines de ces variations d'échelles peuvent notamment être éliminées par des traitements de normalisation dans un second temps.
- *le caractère compact* : Le but est de réduire si possible l'espace de représentation des caractéristiques. En effet, un trop grand nombre de caractéristiques, en particulier lorsque la base de données est petite, peut nuire à la généralisation. Cela peut créer un surapprentissage dû à la trop grande différence entre la complexité du modèle et la taille des données. Dans certains cas, des algorithmes tels que la *Principal Component Analysis* (Analyse en Composantes Principales) (PCA) ou des méthodes de clusterisation peuvent alors aider à réduire le nombre de paramètres en préservant un maximum d'informations.

Plusieurs approches pour préparer l'audio en entrée d'un modèle sont alors possibles : une paramétrisation à l'aide de descripteurs acoustiques, l'utilisation

de spectrogrammes ou encore conserver le signal audio brut. Contrairement à la première approche, la plus ancienne, les modèles récents *end-to-end* (du type CNN et bi-LSTM, cf ci-dessous) et de *Deep learning* principalement utilisés pour les deux autres approches n'utilisent alors plus de paramètres explicites, ce qui les rend plus opaques.

Descripteurs acoustiques

Une première méthode est d'extraire de l'audio des mesures et statistiques numériques relatives au signal. Un des avantages de cette paramétrisation par rapport à d'autres est qu'elle peut permettre de traiter facilement des segments audio de différentes longueurs (le nombre de paramètres statistiques ne changeant pas avec la longueur). Elle nécessite néanmoins de sélectionner le set de paramètres les plus efficaces adaptés au corpus, à l'aide d'un algorithme sur un sous-corpus de développement par exemple [Rong et al., 2009]. Selon Tahon [2012], la sélection des meilleurs paramètres acoustiques pour analyser une voix émotionnelle doit notamment prendre en compte la variabilité de l'environnement, de l'interaction, du locuteur (enfants, adultes ou personnes âgées) et de ses caractéristiques (physiques, émotionnelles).

Les descripteurs acoustiques employés s'inspirent généralement de domaines comme la phonétique et la reconnaissance de la parole ou du chant. Scherer a par exemple cherché à relier des aspects de phonétique articulatoire aux émotions à travers une modélisation de 3 types majeurs de voix [Scherer, 1986].

Les descripteurs acoustiques contiennent ainsi des indices :

- *sur le signal audio* : énergie (amplitude), durées
- *fréquentiels* : Il s'agit de paramètres relatifs à la fréquence fondamentale et aux premiers harmoniques. Les *jitters* et *shimmers* sont également des indices pertinents correspondant à des micro-variations de la fréquence fondamentale et de l'énergie suivant les périodes de fermeture de la glotte.
- *spectraux* : Le spectrogramme montre la répartition de l'énergie en fonction des fréquences. Il permet ainsi de représenter graphiquement les caractéristiques vocales. Les indices utilisés comprennent l'enveloppe spectrale, le spectre à court terme (FFT), les ondelettes et les coefficients LPC : permettant d'obtenir des formants (maxima de fréquence) et des informations relatives au conduit vocal [Ruiz et al., 2010].
- *cepstraux* : Le cepstre est une représentation du spectre utilisant la *Fast Fourier Transform* (Transformée de Fourier rapide) (FFT) inverse. Les mesures cepstrales utilisées incluent les coefficients MFCC, robustes au bruit

de fond et montrant les changements ou la périodicité dans le spectre [Bogert et al., 1963] et LPCC (coefficients dérivés des LPC).

Dans le cadre de cette thèse, cette approche a été sélectionnée pour traiter les signaux audio utilisés par nos modèles de reconnaissance émotionnelle. Plus précisément, nous avons extrait des fichiers audio des paramètres acoustiques experts grâce au logiciel open-source *OPENSIMILE* [Eyben and Schuller, 2015]. En effet, ce logiciel permet d'extraire des ensembles standardisés de descripteurs acoustiques, comme l'ensemble minimaliste de Genève *eGeMAPs* [Eyben et al., 2016].

Spectrogrammes

Une autre approche est d'appliquer des techniques de machine learning performantes sur les images comme les *Convolutional Neural Network* (Réseau Neuronal Convolutif) (CNN) directement sur les spectrogrammes associés à l'audio [Satt et al., 2017][Etienne, 2019]. La tâche est ainsi transformée en un problème d'apprentissage sur images (en utilisant l'aspect graphique des spectrogrammes) sur lequel des modèles performants existent déjà [Tan and Le, 2019][He et al., 2015]. Pour conserver l'aspect séquentiel du signal audio, un bi-LSTM (*Long Short Term Memory*) est alors ajouté à la suite du CNN avec un apprentissage de bout en bout (*end-to-end*) [Satt et al., 2017][Etienne, 2019].

L'un des avantages d'utiliser cette paramétrisation est qu'il est possible d'éliminer les bruits non vocaux du spectrogramme (à l'aide d'une modélisation harmonique) [Satt et al., 2017]. Une contrainte cependant est le besoin de conserver des segments audio de taille identique (avec un *zero-padding* par exemple : en complétant les segments plus courts avec des zéros).

Signal Brut

Des modèles neuronaux peuvent également être entraînés directement sur le signal brut, utilisant par exemple une architecture similaire composée de CNN et bi-LSTM [Trigeorgis et al., 2016].

Le modèle *Wave2Vec2* [Baevski et al., 2020] propose également une représentation de la parole apprise de manière auto-supervisée à partir du signal brut et mettant à profit le succès des mécanismes attentionnels des *transformers* [Vaswani et al., 2017]. Il s'agit néanmoins d'un gros modèle neuronal dont la version large possède plus de 300 millions de paramètres. D'autres modèles plus légers ont ainsi été proposés comme *Audio ALBERT* [Chi et al., 2020]. L'avantage de ces modèles de représentation neuronaux attentionnels est qu'ils

peuvent être préentraînés en amont apprenant une représentation générale de la parole, puis être entraînés rapidement et plus finement (*fine-tuning*) sur une tâche plus précise comme la classification émotionnelle dans une approche *end-to-end*.

3.3.4 Algorithmes classiques pour la détection d'émotions

La détection d'émotions, comme toute tâche de machine learning peut être non supervisée (les données ne sont pas annotées), partiellement ou complètement supervisée (les données sont annotées : la valeur cible à prédire est donc connue) ou être apprise par renforcement. Parmi ces 3 grandes familles d'apprentissage, nous nous concentrons ici sur des méthodes d'apprentissage supervisé, l'approche la plus classique dans ce domaine, et citons certains modèles de référence et les avantages qu'ils offrent.

Les algorithmes en machine learning sont généralement différenciés selon qu'ils sont :

- *génératifs* ou *discriminatifs* : Les algorithmes discriminatifs (les plus courants) associent à une variable d'entrée une valeur de sortie. En classification, leur but est d'apprendre les labels à partir des caractéristiques des exemples. Les algorithmes génératifs, quant à eux, cherchent plutôt à apprendre une représentation des caractéristiques des données. Pour cela, ils apprennent à générer de nouvelles données similaires aux données initiales qui modélisent à travers une nouvelle représentation les caractéristiques essentielles de chaque classe. L'avantage des méthodes génératives est qu'elles offrent alors une estimation de la confiance de prédiction du modèle.
- *paramétriques* ou *non-paramétriques* : Les algorithmes paramétriques utilisent un nombre fixe de paramètres (indépendants du nombre d'exemples) contrairement aux approches non-paramétriques pour lesquelles la complexité du modèle est souvent croissante avec le nombre d'exemples. Les algorithmes non-paramétriques gardent généralement en mémoire de manière explicite les exemples vus et sont donc plus complexes computationnellement a priori. Ils ont néanmoins l'avantage de poser très peu de conditions sur les données.

Les méthodes d'apprentissage classiques utilisés pour la reconnaissance d'émotion incluent les algorithmes donnés ci-dessous.

SVM

Les *Support Vector Machine* (Machine à Vecteurs de Support) (SVM) sont des algorithmes discriminatifs, introduits par Cortes and Vapnik [1995], permettant de faire de l'apprentissage non-supervisé (*Support Vector Clustering*) et supervisé comme de la régression et de la classification. En termes de classification, ils consistent à trouver un hyperplan qui permet de séparer linéairement deux classes. Vapnik a alors montré que lorsqu'une séparation est possible, une solution performante est celle qui maximise la **marge**, c'est-à-dire la distance entre l'hyperplan et les exemples les plus proches de celui-ci pour chaque classe, ceux-ci étant appelés vecteurs supports. Cette maximisation de marge offre notamment des résultats plus robustes en minimisant le risque de surapprentissage.

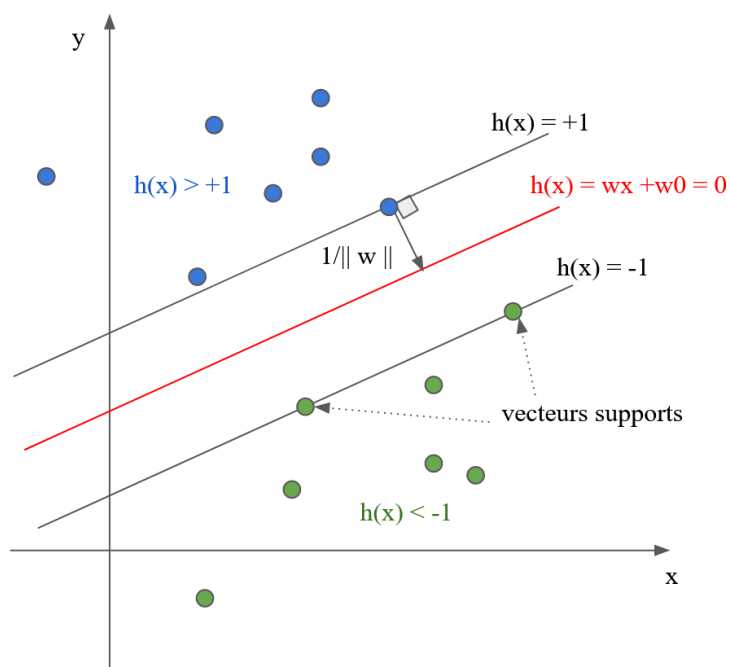


FIGURE 3.3 : Illustration d'un hyperplan h maximisant la marge dans un cas linéairement séparable en dimension 2. L'hyperplan est alors perpendiculaire au segment de droite le plus court séparant un vecteur support de l'hyperplan h et dont la longueur vaut $\frac{1}{\|w\|}$ (en normalisant w et w_0)

La résolution d'un SVM est alors un problème d'optimisation qui sous sa forme primale constitue à minimiser l'inverse de la marge d'un hyperplan h (dans \mathbb{R}^d) séparant correctement tous les exemples x :

$$\begin{cases} \text{Minimiser } \frac{1}{2} \|w\|^2 \\ \text{avec } \text{label}(x) * h(x) \geq 1, \forall x \in \mathbf{R}^d \quad \text{où } h(x) = wx^T + w_0 \end{cases}$$

En pratique, il est résolu sous une forme duale à l'aide d'un lagrangien ne prenant en compte que les vecteurs supports. L'hyperplan solution ne requiert ainsi que le calcul de produits scalaires.

Pour un nombre de classes supérieur à 2, des méthodes classiques de généralisation de classifieurs binaires sont possibles comme le *one-versus-one* (construire un modèle binaire pour chaque couple de classe possible et procéder par vote) ou le *one-versus-all* (construire un modèle binaire par classe et procéder par vote).

Un des points forts des SVM est qu'ils peuvent effectuer une classification non-linéaire en représentant les données dans des espaces de hautes dimensions (voire infinis) où un classifieur linéaire peut alors être utilisé. Pour éviter des calculs lourds en haute dimension, une astuce appelée **kernel trick** est alors employée pour calculer efficacement les produits scalaires grâce à une fonction noyau. Celle-ci représente dans un sens large une mesure de similarité entre les éléments de l'espace initial, suivant le principe intuitif que deux données similaires a priori devraient être classifiées de la même manière.

La fonction noyau employée par la suite (cf section 9) est celle du noyau gaussien, ou noyau de fonction à base radiale (RBF) (qui correspond à un espace de redescription de dimension infinie) :

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{\sigma^2}\right)$$

où σ est un hyperparamètre contrôlant la bande passante de la gaussienne, plus il est faible, plus la gaussienne est étroite (x et y devront être très proches pour que le noyau soit non nul).

Il n'est néanmoins pas toujours possible de trouver une séparation linéaire dans le nouvel espace utilisé. [Cortes and Vapnik \[1995\]](#) ont alors proposé l'idée de marge souple permettant de tolérer certains mauvais classements. Un terme de pénalité C (choisi en hyperparamètre) est alors ajouté servant à contrôler le compromis entre la taille de la marge et le nombre d'erreurs. Plus cette constante C est grande, plus la marge contenant les erreurs sera petite.

L'avantage des modèles SVM est qu'ils sont relativement robustes pour de petites bases de données et pour **un grand nombre de caractéristiques** mais ils ne sont pas adaptés à de grosses bases de données. Pour ces raisons, ils représentent un algorithme idéal pour réaliser une détection d'émotions sur

des petits corpus [Tahon, 2012]. Dans notre cas en particulier, cet algorithme est pertinent pour un apprentissage avec peu de données collectées et pour un grand nombre de descripteurs acoustiques, comme les sets de paramètres experts proposés par le logiciel OpenSmile [Eyben and Schuller, 2015].

HMM

Les *Hidden Markov Model* (Modèle de Markov Caché) (HMM) [Baum and Petrie, 1966] sont un exemple de modèle génératif statistique dans lequel la tâche étudiée est assimilée à un **processus markovien** (une séquence d'états où la probabilité d'arriver dans un état ne dépend que de l'état précédent) composés d'états cachés.

Le but des HMM est ainsi généralement de retrouver une séquence de données non immédiatement observables. En classification, l'algorithme le plus utilisé est celui de Baum-Welch, un cas particulier de l'algorithme EM (Espérance-Maximisation).

Schuller et al. [2003] ont notamment utilisé des modèles HMM continus pour la reconnaissance d'émotions orales. De manière similaire aux mécanismes de processus de la parole, ils utilisent alors des modèles *Left-Right* (un graphe où les états transitent uniquement de gauche à droite) pour modéliser l'avancement temporel, avec un modèle pour chaque émotion. Ces modèles prennent alors en entrée des caractéristiques bas-niveau instantanées (relatives au pitch et à l'énergie détectés sur une courte fenêtre). Le modèle maximisant le maximum de vraisemblance est alors choisi comme l'émotion prédite. Les auteurs montrent alors que la performance de reconnaissance des modèles HMM augmente avec le nombre d'états cachés.

L'avantage des HMM est notamment de pouvoir s'appliquer à de longues séries temporelles et de pouvoir prendre ainsi en compte des variations courtes de comportement humain. Le choix a priori de la typologie du modèle (nombre d'états, règles de transitions) peut cependant rendre difficile l'ajout de connaissances acquises. Bien que le nombre d'observations nécessaires pour une bonne estimation augmentent en fonction de la taille du modèle, il s'agit néanmoins d'un algorithme clair et efficace qui peut être combiné avec d'autres approches pour former un modèle hybride [Shahin et al., 2019].

kNN

Le kNN est parmi les algorithmes les plus simples de machine learning. Il est non-paramétrique, discriminatif et s'utilise en régression comme en classification. En classification, il correspond à classer la nouvelle entrée dans la même

catégorie que ses k plus proches voisins dans l'espace des caractéristiques utilisés. La classification est généralement pondérée pour prendre en compte la distance par rapport aux voisins. Cela permet notamment d'éviter certains biais lorsque les classes ne sont pas équilibrées.

Cet algorithme a par exemple été utilisé pour de la reconnaissance émotionnelle à partir de signaux électriques EEG (électroencéphalogramme) du cerveau [Mehmood and H.J., 2015].

Réseaux de neurones

Les réseaux de neurones artificiels sont des méthodes paramétriques qui regroupent un ensemble de structures fréquemment employées en machine learning et reconnaissance émotionnelle [Etienne, 2019][Satt et al., 2017][Trigeorgis et al., 2016]. Ils sont généralement caractérisés par un enchaînement de couches de neurones prenant leurs entrées dans les sorties des couches précédentes, ce qu'on appelle un modèle *feed-forward* (même si d'autres modèles proposant une boucle de retour vers les couches précédentes sont possibles).

Bien qu'il existe de nombreuses structures de réseaux de neurones, les principaux composants à choisir pour un réseau de neurones sont les suivants :

- *Structure des couches de neurones* : De manière classique, un réseau *feed-forward* dispose d'une couche d'entrée, d'une couche de sortie et d'un certain nombre de couches intermédiaires, appelées couches cachées, ces dernières permettent l'apprentissage de fonctions non-linéaires. Cybenko [1989] a ainsi montré que toute fonction continue définie sur \mathbb{R}^d peut être approximée par un réseau à une seule couche cachée (avec un nombre de neurones suffisants). Le *deep learning* consiste alors à utiliser un nombre important de couches cachées (au moins 3), permettant par exemple d'extraire des patterns visuels dans chaque couche dans le cas des CNN. Si certains principes existent pour choisir le nombre idéal de couches et de neurones par couches [Heaton, 2008], cela demeure néanmoins un domaine de recherche ouvert. Dans nos modèles, ces variables correspondront à des hyperparamètres suggérés dans la littérature et déterminés durant la phase de validation, à l'instar de la grande majorité des modèles actuels.

La propagation des données dans chaque couche fait également intervenir une fonction de combinaison calculant la sortie d'un neurone en fonction de ses entrées (combinaison linéaire dans nos modèles) puis une fonction d'activation, ou fonction seuil, permettant d'introduire une non-linéarité dans ce calcul. De nombreuses fonctions d'activation ont été pro-

posées en fonction du type de critère de décision recherché (pente plus ou moins forte aux alentours de zéro) et pour résoudre les problèmes de gradient durant la rétropropagation (gradient évanescent) : sigmoïde, tangente hyperbolique, softmax, ReLU, LeakyReLU, etc. Pour une classification multi-classe, nous utiliserons alors une fonction softmax en dernière couche pour obtenir une sortie probabiliste, et la populaire fonction ReLU pour les couches intermédiaires car elle permet d'accélérer l'apprentissage (représentation clairsemée puisqu'elle nullifie les valeurs négatives).

- *Fonction de perte* : Apprendre un réseau de neurones peut être assimilé à un problème d'optimisation correspondant à minimiser l'erreur de prédiction. Plusieurs métriques existent alors selon la tâche considérée : classification binaire, multi-classe ou régression. Ces métriques correspondent à un coût à minimiser auquel peut être ajouté un terme de régularisation pour éviter le surapprentissage.
- *Optimizer* : Pour converger vers un minimum local, les poids des neurones sont mis à jour dans la direction opposée du gradient de l'erreur (rétropropagation du gradient) selon un pas défini par un hyperparamètre α , nommé taux d'apprentissage. Différents algorithmes d'optimisation peuvent alors être employés comme *Stochastic Gradient Descent* (Algorithme du gradient stochastique) (SGD), ou Adam [Kingma and Ba, 2015], choisi ici pour sa rapidité de convergence.

Si la reconnaissance émotionnelle à partir de modèles neuronaux repose en grande partie sur l'utilisation de CNN [Trigeorgis et al., 2016][Satt et al., 2017][Etienne, 2019], nous nous intéresserons dans cette thèse en particulier aux réseaux adaptés à un *few-shot* learning.

3.3.5 Apprentissage Few-shot

Les performances des modèles présentés ici sont très dépendantes du corpus utilisé et du nombre d'émotions à reconnaître choisi. Une bonne référence néanmoins est la performance humaine obtenue sur le corpus en question.

Dans le cadre de cette thèse, nous nous sommes intéressés tout particulièrement à la reconnaissance d'émotion dans un contexte d'apprentissage rapide *few-shot* (en quelques exemples). L'apprentissage *few-shot* signifie ici apprendre à reconnaître des émotions en n'entraînant le système que sur quelques exemples par émotion par utilisateur sur la base de données d'intérêt.

Méthodes

Plusieurs approches et modèles sont possibles. D'une part, un apprentissage par transfert permet de compenser le peu de données d'entraînement du corpus par une connaissance a priori sur d'autres données. Cela offre également une robustesse supplémentaire au système. D'autre part, une méthode qui peut être complémentaire est l'utilisation d'algorithmes apprenant une métrique cherchant à mesurer la similarité des classes entre elles plutôt que d'apprendre directement la distribution de ces classes comme la plupart des algorithmes de classification. L'idée est alors d'apprendre une représentation des caractéristiques (ou *embedding*) qui puisse ensuite être transférée à de nouvelles données d'apprentissage. L'apprentissage d'une telle métrique pour l'apprentissage *few-shot* a ainsi été appliqué à des réseaux de relation [Sung et al., 2018], des réseaux de neurones prototypes [Guibon et al., 2021] et des réseaux siamois [Feng and Chaspari, 2021]. Nous développerons notamment ce dernier modèle section 9.3.

Challenges de la tâche

L'approche de Feng and Chaspari [2021] a l'avantage de proposer un apprentissage rapide à partir de données audio preprocessées à l'aide du logiciel OpenSMILE [Eyben and Schuller, 2015] en profitant de données actées abondantes en anglais pour le préentraînement du réseau siamois. Les auteurs effectuent alors un transfert de connaissance de ces données sur d'autres domaines, en particulier sur des données spontanées, en copiant les poids du réseau siamois pour apprendre un réseau *Feed-forward* sur 10 personnes pour une classification sur 3 émotions. Ils évaluent alors différentes combinaisons de sets de données et différentes métriques pour l'apprentissage de la similarité relative des exemples, en introduisant notamment un échantillonnage de paires d'exemples adaptatives pour le préentraînement du réseau siamois. Cette approche nous a alors semblé pertinente pour notre étude cherchant à détecter des émotions spontanées dans l'audio mais dans le cas d'un dialogue. Guibon et al. [2021] s'intéresse également à l'apprentissage épisodique rapide dans le dialogue mais à partir de messages textuels cependant. Les auteurs montrent qu'un apprentissage *few-shot* est possible à l'aide de réseaux prototypes sur leur corpus mais qu'il demeure difficile d'atteindre des performances similaires à celles de méthodes supervisées. Ils montrent aussi les challenges d'utiliser des données collectées sur le terrain dans le cadre d'applications réelles.

Nous avons vu ici les principales approches pour la représentation et la re-

connaissance d'émotion dans la parole. Dans le cadre d'un dialogue cherchant à optimiser l'engagement de l'utilisateur, l'émotion est un facteur important, offrant un feedback, une estimation en temps réel de sa réaction et qui est ainsi souvent exploitée dans la conception d'un système de dialogue.

Chapitre 4

Structure classique d'un système de dialogue oral

Bien qu'il existe de nombreuses formes de systèmes de dialogue, nous nous intéressons ici surtout à un système dans lequel une machine (robot ou agent conversationnel) interagit oralement avec un humain dans le but d'accomplir une tâche précise (*goal-oriented*).

4.1 Historique et types de systèmes

Les systèmes de dialogue (ou agents conversationnels) sont un domaine de recherche très actif et ont connu de nombreux développements depuis les débuts de la recherche en intelligence artificielle. Les premiers systèmes fonctionnaient principalement à base de règles, comme le célèbre ELIZA (Weizenbaum, 1966) qui simulait un psychanalyste en transformant en questions les répliques précédentes. Selon l'enquête réalisée par [Dai et al. \[2021\]](#), ils constituaient la première des 3 générations de systèmes de dialogue.

D'après ces auteurs, les systèmes de dialogue ont ensuite évolué vers des modèles statistiques [[Young et al., 2010](#)][[Daubigney et al., 2012](#)][[Gasic and Young, 2013](#)] mettant à profit l'émergence des *big data* et décortiquant les étapes nécessaires à la compréhension et génération du dialogue en différents modules. L'utilisation de RL étaient alors fréquentes, profitant d'interactions en ligne avec des utilisateurs pour en faire ensuite une simulation ou bien entraîner le système de zéro [[Ferreira, 2015](#)]. Ces modèles statistiques nécessitent assez peu de connaissances expertes. Ils sont néanmoins difficiles à maintenir dans la durée et sur des très grosses données.

Récemment, les systèmes de dialogue tendent à utiliser du deep learning, remplaçant les différents modules par des réseaux de neurones profonds. Ces

systèmes reposent souvent sur des approches *end-to-end*, et/ou mettent à profit les récents avancements des modèles de langue reposant sur l'architecture *Transformers* [[Vaswani et al., 2017](#)].

Les systèmes de dialogue sont généralement répartis en 3 catégories [[Deriu et al., 2020](#)] :

- *Les systèmes orientés tâches* sont développés pour aider l'utilisateur à résoudre une tâche spécifique aussi efficacement que possible. Ils suivent une structure bien définie et sont en général dépendant d'un domaine. En principe, l'utilisateur comme le système peut prendre l'initiative durant le dialogue.
- *Les agents purement conversationnels* cherchent plutôt à entretenir des dialogues dans un domaine ouvert sans tâche spécifique à résoudre. Ils cherchent davantage à divertir et à reproduire un lien social. Ils sont généralement entraînés à l'aide de modèles prédictifs ou retrieval-based.
- *Les systèmes de réponse aux questions (QA)* sont construits pour la tâche spécifique de répondre aux questions. Ils ne sont pas définis par une structure mais suivent plutôt un modèle de questions-réponses. Ces systèmes cherchent souvent à extraire l'information sémantique contenu dans la requête de l'utilisateur, à travers certains mots clés par exemple, puis consultent ensuite leur base de données pour générer une réponse adéquate. Ces systèmes ont connu un récent succès avec des modèles comme GPT-3 de Open-AI et ChatGPT, mettant à profit les performances de très gros modèles *Transformers* (de l'ordre de centaines de milliards de paramètres) pour produire des réponses particulièrement réalistes.

Dans la suite, nous nous concentrerons principalement sur les systèmes de dialogue orientés tâches. En effet, pour mesurer concrètement l'influence des stratégies de nudges de notre système, nous chercherons à le borner sur une tâche bien précise.

4.2 Structure pipeline

Un dialogue est traditionnellement découpé en tours formant ainsi des cycles de dialogue. Un système de dialogue introduit en plus l'idée qu'un des interlocuteurs (au moins un) est une machine : robot, chatbot, etc. Pour un système de dialogue oral (en anglais SDS), le système doit alors prendre en entrée un

signal vocal de son interlocuteur et renvoyer une réponse orale. Généralement, l'utilisateur et le système sont considérés comme se répondant en alternance, mais il est également possible d'envisager une conception plus flexible du tour de parole. Par exemple, l'utilisateur pourrait être autorisé à couper la parole au système. Il pourrait également continuer de relever des feedbacks de l'utilisateur pendant qu'il formule une réponse. Dans notre cas, tous les tours de parole ont été considérés en alternance pour simplifier l'enregistrement de la réponse sur le serveur dont les contraintes sont notamment expliquées section 7.2.

Bien qu'il n'existe pas de représentation standard pour ce genre de système, il se décompose traditionnellement en une suite de différents modules, formant ainsi une pipeline (cf Figure 4.1) :

- la Reconnaissance automatique de la parole (ASR : *Automatic Speech Recognition*) et tout autre module de détections paralinguistiques
- la Compréhension du Langage Parlé (SLU : *Spoken Language Understanding*) qui fait appel généralement à une représentation sémantique
- la Gestion du Dialogue (DM : *Dialog Manager*) pour choisir l'action adaptée à la tâche demandée au système de dialogue
- la Génération en Langage Naturel (NLG : *Natural Language Generation*)
- la Synthèse Vocale (SS : *Speech Synthesis*)

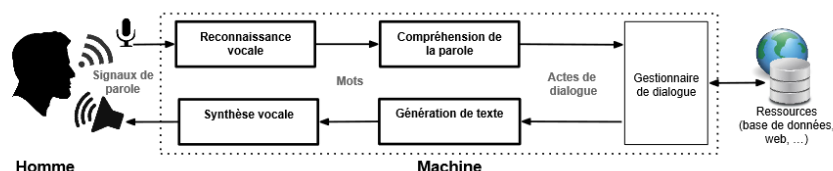


FIGURE 4.1 : Architecture d'un système de dialogue (extrait de [Ferreira, 2015])

4.2.1 Reconnaissance Automatique de la Parole : ASR

Le module d'*Automatic Speech Recognition* (Reconnaissance automatique de la Parole) (ASR) est le premier module du système. Son but est de transcrire la parole en texte (ou hypothèses de transcription) en des temps proches du temps réel. Parmi les modèles utilisés pour ce module, nous pouvons citer les HMM (*Hidden Markov Model*) et les réseaux de neurones profonds. Aujourd'hui, les

modèles de l'état de l'art utilisent notamment des modèles transformers comme *Wav2vec2* [Baevski et al., 2020] ou *Whisper* d'OpenAI [Radford et al., 2022] et ont des performances similaires à l'être humain.

L'évaluation de ce module se fait à l'aide de métriques comme le WER (Word Error Rate), qui mesure le taux d'erreur par mot. Cette évaluation est néanmoins à prendre dans le contexte d'un système de dialogue : paroles spontanées, bruit ambiant, etc. Selon la thèse de Ferreira (présentée en 2015) [Ferreira, 2015], de nombreux travaux de référence sur le domaine font ainsi en moyenne une faute tous les 5 mots. On sélectionne alors en sortie généralement la liste des N meilleures hypothèses pour la suite du traitement afin de prendre en compte les erreurs potentielles de l'ASR et limiter leur propagation le long de la pipeline des modules.

4.2.2 Compréhension de la Parole : SLU

Le module de *Spoken Language Understanding* (Compréhension du Langage Parlé) (SLU) prend généralement en entrée les hypothèses de transcription et cherchent alors à représenter sémantiquement l'information du tour de parole de l'utilisateur.

Actes de Dialogue

Le but est de récupérer l'intention du locuteur, souvent catégorisée par des actes de dialogue [Traum, 1999]. Traum précise alors que ces actes de dialogue ou *speech acts* relèvent davantage de la pragmatique que de la sémantique car leur théorie se concentre plus sur l'utilisation d'un message par des agents conversationnels plutôt que sur sa signification réelle. Un acte de dialogue est donc caractérisé par une action à laquelle peut être associée des arguments supplémentaires, par exemple : *Inform* ($a = b$) ou *Request* (a). Cette représentation est pertinente notamment pour des tâches d'apprentissage dites de *slot-value filling*, c'est-à-dire pour un système de dialogue orienté tâche où le domaine du dialogue est représenté par des variables (adresse, prix, etc.) à valeurs restreintes que le système a pour but d'estimer pour un utilisateur donné. Un exemple de tels domaines peut être trouvé dans le corpus *MultiWoz* [Budzianowski et al., 2018] rassemblant des dialogues sur des domaines comme la recherche de restaurants ou d'hôtels.

De manière générale, le module SLU cherche à extraire le domaine du dialogue (dans le cas d'un dialogue à multi-domaines), l'intention de l'utilisateur et la valeur des slots considérés pour le domaine.

Types de Modèles

La compréhension sémantique peut reposer soit sur des règles expertes (comme les premiers systèmes conversationnels, tel ELIZA, 1966), soit sur un apprentissage automatique, soit sur un mélange des deux, ce qui permet souvent de faciliter la tâche d'apprentissage et certains cas difficiles à interpréter. Les méthodes de machine learning (utilisées en grande majorité) sont plus performantes et facilement extensibles (scalable) mais nécessitent une grande quantité de données d'entraînement.

Il est utile ici de rappeler la distinction entre les méthodes génératives et discriminatives (cf section 3.3.4) afin d'expliquer l'intérêt des méthodes génératives dans le cas d'un modèle SLU.

Les méthodes **discriminatives**, qui représentent la majorité des techniques de machine learning, associent aux caractéristiques d'une entrée une étiquette ou valeur de sortie. Il peut s'agir de régression ou de classification. Plus formellement, ces méthodes modélisent directement $p(y|x)$ la probabilité d'obtenir un label y étant donné une entrée x . Un de leur avantage est qu'elles sont plus flexibles car elles ne nécessitent pas de supposer une indépendance des caractéristiques descriptives utilisées (contrairement aux méthodes génératives). Parmi ces méthodes, nous pouvons citer par exemple des algorithmes statistiques comme les *Conditional Random Fields* (Champ aléatoire conditionnel) (CRF) [Lafferty et al., 2001] dont l'application à la compréhension du langage parlé a été comparée positivement par rapport à 2 autres modèles de l'état de l'art (SVM et transducteur à état fini) dans Raymond and Riccardi [2007]. Les réseaux de neurones comme les CNN sont également un exemple classique d'utilisation d'algorithmes discriminatifs.

Les méthodes **génératives** apprennent plutôt à générer de nouvelles données similaires aux données initiales qui modélisent à travers une nouvelle représentation les caractéristiques essentielles de chaque classe. Formellement, ces méthodes modélisent d'abord, pour des données x et labels y , la loi de probabilité jointe $p(x,y)$, puis déduisent la probabilité conditionnelle $p(y|x)$ grâce à l'application de la règle de Bayes : $p(y|x) = \frac{p(x,y)}{p(x)}$

Elles ont l'avantage alors de permettre une estimation de la confiance de la prédiction du modèle, même si leur entraînement peut être plus coûteux. Cet aspect est particulièrement intéressant pour un module SLU car il offre plus de visibilité sur les performances du système et permet de classer les hypothèses de compréhension sémantique selon la confiance associée. En sélectionnant les N meilleures hypothèses, le Dialogue Manager peut alors prendre

en compte l'incertitude des états du dialogue. Parmi ces méthodes génératives, nous pouvons citer les HMM [Baum and Petrie, 1966] ainsi que les modèles de langues auto-régressifs récents, tels que BERT [Devlin et al., 2019] sur une entrée texte ou *Wave2vec2* [Baevski et al., 2020] sur une entrée audio, utilisant les mécanismes attentionnels des *transformers* [Vaswani et al., 2017], développés davantage dans le chapitre suivant.

Il est à noter que la structure pipeline avec une première transcription ASR, puis un module SLU tend à être remplacée aujourd'hui par des approches end-to-end fondée sur des architectures *sequence-to-sequence*, apprenant une représentation sémantique neuronale directement à partir de la parole.

Sans aller jusqu'à l'apprentissage complet en *end-to-end* du système, une autre possibilité est d'apprendre de manière jointe le module SLU avec un autre module, par exemple l'apprentissage du Dialogue Manager. Riou et al. [2018] ont notamment montré qu'un apprentissage joint du DM et d'un parseur sémantique en *zero-shot learning* était possible à l'aide d'apprentissage par renforcement.

4.2.3 Dialogue Manager : DM

Le Gestionnaire du Dialogue (DM : *Dialog Manager*) est la partie centrale du système de dialogue et cherche à sélectionner l'action adaptée (nouvelle réplique du système) en fonction de l'état du système. Il remplit donc deux fonctions :

- le suivi de l'état du dialogue, qui constitue un champ de recherche à part entière appelé *Dialog State Tracking* (Gestion du Dialogue) (DST). Le but est de mettre à jour efficacement à chaque tour de parole l'état estimé du système de dialogue à partir des entrées des modules précédents (ASR, SLU).
- le choix de l'action suivante, au niveau sémantique, pour le système de dialogue, que l'on nomme aussi *Dialog Policy* (Stratégie de Dialogue) (DP). Cette action est ensuite transformée en réponse orale à l'aide des modules NLG et SS.

Suivi du Dialogue : DST

Un dialogue se présentant comme une suite d'interactions, il est souvent modélisé comme une suite d'états, chaque état devant rendre compte de l'observation actuelle des modules d'entrée (ASR, SLU) et du contexte précédent de la conversation.

Il est alors possible de faire une représentation markovienne du dialogue à l'aide d'un *Markov Decision Process* (Processus de Décision Markovien) (MDP).

Un MDP [Bellman, 1957] permet de décrire les interactions d'un agent (un système de dialogue par exemple ici) avec son environnement et est défini par 4 composants :

- un ensemble d'états S , décrivant les différentes configurations dans lesquelles peut se trouver le système. Il peut être discret ou continu selon la modélisation choisie.
- un ensemble d'actions A , caractérisant les interactions du système avec son environnement, c'est-à-dire les différentes répliques que peut choisir le système pour échanger avec l'utilisateur. De même, il peut être discret ou continu même si dans le cas du dialogue, il est généralement discret.
- une fonction de transition $P : S \times A \times S \rightarrow [0;1]$, où $P(s,a,s')$ correspond à la probabilité de se retrouver dans l'état s' en partant de l'état s et en effectuant l'action a .
- une fonction de récompense $R : S \times A \times S \times \mathbf{R} \rightarrow [0;1]$, où $R(s,a,s',r)$ est la probabilité d'obtenir une récompense r en passant de l'état s à s' avec l'action a .

Les fonctions de transition et de récompense peuvent notamment être simplifiées dans les cas déterministes en $P : S \times A \rightarrow S$ et $R : S \times A \times S \rightarrow \mathbf{R}$.

Le MDP repose sur l'**hypothèse de markov** selon laquelle les transitions entre les états ne dépendent que de n états précédents. En général, on ne considère que le dernier état ($n=1$). La représentation de l'état du dialogue doit ainsi rendre compte de l'ensemble des paramètres importants à l'instant t pour choisir l'action suivante.

En ce qui concerne les dialogues *goal-oriented*, Young et al. [2010] ont suggéré qu'un *Partially Observable Markov Decision Process* (MDP Partiellement Observable) (POMDP)[Sondik, 1971], maintenant une distribution de probabilité sur tous les états du dialogue (appelée *belief state*), était le modèle le plus adapté car il permet de prendre en compte l'incertitude des modules (ASR, SLU) et de manière générale l'incertitude de l'état mental de l'interlocuteur qu'on essaie d'estimer. Un des enjeux est alors de représenter l'espace possible des états et des actions de manière compacte à travers des paradigmes comme *Hidden Information State* (Etat d'Informations cachées) (HIS) [Young

[et al., 2010](#)] où les états vont être regroupés en classes équivalentes (indistinguable au regard des observations). De manière générale, la complexité du POMDP peut être diminuée en décomposant les états du dialogue en trois composants : un pour le *but poursuivi* par l'utilisateur (ou plutôt dans notre cas, l'estimation de l'avancement de la tâche sur laquelle on souhaite le nudger), un pour ce qui est effectivement dit au *tour actuel* et un pour l'*historique du dialogue* gardant les informations pertinentes des tours précédents (afin de maintenir l'hypothèse de markov notamment).

Dans cette thèse, nous avons repris ce paradigme qui est adapté à un apprentissage par renforcement. Néanmoins, les algorithmes pour résoudre le DST peuvent être :

- à base de règles (*handcrafted*). Un des défauts d'une telle méthode est qu'elle souffre de la propagation des erreurs des modules précédents.
- supervisé suivant le succès notamment des approches *end-to-end*. L'utilisation de CNN a notamment montré ses performances [[Hashemi et al., 2016](#)]. Ces méthodes ont notamment profité de l'augmentation de corpus de données labellisés comme les Dialog State Tracking Challenges (DSTCs) (tel DSTC2 : [[Williams et al., 2016](#)]) permettant de standardiser l'évaluation de modèles de machine learning. Ces données sont cependant *off-line* ce qui implique généralement que ces modèles sont entraînés indépendamment du choix de la politique du système (module Dialogue Policy), ce qui peut les empêcher de généraliser correctement. Le manque de données peut aussi être un problème pour une utilisation en ligne de ce module.
- non-supervisé grâce aux modèles *sequence-to-sequence* auto-régressifs comme BERT ou GPT-3. L'utilisation de large modèle de langues préentraînés offre alors l'avantage d'un apprentissage *few-shot* voire *zero-shot* [[Hu et al., 2022](#)]. Il s'agit néanmoins de très gros modèles qui correspondent souvent à des approches *end-to-end*.
- à l'aide de RL ou Deep RL, permettant notamment un apprentissage en ligne [[Chen et al., 2020](#)] et l'utilisation de feed-backs humains.

Il est à noter que les deux aspects DST et DP du DM ne sont généralement pas traités séparément. Le POMDP permet ainsi de prendre en compte à la fois la représentation incertaine de l'état du dialogue et la décision de l'action à choisir selon les récompenses du système.

Choix de l'action : DP

Il existe différentes approches pour la conception d'un DM selon la manière dont les actions sont décidées :

- *rule-based* : Un ensemble de règles contrôle le choix de la réponse du système, comme dans le cas d'ELIZA (1966). Ces règles sont souvent très dépendantes du domaine et de la tâche considérés.
- *data-driven* : Le DM est entraîné à partir de corpus de dialogues, d'un simulateur ou d'interactions successives avec des utilisateurs, de manière en ligne. Lorsque l'entraînement est de bout en bout (end-to-end), cette méthode est moins dépendante d'un domaine spécifique et de ses contraintes mais nécessite une grande quantité de données d'entraînement.
- *hybride* : Il s'agit d'un mélange des 2 autres approches qui permet notamment de pouvoir insérer des connaissances expertes externes sur la tâche en question.

Dans la suite, nous nous concentrerons sur l'apprentissage par renforcement du DM, développé davantage au chapitre suivant.

4.2.4 Génération en langue naturelle et synthèse vocale

Le module de *Natural Language Generation* (Génération en Langage Naturel) (NLG) a pour but de générer les réponses à énoncer à l'utilisateur en fonction de la décision faite par le DM.

Il existe plusieurs approches selon la manière dont les réponses sont produites :

- *retrieval-based* : Les réponses sont reconstituées à partir d'un ensemble de réponses prédéfinies.
- *generative-based* : En entraînant un modèle génératif, de nouvelles réponses sont alors possibles. Cette méthode est cependant difficile à entraîner et nécessite une grande quantité de données. Les succès récents de modèles comme ChatGPT semblent avoir résolu beaucoup des limitations des anciens modèles, même s'ils n'offrent pas encore de garantie explicite de résultats (avec des risques de réponses erronées ou injurieuses, bien qu'ils restent faibles).

Dans notre cas, nous avons opté pour un module *retrieval-based* (les réponses sont déjà prédéfinies) pour nous concentrer sur le choix des stratégies plutôt que sur leur génération ou formulation. En effet, la génération de réponses apporte une variabilité non contrôlée qui peut nuire à notre évaluation de l'influence des nudges. D'autre part, ne disposant que de peu de données d'entraînement, un module *generative-based* nécessiterait probablement l'utilisation d'un lourd modèle transformers pré-entraîné.

La synthèse vocale, de manière symétrique à l'ASR, cherche à transformer la réponse explicite du système en signal audio. Une approche classique est de concaténer de petits segments audio pré-enregistrés avec une seule voix [Hunt and Black, 1996]. Une autre approche est d'utiliser un modèle TTS (*Text-to-Speech*) ou d'entraîner un modèle similaire à ceux utilisés pour l'ASR. Il peut alors être intéressant d'adapter la voix (genre, intensité, etc.) pour augmenter l'engagement de l'utilisateur.

4.2.5 Challenges de l'approche pipeline et approches end-to-end

Si l'approche pipeline correspond à l'architecture traditionnellement employée dans les systèmes de dialogue orienté tâches, elle possède néanmoins certaines contraintes et défis à relever :

- Elle est *coûteuse à mettre en place* : Chaque module doit en effet être conçu, entraîné et évalué séparément, ce qui nécessite la création de multiples annotations et généralement de règles (*hand-crafted*) [Wen et al., 2017].

Une des manières de répondre à ce problème est d'avoir recours à de l'apprentissage par renforcement [Riou et al., 2018]. L'espace des actions et des états a cependant besoin d'être bien définis pour la tâche [Young et al., 2010]. L'ajout de règles (*handcrafted*) pour aider la représentation de ces espaces peut alors rendre difficile l'extension du système à d'autres domaines.

- Elle conduit à une *propagation des erreurs* : Chaque module dépendant du précédent, les erreurs se propagent le long de l'architecture. De même, les feedbacks du système étant généralement obtenues à la fin, il peut exister un problème *d'assignement de crédit* : il est difficile de dire quel module est responsable de quelle récompense. L'*interdépendance* des modules implique également de devoir adapter tous les modules lorsque l'un d'entre eux est modifié [Zhao and Eskenazi, 2016].

Pour ces raisons, la recherche actuelle tend à se concentrer sur des approches *end-to-end*, où l'ensemble du dialogue est entraîné en même temps. Les premières approches cherchaient notamment à établir une correspondance directe entre l'historique d'une conversation et la réponse suivante du système, pour des systèmes orientés conversationnels [Vinyals and Le, 2015], puis orientés tâches [Wen et al., 2017]. Aujourd'hui, les systèmes de dialogue reposent souvent sur l'emploi de modèles de langues et d'architectures *transformers* (cf section 5.2).

Néanmoins, l'approche *end-to-end* nécessite de très grandes quantités de données d'entraînement. C'est une des raisons qui nous a fait préféré une approche *pipeline* (cf section 8).

4.3 Evaluation

L'évaluation d'un système de dialogue fait traditionnellement intervenir un jugement humain, avec les contraintes de disponibilité et d'organisation associées. De nombreux travaux ont alors cherché à réduire l'implication humaine, comme le rapporte une enquête sur les méthodes d'évaluation des systèmes de dialogue [Deriu et al., 2020].

Néanmoins, l'automatisation de l'évaluation des systèmes de dialogue se heurte à un certain nombre de challenges :

- la difficulté de définir et mesurer exactement le succès d'un dialogue, en particulier pour un système de dialogue ouvert. Le succès est aussi souvent dépendant de la tâche.
- la difficulté de prendre en compte les performances de tous les modules
- il s'agit généralement d'une méthode longue et coûteuse, en particulier si elle implique des utilisateurs humains.

Pour un système de dialogue modulaire, l'évaluation peut alors être unitaire sur les différents modules ou jointe.

4.3.1 Evaluation unitaire

Différentes métriques standards (précision, ROC) sont possibles pour l'évaluation unitaire bien qu'ils puissent y avoir pour certains modules un grand nombre de sorties valides.

De manière générale, l'entrée est évaluée par rapport à une référence, généralement à l'aide de mesures classiques relatives aux fouilles de texte. Celles-ci reposent alors souvent sur des analyses fréquentielles : caractéristiques et longueurs des mots et du texte, comme le *Word Error Rate* (taux d'erreur de mots) (WER), le *Concept Error Rate* (taux d'erreur de concepts) (CER) ou le *Slot Error Rate* (taux d'erreur de slots) (SER). Des métriques de classification (f1-mesure, UAR) sont également utilisées, notamment pour l'analyse de sentiments.

L'évaluation du DM est dépendante du type d'algorithmes utilisés selon qu'il s'agit d'un entraînement supervisé (précision, rappel, f1-score, etc.) ou avec renforcement (évolution de la récompense moyenne dans le dialogue par exemple).

La génération des réponses est quant-à-elle évaluée à l'aide de métriques de traduction et résumé automatique : BLEU, ROUGE, METEOR, etc. Ces métriques ne sont néanmoins pas nécessaires dans le cas d'une génération *retrieval-based* comme c'est le cas dans cette thèse.

4.3.2 Evaluation jointe

Dans les systèmes de dialogues orientés tâches, l'évaluation jointe se fait en fonction de 2 aspects : le *taux de succès de la tâche* et l'*efficacité du dialogue* (mesurée en général selon sa longueur).

Pour automatiser cette évaluation, 2 métriques principales existent selon [Deriu et al., 2020] :

- *La modélisation de la satisfaction de l'utilisateur* : l'aspect fonctionnel du système est mesuré par la satisfaction de ses utilisateurs, récoltée grâce à des questionnaires par exemple. Ces mesures peuvent alors servir de labels cibles pour entraîner le système. Un exemple d'une telle méthode d'évaluation est le cadre PARADISE (*PARAdigm for Dialog System Evaluation*) [Walker et al., 1997].
- *La simulation des utilisateurs* : Cette méthode permet de tester à loisir les performances du système et de proposer un environnement pour un apprentissage par renforcement. Un exemple est le simulateur d'utilisateur *agenda-based* [Schatzmann et al., 2007] qui consiste à générer aléatoirement pour un utilisateur un but (sous forme d'une liste de valeurs pour les variables d'intérêt du domaine) et un agenda, c'est-à-dire une liste dynamique d'actes de dialogue pour satisfaire ce but.

Si la simulation d'un interlocuteur est potentiellement plus efficace pour l'apprentissage et le test, il reste préférable de conduire l'évaluation jointe finale sur

de véritables utilisateurs. Les enjeux relatifs à l'entraînement par renforcement d'un système sont développés plus précisément dans le chapitre suivant.

Une tâche classique pour les systèmes de dialogue orientés est par exemple de recommander un produit, un lieu à un utilisateur à travers un système de *slot-filling* généralement, c'est-à-dire trouver les variables d'intérêt inconnues comme le type de restaurant recherché, la gamme de prix, etc.) (cf DSTC2 [Williams et al., 2016], MultiWoZ [Budzianowski et al., 2018]).

Pour pouvoir comparer différents modèles d'apprentissage par renforcement sur le dialogue, des cadres de références ont également été proposés comme le *benchmark* d'environnements proposé par [Casanueva et al., 2017].

Chapitre 5

Méthodes d'apprentissage d'un système de dialogue

Les systèmes de dialogue utilisent en général soit des modèles sequence-to-sequence, soit des algorithmes d'apprentissage par renforcement. Dans cette partie, nous allons tout d'abord présenter les principales approches pour entraîner un système de dialogue avec du RL ou *Reinforcement Learning* (Apprentissage par Renforcement). Nous verrons ensuite les avancées récentes en traitement du langage à l'aide des *transformers*. Enfin, nous verrons différents efforts de personnalisation d'un système de dialogue à son interlocuteur.

5.1 Apprentissage par renforcement

5.1.1 Paradigme général, Modélisation en problème de RL

L'apprentissage par renforcement ou RL, initialement employé majoritairement pour des tâches de contrôle ou de jeux, est de plus en plus utilisé dans les systèmes de dialogue, notamment pour ceux qui sont personnalisés [den Hengst et al., 2020]. Le principe du RL est d'apprendre à un agent (le système) par essais et erreurs (*trials and errors*) quelles actions prendre dans un environnement pour maximiser ses récompenses cumulées dans celui-ci.

Plus précisément, on utilise généralement le cadre théorique du POMDP [Sondik, 1971], dans lequel les états du système ne sont pas directement accessibles et où l'agent va plutôt avoir accès à des observations de l'environnement. On appelle alors **politique** la fonction qui à chaque état préconise une action à exécuter, permettant de transitionner vers un nouvel état. L'objectif d'un algorithme de RL est ainsi d'estimer la politique optimale pour maximiser les récompenses. Une des problématiques est donc de définir des récompenses

pertinentes pour qu'un système de dialogue puisse apprendre la tâche voulue malgré le fait qu'on ne sache souvent qu'à la fin du dialogue si elle a été un succès ou non.

Dans le paradigme du MDP, une récompense est obtenue à chaque changement d'états résultant d'une action de l'agent. Sa résolution consiste alors à chercher une **politique optimale** qui maximise l'espérance des récompenses cumulées. La manière classique de comptabiliser ces récompenses dans ce paradigme est d'employer la notion de **gain** comme critère d'optimisation. Celui-ci correspond à la somme des récompenses R_i obtenues pour chaque état i , selon cette politique, pondérée par un paramètre γ (entre 0 et 1) afin de pénaliser les récompenses trop lointaines. Plus ce paramètre γ est faible, plus les récompenses des états suivants seront de moins en moins considérées.

Le critère de gain G selon une politique π à l'état t , dans le cas d'un horizon infini, est alors définie par la formule suivante :

$$G_t^\pi = \sum_{i=0}^{\infty} \gamma^i R_{t+i}$$

Pour apprendre une politique qui maximise le gain attendu, 2 cas peuvent être distingués selon le fait qu'un modèle parfait de l'environnement (comprenant notamment la connaissance de la matrice de transitions entre états) soit disponible ou non. Dans le premier cas, il s'agit d'algorithmes *model-based*, reposant souvent sur des méthodes de programmation dynamique. La plupart des algorithmes sont cependant *model-free* et doivent ainsi explorer leur environnement suffisamment pour pouvoir apprendre une politique optimale.

Deux approches sont alors possibles : l'approche *value-based* et l'approche *policy-based*, qui peuvent également être combinées dans un même algorithme. Les algorithmes *model-free* doivent alors faire face au dilemme *exploration-exploitation* : continuer de choisir des actions pour explorer l'environnement ou exploiter les connaissances déjà acquises pour prendre une action estimée meilleure.

Le cas model-based

Le but de ces méthodes est de calculer les politiques optimales à partir d'un modèle parfait de l'environnement sous la forme d'un MDP. Des exemples classiques sont les algorithmes *Value Iteration* [Bellman, 1957] et *Policy Iteration* [Howard, 1960]. En pratique, ces algorithmes sont peu utilisables car ils sont

coûteux en calculs et nécessitent un modèle parfait, mais ils restent théoriquement intéressants [Sutton and Barto, 2014]. Pour des états de grande dimension, ces méthodes model-based sont peu applicables. Elles peuvent néanmoins être utilisées (au moins partiellement) pour des environnements bien connus comme dans le cas d'*AlphaZero* un algorithme général pouvant être entraîné sur divers jeux en jouant contre lui-même (*self-play*), sans connaissances du domaine excepté les règles du jeu, jusqu'à atteindre des compétences supérieures aux champions humains dans des jeux complexes comme les échecs [Silver et al., 2017].

L'approche value-based

L'approche *value-based* cherche à estimer à quel point il est favorable pour un agent de se trouver dans un certain état (ou d'effectuer une certaine action dans cet état). Plus précisément, l'objectif de cette approche est d'avoir accès à la valeur d'un état, ou d'un couple état-action, en termes de gain attendu à partir de cet état selon une politique donnée. Il s'agit alors d'apprendre une **fonction d'utilité** qui permet ensuite de choisir la meilleure action localement, dans un état donné, sans avoir besoin pour cela de réaliser une exploration (simulée) en avant. La plupart des algorithmes de RL utilisent ce paradigme, qui nécessite donc une exploration suffisante de l'environnement pour apprendre une bonne fonction d'utilité.

Les deux fonctions d'utilité employées sont la fonction de valeur $V : S \rightarrow \mathbf{R}$ et la fonction valeur-action ou Q fonction $Q : S \times A \rightarrow \mathbf{R}$, définies comme l'espérance moyenne du gain selon une politique π et un couple initial d'état s_0 et d'action a_0 :

$$V_{\pi}(s_0) = E_{\pi}[G_t | s_t = s_0]$$

$$Q_{\pi}(s_0, a_0) = E_{\pi}[G_t | s_t = s_0, a_t = a_0]$$

Il existe notamment la relation : $V_{\pi}(s) = Q_{\pi}(s, \pi(s))$.

Ces fonctions d'utilité peuvent également être définies récursivement par :

$$V_{\pi}(s) = E[R_t + \gamma V_{\pi}(s_{t+1}) | s_t = s, a_t = \pi(s)]$$

La politique optimale π^* consiste alors à maximiser la fonction de valeur :

$$\pi^*(s) = \operatorname{argmax}(\pi)(V_{\pi}(s))$$

Une fonction de valeur V^* associée à cette politique optimale est alors obtenue par l'équation d'optimalité de Bellman (5.1). De même Q^* , la fonction Q associée, se définit de manière similaire (5.2) :

$$V^*(s) = \max_a \sum_{s' \in S} P(s, a, s') (R(s, a) + \gamma V^*(s')) \quad (5.1)$$

$$Q^*(s, a) = \sum_{s' \in S} P(s, a, s') (R(s, a) + \gamma \max_{a'} Q^*(s', a')). \quad (5.2)$$

Une fois Q^* appris, la politique optimale consiste alors à prendre l'action qui maximise Q^* :

$$\pi^*(s) = \arg \max_a Q^*(s, a)$$

Pour apprendre une politique optimale en l'absence de modèle de l'environnement (cas *model-free*), plusieurs méthodes d'exploration existent, comme l'approche de Monte-Carlo ou celle des différences temporelles TD (*Temporal Differences*). Elles cherchent à mettre à jour une valeur selon la règle d'amélioration suivante :

NouvelleEstimation = AncienneEstimation + Pas[CIBLE – AncienneEstimation]

De manière générale, ces méthodes prennent une forme d'estimation par moyennage qui serait calculée incrémentalement. Plus précisément, le *TD-learning* prend l'expression suivante :

$$V(s_t) \leftarrow V(s_t) + \alpha [R_{t+1} + \gamma V(s_{t+1}) - V(s_t)]$$

Il permet ainsi de mettre à jour la fonction d'utilité incrémentalement par ré-estimations successives et locales.

Q learning :

Parmi les algorithmes *value-based*, l'un des plus connus est celui du *Q-learning* [Watkins, 1989] faisant appel à une forme d'apprentissage TD. Il consiste à estimer les valeurs Q d'une manière "en ligne", en mettant à jour la fonction Q à chaque action prise selon la règle suivante :

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [R_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)]$$

Cet algorithme offre alors des garanties de convergences, puisque pour un nombre infini d'action, les valeurs de Q calculées convergent vers Q^* , permettant d'obtenir une politique optimale. Sous sa version initiale, le Q-learning repose donc sur une représentation tabulaire des valeurs de la fonction Q, qui est peu utilisable pour des espaces continus ou de grandes dimensions. Cet

algorithme a ensuite été adapté à de très nombreuses approches, proposant différentes manières d'estimer la fonction Q, à l'aide de filtre de kalman (*Kalman Temporal Difference* (Différences temporelles par filtre de Kalman) (KTD)) tel KTD-Q [Geist and Pietquin, 2010] par exemple. Le développement du *Deep learning*, a également donné lieu au *Deep Q Network* (Réseau profond modélisant la fonction Q) (DQN)[Mnih et al., 2013] qui a contribué au démarrage du Deep Reinforcement Learning. Cet algorithme a depuis connu de nombreuses améliorations : Double DQN, Dueling DQN, etc. Différentes utilisations d'algorithmes *value-based* dans le dialogue sont notamment présentées dans la table 5.1 ci-dessous.

L'algorithme Q-learning, et ses algorithmes dérivés, sont des algorithmes *off-policy*, c'est-à-dire que la politique à apprendre est améliorée en utilisant les interactions collectées par une seconde politique, appelée politique de contrôle. Cela permet notamment un apprentissage à partir d'une politique de contrôle sous-optimale mais garantissant une exploration suffisante. Les méthodes *off-policy* sont ainsi capables de généraliser sur des corpus et des données collectées par des politiques souvent sous-optimales. En effet cette propriété vient du fait que ces méthodes font appel à l'équation d'optimalité de Bellman qui utilise l'opérateur non-linéaire max dans l'optimisation.

Les algorithmes *value-based* peuvent également être *on-policy*, c'est-à-dire qu'ils utilisent la même politique pour évaluer et améliorer le choix des actions. Ils ont ainsi l'avantage de converger plus vite mais sont dépendants des trajectoires observées dans l'environnement. Un exemple classique est par exemple l'algorithme *State-Action-Reward-State-Action* (Etat-Action-Récompense-Etat-Action) (SARSA), ou un de ses dérivés, l'algorithme GP-SARSA [Gasic and Young, 2013], utilisant le mécanisme des Processus Gaussien pour approximer la fonction action-valeur à partir de laquelle une politique optimale peut ensuite être obtenue.

L'approche policy-based

Plutôt que d'apprendre une fonction de valeur, une autre approche est d'apprendre directement la fonction politique, reliant pour chaque état l'action à prendre. L'action est alors décidée en échantillonnant parmi certaines politiques. La plupart de ces algorithmes sont *on-policy*. L'optimisation de la politique se fait généralement à l'aide d'un calcul de gradient. Cette approche offre les avantages d'avoir une bonne convergence, ainsi que de bonnes performances dans des espaces d'états de grandes dimensions ainsi que continus. Elle est cependant dépendante de l'environnement exploré et peut donc converger vers un optimum local plutôt que global. Elle souffre aussi de hautes variances, même

si différentes méthodes ont été proposées pour la réduire, notamment les algorithmes *Actor-Critic* (AC) et *Advantage Actor-Critic* (A2C). La plupart des algorithmes *policy-based* reposent sur la méthode **actor-critic** qui modélise 2 fonctions à l'aide de réseaux de neurones pour apprendre la politique :

1. Le *critique* estime la fonction d'utilité (fonction de valeur ou d'action-valeur)
2. L'*acteur* met à jour la distribution de la politique dans la direction suggérée par le critique (selon le gradient de la politique par exemple).

Parmi les algorithmes *policy-based* appliqués au dialogue, nous pouvons alors citer des algorithmes de *deep-learning* (*actor-critic*) comme ACER [Weisz et al., 2018] ou l'algorithme PPO [Schulman et al., 2017] utilisé par exemple dans ChatGPT.

Dilemme Exploration-Exploitation

Différentes approches existent pour répondre au dilemme d'*exploration-exploitation*. La plus connue est celle *ε-greedy* (ou *ε-glouton*).

Elle consiste à sélectionner une action selon un hyperparamètre ϵ compris entre 0 et 1 :

- selon une probabilité ϵ : une action est choisie aléatoirement pour explorer davantage l'environnement
- sinon : l'action donnant la valeur maximum pour la fonction Q est choisie, privilégiant ainsi l'exploitation des connaissances acquises

Généralement, la valeur de epsilon est réduite progressivement durant l'apprentissage au fur et à mesure que l'environnement a été estimé exploré.

Néanmoins, une exploration aléatoire peut être problématique face à de vrais utilisateurs. D'autres critères de décision ont ainsi été proposés comme l'approche *bonus-glouton* [Ferreira, 2015] qui permet une exploration plus subtile en se basant sur la moyenne courante estimée pondérée par l'incertitude de la fonction Q. L'action à l'état t est alors choisie selon la formule :

$$a_t = \arg \max_a \mu_Q(s_t, a) + \beta \frac{\sigma_Q^2(s_t, a)}{\beta_0 + \sigma_Q^2(s_t, a)}$$

où β et β_0 sont deux hyperparamètres constants et $\mu_Q(st, a)$ et $\sigma_Q^2(st, a)$ sont respectivement la moyenne et la variance de l'estimation actuelle de la fonction Q.

L'exploration peut également être guidée, en particulier pour les premiers pas, par des démonstrations à l'aide d'un utilisateur expert par exemple. Il existe enfin des approches sensibles aux risques dans l'exploration, comme l'approche Budgeted-Fitted-Q [Carrara, 2018] qui cherche à construire puis transférer une politique générique sûre pour gérer plus facilement les premières interactions avec l'utilisateur cible.

Modélisation des récompenses

Les récompenses dans le dialogue dépendent généralement du succès de la tâche et de la rapidité à l'effectuer. Dans le cas de dialogues orientés tâches, les récompenses sont alors relativement explicites, reposent souvent sur des connaissances expertes et dépendent du domaine. Néanmoins, il faut a priori attendre la fin du dialogue pour obtenir une récompense relative à son succès. Cela pose des difficultés de convergence et se révèle peu applicable en pratique où l'apprentissage doit parfois être réalisé en ligne sur de véritables utilisateurs.

Une solution est alors d'ajouter des feedbacks durant l'apprentissage. Ces feedbacks peuvent être explicites et être ajoutés dans la récompense selon le concept de *reward shaping* ou être implicites et être intégrés dans l'état du dialogue.

Le concept de *reward shaping* est ainsi une solution envisagée depuis longtemps en RL. Il s'agit d'ajouter des récompenses supplémentaires durant le dialogue (à partir d'une connaissance experte notamment) pour éviter les récompenses sinon faibles et retardées, ce que Minsky [1961] nomme le *temporal credit assignment problem*. Un exemple de feedback explicite est ainsi d'utiliser un *signal social* : l'utilisateur (expert ou non) ajoute à chaque tour de parole une récompense supplémentaire en fonction de son ressenti actuel de l'interaction [Ferreira, 2015].

Pour les systèmes davantage conversationnels, il peut également être pertinent de définir un modèle de récompense s'il est possible de l'apprendre. C'est par exemple le cas de modèles de l'état de l'art comme ChatGPT ou InstructGPT [Ouyang et al., 2022] qui apprennent un modèle de récompense à partir de feedbacks humains (RLHF) et qui sont expliqués davantage dans la section suivante. Des méthodes d'*Inverse Reinforcement Learning* (Apprentissage par Renforcement Inversé) (IRL) où l'agent apprend une fonction de récompense sur une base de données collectée, peuvent également aider à simuler des utilisateurs [Chandramohan et al., 2011] mais nécessitent généralement une grande quantité de données.

Une liste non exhaustive des différents algorithmes de RL utilisés ces dernières années dans les systèmes de dialogue, avec leur paramétrisation de récompenses, est donnée dans la Table 5.1.

Le DM étant la partie centrale du système de dialogue, nous présentons également les challenges de la recherche actuelle pour son apprentissage.

5.1.2 Challenges pour l'apprentissage du DM

Selon une enquête réalisée par [Dai et al. \[2021\]](#), l'apprentissage du *Dialogue Manager* souffrent généralement de 3 défauts :

- *peu de perspectives d'évolution (scalability)* : Un des enjeux est notamment de prendre en compte le potentiel changement d'intention de l'utilisateur ou bien gérer des intentions encore non observées. Des algorithmes tenant compte de la non-stationnarité peuvent alors être utilisés. [Wang et al. \[2018\]](#) proposent également d'utiliser le cadre d'apprentissage *Professeur-Elève* où l'ancien modèle ainsi que des règles logiques pour les nouvelles intentions détectées sont utilisées pour enseigner au nouveau modèle.
- *des données labellisées insuffisantes* : L'apprentissage des dialogues orientés-tâche dépend fortement de la qualité des données labellisées. Leur annotation étant généralement coûteuse, plusieurs options sont alors possibles : automatiser la labellisation des données, optimiser leur collecte ou encore exploiter l'information de données non-labellisées pour apprendre leur structure [[Shi et al., 2019](#)]. Dans cette thèse, nous nous sommes principalement tournés vers des approches d'automatisation pour la collecte et l'annotation des données.
- *une faible efficacité d'entraînement* : Différents algorithmes efficaces par échantillon ont été proposés (cf section 5.1.3). Des méthodes pour diviser l'espace des actions en différents niveaux ont aussi été présentés, comme le HRL (*Hierarchical Reinforcement Learning*), un apprentissage par renforcement hiérarchisé [[Peng et al., 2017](#)]. Une autre solution est d'utiliser des données collectées sur de vrais utilisateurs pour mettre à jour de manière itérative un modèle de l'environnement sur lequel continuer l'entraînement, comme l'algorithme *Deep Dyna-Q* [[Peng et al., 2018](#)].

Publication	Algorithme	on/off-policy	récompenses	Domaine
Ferreira [2015]	KTD	les deux	<i>Reward Shaping</i> (RS) : signal social émotionnel explicite	orienté tâche, slot-filling
Gordon et al. [2016]	Q-learning	off	RS : valence et engagement mesurés à partir d'expressions faciales	orienté tâche, jeu pour apprendre une seconde langue
Weisz et al. [2018] (25)	ACER	off	pas de RS	orienté-tâche, slot-filling pour simulateur <i>agenda-based</i>
Carrara [2018]	Budgeted-Fitted-Q	off	pas de RS	orienté-tâche, slot-filling
Ritschel [2022]	Q-learning, approximation par fonction linéaire et k-bandits armés	off	RS : signal social explicite ou implicite multimodal	orienté-tâche : divers (choix d'extraversion pour le dialogue, raconter une blague, etc.)
ChatGPT, Open AI, 2023	PPO	on	RLHF : modèle appris	conversationnel

TABLE 5.1 : Principaux algorithmes de RL pour l'apprentissage de systèmes de dialogue

Nous allons voir à présent comment un apprentissage en ligne peut être réalisé de manière efficace dans le dialogue, en particulier en termes de données.

5.1.3 Apprentissage en ligne efficace

De nombreuses approches existent faisant appel à des algorithmes de RL pour résoudre le POMDP dans le cadre du dialogue. Les approches *model-based*, comme par exemple la première application du modèle POMDP à la problématique du dialogue [Roy et al., 2000], étant généralement assez coûteuses en données, nous nous intéressons ici uniquement aux approches sans modèles. On distingue alors 3 types d'approche :

- sur corpus (hors-ligne)
- en simulant des utilisateurs
- sur de vrais utilisateurs

Les approches **sur corpus** nécessitent qu'une certaine quantité de données pertinentes à la tâche de dialogue souhaitée aient déjà été collectées. On peut alors utiliser des algorithmes *off-policy* efficaces comme *Least Squares Policy Iteration* (LSPI) [Li et al., 2009]. Les algorithmes *off-policy* sont en effet préférables pour l'apprentissage sur corpus car ils permettent de généraliser sur des données collectées à partir d'une politique sous-optimale. Néanmoins, l'exploration de l'environnement reste borné par le corpus.

L'apprentissage **en ligne** (i.e. lorsque les données deviennent disponibles au fur et à mesure des interactions, de manière séquentielle) permet de se libérer de certaines de ces contraintes. La simulation d'utilisateurs peut sembler plus pratique en phase de développement mais le test du système avec de vrais utilisateurs est à privilégier pour l'évaluation finale. Pour simuler des utilisateurs, on a recours en général à un modèle utilisateur ainsi qu'à un simulateur d'erreurs pour mieux représenter l'incertitude des interactions utilisateurs-système.

Pour apprendre en des temps raisonnables (quelques centaines de dialogues) face à de vrais utilisateurs, un algorithme efficace par échantillon est alors primordial. Il existe plusieurs approches :

- à partir de *deep-learning* en utilisant par exemple l'algorithme *Actor-Critic with Experience Replay* (ACER) [Su et al., 2017][Weisz et al., 2018]. [Weisz et al., 2018] montrent alors que cet algorithme peut être utilisé même pour de large espaces d'action. Su et al. [2017] comparent notamment l'efficacité de différents algorithmes utilisant la technique *experience replay*. Celle-ci consiste à stocker les transitions observées par l'agent (quadruplets rassemblant à un instant t : état, action, récompense et état suivant),

dans une table par exemple, puis de les sélectionner de manière aléatoire lors de l'apprentissage. Utilisé avec un algorithme *off-policy*, cela permet d'obtenir une meilleure convergence (les données étant ainsi mieux identiquement distribuées) sans que les données rejouées aient besoin de suivre un ordre séquentiel. Cela rend également l'algorithme plus efficace en termes de données car les expériences peuvent être rejouées plusieurs fois permettant de renforcer la "mémoire" du modèle (en particulier lorsqu'il y a peu de variance entre les récompenses d'états similaires). Les auteurs montrent alors les performances des algorithmes *deep actor-critic* en termes de convergence, d'utilisation efficace des données et de taux de succès par dialogue.

- à partir d'algorithmes non paramétriques comme GPTD ou GP-SARSA [Gasic and Young, 2013]. Cet algorithme possède de bonnes propriétés de convergence mais souffre néanmoins d'un coût de calcul élevé, ce qui peut le rendre difficile d'utilisation pour des espaces d'actions de grande taille par exemple [Weisz et al., 2018].
- des approches permettant de traiter la non-stationnarité comme *Kalman Temporal Difference* (Différences temporelles par filtre de Kalman) (KTD) [Geist and Pietquin, 2010][Daubigney et al., 2012]. L'algorithme KTD est notamment considéré comme efficace par échantillon car il est fondé sur l'estimation des moments du second ordre et offre l'avantage de pouvoir être appliqué *on-policy* (comme l'algorithme KTD-Q) ou *off-policy* (comme l'algorithme KTD-SARSA).

Des apprentissages conjoints de différents modules ont également été proposés récemment, comme l'apprentissage efficace et simultané en ligne du DM et de l'analyseur sémantique proposé dans [Riou et al., 2018], dans lequel les auteurs discutent notamment des difficultés d'obtenir une stratégie stable durant la phase d'apprentissage.

Néanmoins, si un apprentissage en ligne de zéro est possible, collecter des données peut nous permettre d'initialiser ou d'enrichir le système. Nous allons alors voir dans la section 5.3 comment cet apprentissage par renforcement peut se faire de manière personnalisée.

5.2 Les modèles de langue Transformers

5.2.1 Motivations

Nous rappelons ici brièvement le fonctionnement des *transformers*, une architecture neuronale fortement utilisée dans les résultats de l'état de l'art en traitement automatique de la langue et de la parole. Le but de ceux-ci est de résoudre des tâches de *sequence-to-sequence* (convertir une séquence d'un domaine à un autre, par exemple traduire une langue) en gérant facilement les longues dépendances. Initialement, l'état de l'art pour modéliser l'aspect temporel de l'apprentissage séquentiel dans les langues comprenaient les algorithmes *Recurrent Neural Network* (Réseau de neurones récurrent) (RNN), *Long Short Term Memory* (Réseau à mémoire longue et court terme) (LSTM)[[Hochreiter and Schmidhuber, 1997](#)] et *Gated Recurrent Network* (Réseau récurrent à portes) (GRN)[[Chung et al., 2014](#)]. Les modèles RNN se présentent comme une succession de cellules (*hidden states*), chacune prenant en entrée la représentation de l'unité séquentielle courante (caractère, mot, etc.) ainsi que la sortie de la cellule précédente.

Une première limite de ces modèles est la gestion de la mémoire des événements précédents (qui se dégrade pour de longues séquences), ce que les modèles LSTM améliore en séparant les informations importantes à court terme et à long terme dans deux états différents (hidden state et cell state respectivement) dans leur succession de cellules.

Un second défaut de ces méthodes est qu'elles ne permettent pas de parallélisation car un LSTM doit avoir vu une séquence entière avant de pouvoir faire une prédiction. Cela pose notamment des limites computationnelles pour de longues séquences même si des astuces de factorisation sont possibles [[Kuchaiev and Ginsburg, 2017](#)].

Les transformers vont remédier à ces 2 problèmes en permettant un entraînement davantage parallélisé sur de très grandes quantités de données et en cherchant à extraire un "contexte" utile pour chaque mot, partant du principe que tous les mots n'ont pas le même poids sémantique dans une phrase. Ils reposent ainsi entièrement sur l'auto-attention pour calculer les représentations en hautes dimensions des entrées du modèle sans avoir recours à des réseaux récurrents ou convolutifs (CNN).

5.2.2 Principe

L'architecture entière d'un transformer est présentée dans la figure 5.1 De manière générale, un transformer est un *encoder-decoder*, une structure où :

1. l'*encoder* apprend à générer une représentation continue d'une séquence de symboles (mot, token, etc.),
2. puis, le *decoder* génère une séquence de symboles, un par un, à partir de cette représentation et ceci de manière auto-régressive (tous les nouveaux symboles déjà générés de la séquence sont ajoutés en entrée du decoder pour générer le symbole suivant). Il se distingue ainsi des modèles *sequence-to-sequence* classiques qui récupèrent en input seulement le dernier symbole généré.

Un point important à noter est que la séquence cible est donnée au decoder durant l'apprentissage. Cette technique, appelée *Teacher Forcing*, permet de donner un indice sur la séquence à obtenir. Cela permet notamment au modèle d'être moins sensible à une entrée potentiellement erronée et d'accélérer l'apprentissage.

Le decoder comme l'encoder sont en fait constitués d'une série de couches (6 pour l'architecture présentée dans l'article introduisant les transformers)[[Vaswani et al., 2017](#)], composées elles-mêmes de 2 éléments : un mécanisme d'auto-attention et un réseau *feed-forward* dense. Pour maintenir l'information sur l'ordre dans la séquence pour chaque terme, un encodage positionnel est effectué en amont pour l'encoder et le decoder.

Mécanismes attentionnels

L'apport des transformers est surtout dans la prise en charge efficace de l'attention dans l'entraînement de tâches *sequence-to-sequence*. Il repose sur deux mécanismes (cf figure 5.2).

Scaled Dot-Product Attention

Le premier mécanisme est le calcul de l'attention elle-même que les auteurs nomment "Scaled Dot-Product Attention". Elle est définie par la formule :

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

où d_k correspond à la dimension d'une query, la représentation (*embedding*) d'un terme d'une séquence, les queries d'une séquence étant regroupées dans la matrice Q.

L'attention définie ainsi consiste tout d'abord à mesurer la similarité entre 2 représentations d'une séquence de mots (de même longueur), stockées dans

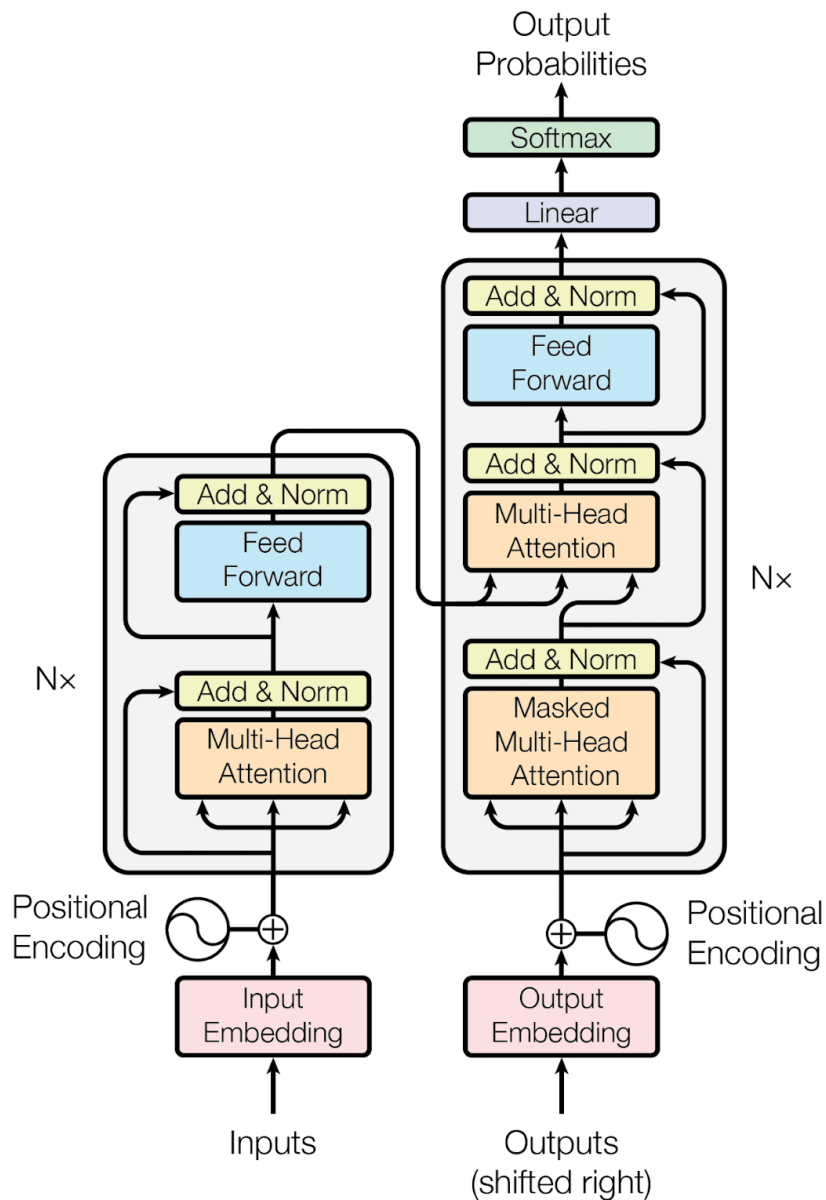


FIGURE 5.1 : Architecture Transformers (extrait de [Vaswani et al., 2017])

les matrices de poids Q et K, à travers leur calcul matriciel faisant intervenir des produits scalaires.

La matrice Q correspond alors à la représentation des *queries*, c'est-à-dire en un sens la séquence contextuelle, et la matrice K à celle des keys, la séquence d'intérêt. Deux cas peuvent être distingués :

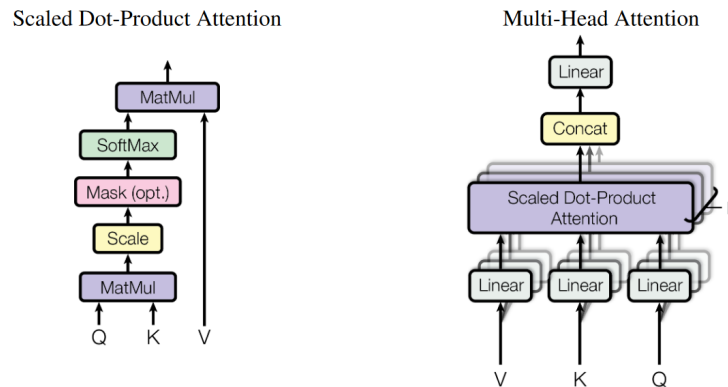


FIGURE 5.2 : Les 2 mécanismes attentionnels des transformers (extrait de [Vaswani et al., 2017])

- Lorsqu'il s'agit d'**auto-attention** (dans l'encoder et la première étape du decoder), ces matrices codent la même séquence. Dans l'encoder, elles codent ainsi la similarité entre chaque terme de l'input. Dans le decoder, cette auto-attention est masquée car elle s'effectue sur la séquence de sortie du decoder, obtenue de manière autorégressive : les outputs successifs du decoder sont passés en input et l'auto-attention ne doit donc être calculée que sur ceux disponibles actuellement.
- Dans le decoder, une seconde **attention** est également calculée (cf figure 5.1) prenant en compte l'information de la séquence initiale de l'encoder codée par la matrice K. La matrice Q reste alors relative à l'information de la séquence du decoder.

Après ce premier calcul matriciel, suit une normalisation (pour des gradients plus stables), puis une fonction softmax permet ensuite d'obtenir une distribution probabiliste de cette mesure de similarité entre les deux représentations de séquences. Ces nouveaux poids multipliés à la matrice V, représentant la même séquence que K, permettent enfin d'obtenir un poids attentionnel (relatif à chaque mot de la séquence représentée par Q) sur chaque mot de la séquence d'intérêt représentée par V.

Multi-head Attention

Le deuxième mécanisme attentionnel consiste à calculer plusieurs attentions plutôt qu'une seule afin d'obtenir simultanément des informations de sous-

espaces de représentations pour une même séquence, ce que les auteurs appellent “Multi-head attention”. Ces attentions jointes (8 têtes d’attention dans la version originale) sont ensuite concaténées en une seule mesure finale. La mesure finale d’attention connecte ainsi chaque terme de la séquence.

Avantages de la méthode

Les avantages des architectures transformers par rapport aux réseaux récurrents et convolutionnels sont alors :

- une complexité de calculs par couche plus faibles
- plus d’opérations parallélisables
- une longueur constante et plus faible du chemin à parcourir dans le réseau entre les dépendances à long-terme
- des résultats davantage interprétables selon les auteurs [[Vaswani et al., 2017](#)] (une visualisation des distributions de l’attention du modèle pour chaque exemple peut être affichée pendant l’apprentissage)

5.2.3 Modèles de langage préentraînés et applications au dialogue orienté tâche

L’architecture des transformers a été reprise dans de nombreux modèles de langage préentraînés PLM (*Pretrained Language Model*), entièrement ou partiellement. Il peut alors s’agir de modèles :

- *encoder-decoder* comme le modèle T5 [[Raffel et al., 2020](#)] (et ses modèles dérivés : mT5, FLAN-T5, etc.) Le modèle T5 utilise notamment une approche text-to-text où chaque tâche de NLP (traduction, classification, QA) est présentée sous forme de consigne texte en entrée pour l’apprentissage d’une sortie texte cible.
- seulement *encoder* tel BERT [[Devlin et al., 2019](#)] ou RoBERTa [[Liu et al., 2019](#)]. Des modèles entraînés spécifiquement sur des données françaises ont également été proposés comme CamemBERT [[Martin et al., 2020](#)] et FlauBERT [[Le et al., 2020](#)].
- seulement *decoder* comme GPT 3 ou ChatGPT d’OpenAI.

Ces modèles peuvent notamment être appris sur de grandes quantités de textes non annotés de manière auto-supervisée en cherchant à prédire par

exemple le token (unité sémantique faite d'un groupe de caractère) suivant dans la phrase (comme les modèles GPT 1,2,3 de OpenAI). Pour prendre en compte la bidirectionnalité du texte, le modèle encodeur BERT [Devlin et al., 2019], quant à lui, est entraîné sur deux tâches simultanément : la prédiction d'un token masqué dans la phrase et celle de la phrase suivante.

Cet apprentissage auto-supervisé sémantique peut également se faire à partir d'un signal audio, comme pour Wave2vec2 [Baevski et al., 2020] ou Audio ALBERT [Chi et al., 2020].

Dans le cas des dialogues orientés tâche, ces modèles PLM peuvent être adaptés à des données de dialogue comme TOD-BERT [Wu et al., 2020] qui continuent l'apprentissage auto-supervisé d'un modèle BERT sur 9 corpus de dialogues orientés-tâche. Ils montrent alors une amélioration importante des performances comparé au modèle BERT sur plusieurs tâches de dialogues : la reconnaissance de l'intention de l'interlocuteur, le suivi de l'état du dialogue (*Dialogue State Tracking*), la prédiction de l'acte de dialogue et la sélection de réponse.

D'autres travaux ont montré également l'avantage d'adapter des données au domaine cible du dialogue [Henderson et al., 2019][Hung et al., 2022]. Une méthode est alors de filtrer des données conversationnelles possédant les termes caractéristiques au domaine cible, par exemple des données conversationnelles de Reddit [Hung et al., 2022].

5.2.4 Avancées récentes et alignement de large modèles de langue (LLM)

Bien qu'il s'agisse davantage de chatbots conversationnels, les systèmes de dialogues de l'état de l'art ont recours à des *Large Language Model* (Large modèle de Langue) (LLM) comme GPT-3 (175 milliards de paramètres) qui sont capables de générer des textes similaires à celui que pourrait écrire un être humain. Néanmoins, un des défauts de l'entraînement des modèles de langue transformers est qu'il ne prend pas vraiment en compte l'importance de l'erreur. En effet, les techniques d'apprentissage classiques des modèles de langue - prédiction du token suivant ou d'un token masqué dans la phrase - se concentrent sur l'apprentissage de la structure statistique de la langue mais ne garantissent pas une compréhension sémantique profonde. Ainsi, le modèle aura du mal à différencier 2 mots statistiquement équivalents (ayant chacun de grandes occurrences dans le contexte de la phrase) mais de significations différentes. Ces modèles peuvent alors être mal alignés avec la tâche voulue par l'utilisateur, c'est-à-dire avoir des prédictions correctes mais qui ne sont pas

celles attendues.

Pour remédier à ce problème, le modèle ChatGPT utilise un apprentissage par renforcement à partir de feedbacks humains (RLHF) afin de prendre en compte plus précisément les préférences humaines. A partir d'un modèle de langue pré-entraîné (GPT-3.5) et fine-tuné de manière supervisée, les auteurs entraînent un modèle de récompense en demandant à des annotateurs de comparer et classer par ordre de pertinence les différents outputs du premier modèle (même méthode que pour InstructGPT [Ouyang et al., 2022]). Ce modèle de récompense est alors utilisé pour optimiser le choix des outputs à l'aide de l'algorithme PPO [Schulman et al., 2017] et fine-tuner ainsi de manière itérative le modèle initial.

Le modèle de langue LaMDA de Google, quant-à-lui, est entraîné directement sur des données de dialogue avec pour but de porter une attention particulière à la sensibilité (sensibleness) au contexte dans l'enchaînement d'une conversation.

5.3 Systèmes de dialogues personnalisés

5.3.1 Types de personnalisation

Nous avons vu dans la section 5.1 comment un système de dialogue peut être appris par renforcement. Lorsque cet apprentissage cherche à s'adapter de manière plus pertinente aux goûts et besoins d'un utilisateur, le terme de personnalisation [Fan and Poole, 2006] est alors utilisé. Ces approches se sont développés dans de nombreux domaines (santé, commerce, éducation), notamment pour des systèmes de recommandation [Mahmood et al., 2014].

De manière générale, les systèmes personnalisés cherchent à extraire un contexte personnel qui peut alors être modélisé à différents niveaux :

- *dans un modèle unique à travers l'état du dialogue* : Cette méthode permet alors une personnalisation progressive, émergeant lorsque suffisamment de données ont été collectées ou que le modèle d'utilisateur utilisé est suffisamment informatif [den Hengst et al., 2019]. L'espace d'états du POMDP peut alors être augmenté par d'autres caractéristiques relatives au contexte personnel de l'utilisateur. La plupart des systèmes personnalisés reposent sur cette approche [Gordon et al., 2016][Shi and Yu, 2018][den Hengst et al., 2020].
- *de manière groupée, à partir de clusters* : Cette approche peut être pertinente lorsque qu'un groupe de classes d'utilisateur aux comportements

similaires est à identifier. [den Hengst et al. \[2019\]](#) compare notamment une personnalisation à travers l'état du dialogue et une personnalisation en segmentant les utilisateurs en groupes puis en utilisant un modèle d'apprentissage par segment. Les performances des modèles semblent alors varier selon l'environnement, le domaine et l'algorithme employé (A2C, DQN, GP, ...). Ils montrent néanmoins que ces approches à base de RL sont généralement plus performantes qu'une méthode de référence faite à la main (*handcrafted gold standard*).

- *de manière séparée en générant un modèle par utilisateur* : Cette méthode peut être davantage coûteuse, en particulier si l'adaptation est dynamique, c'est-à-dire qu'elle se fait en temps réel. Une approche possible est d'utiliser un apprentissage par transfert. [Casanueva et al. \[2015\]](#) propose par exemple d'utiliser une métrique de similarité entre utilisateurs pour sélectionner quelles données transférer pour un nouvel utilisateur.

La personnalisation peut également servir des objectifs différents, selon que l'adaptation est fonctionnelle ou non-fonctionnelle [[Martins et al., 2019](#)] :

- *adaptation fonctionnelle* : contrôle ce que fait le système. L'objectif est alors de mieux prédire les futurs choix de l'utilisateur, comme la plupart des systèmes de recommandation.
- *adaptation non-fonctionnelle* : contrôle comment le système interagit avec l'utilisateur. L'adaptation n'est alors pas strictement nécessaire pour le succès de la tâche. Elle cherche plutôt à garder l'utilisateur plus investi avec une interaction plus naturelle, comme certains systèmes exploitant l'information émotionnelle et l'engagement de l'utilisateur.

La personnalisation d'un système de dialogue pose de nombreux challenges. Par exemple, la personnalisation d'un contexte humain est difficile à simuler et l'utilisation d'agents autonomes peut alors soulever des questions relatives à la vie privée. Plus précisément, une étude sur la personnalisation à l'aide de RL [[den Hengst et al., 2020](#)] propose un cadre pour classifier les différents paramètres de personnalisation possibles (cf table 5.2).

5.3.2 Modélisation du contexte personnel dans les systèmes de dialogue

La modélisation du contexte personnel d'un utilisateur est une tâche difficile. Une simulation de différents types d'utilisateurs peut néanmoins être effectuée

Catégorie	A#	Aspect	Description	Intervalle
Adéquation à la tâche	A1	Contrôle	La mesure dans laquelle l'utilisateur définit explicitement l'adéquation d'un comportement	Explicite - Implicite
	A2	Sécurité	Le degré d'importance accordé à la sécurité	Trivial - Critique
Connaissances préalables	A3	Modèles d'utilisateurs	La disponibilité a priori de modèles décrivant les réactions des utilisateurs au comportement du système	Indisponible - Illimité
	A4	Disponibilité des données	La disponibilité a priori des réponses humaines au comportement du système	Indisponible - Illimité
Nouvelles expériences	A5	Disponibilité des interactions	La disponibilité de nouveaux échantillons d'interactions avec des individus	Indisponible - Illimité
	A6	Sensibilité au respect de la vie privée	Le degré de préoccupation concernant le respect de la vie privée	Banal - Critique
	A7	Observabilité de l'état	La mesure dans laquelle toutes les informations nécessaires à la personnalisation peuvent être mesurées	Partiel - Complet

TABLE 5.2 : Cadre pour catégoriser les paramètres de personnalisation (traduit de [den Hengst et al., 2020], table 1)

selon l'objectif recherché, notamment si la tâche est suffisamment claire à formuler. Dans le cas d'un système de recommandation *slot-filling*, un modèle utilisateur *agenda-based* [Schatzmann et al., 2007] a par exemple été utilisé [den Hengst et al., 2019]. Le système cherche alors à s'adapter aux caractéristiques individuelles et au but de chaque utilisateur. L'adaptation peut également se

faire selon une caractéristique particulière de l'individu, comme dans une des expériences proposées par [Ritschel \[2022\]](#) où le degré d'extraversion de l'utilisateur est simulée de manière non-stationnaire.

De nombreux systèmes personnalisés utilisent des critères d'*engagement* [[Gordon et al., 2016](#)][[Ritschel, 2022](#)] et de *feedbacks émotionnels* pour décrire et mesurer le contexte personnel d'un utilisateur.

En effet, l'ajout d'informations sentimentales permet de réduire la durée du dialogue et d'améliorer le taux de réussite de la tâche, comme l'ont montré [Shi and Yu \[2018\]](#) sur une tâche de recherche d'informations sur les autobus.

[Gordon et al. \[2016\]](#) compare également l'effet d'un feedback affectif personnalisé et non personnalisé dans l'apprentissage des langues avec un robot social pour enfants et constate que la personnalisation a un impact significatif sur la valence psychologique lors de l'interaction.

Dans leur étude, [den Hengst et al. \[2020\]](#) examinent les stratégies d'évaluation utilisées dans les systèmes personnalisés avec du RL. Les auteurs montrent notamment qu'une grande majorité des algorithmes de RL utilisés pour de la personnalisation ne le sont qu'une seule fois. L'algorithme Q-learning et ses algorithmes dérivés semblent néanmoins être les plus populaires.

Chapitre 6

Conclusions sur l'état de l'art

En conclusion, cet état de l'art (chapitres 2 à 5) permet d'établir de premières pistes de recherche sur les hypothèses formulées. Nous verrons alors dans la partie II comment elles peuvent être testées.

Comment étudier les nudges linguistiques et leur personnalisation dans le dialogue ?

Le Chapitre 2 a permis de montrer les résultats existants dans la littérature concernant l'efficacité des nudges numériques. Il y a très peu d'études actuellement sur les nudges linguistiques et sur les nudges personnalisés, sujet de cette thèse. Dans le cadre de nudges linguistiques, nous formulons alors 4 hypothèses de recherche sur leur efficacité et fonctionnement dans l'interaction orale, rappelées ci-dessous :

1. Nous supposons que le type d'agent (humain, robot ou enceinte) peut avoir une influence sur l'efficacité des nudges.
2. Nous faisons l'hypothèse qu'un système de dialogue utilisant des questions de nudges oraux pourrait changer significativement le choix d'un utilisateur pendant le dialogue comparativement à des questions sans nudges.
3. Nous supposons qu'une adaptation personnelle de stratégies de nudges avec RL est significativement plus efficace pour influencer une personne.
4. Pour estimer l'état actuel du dialogue ainsi que l'état mental et émotionnel de la personne, nous choisissons d'extraire certaines informations émotionnelles, paralinguistiques et contextuelles que nous supposons pertinentes pour prédire les nudges.

Comment tester l'impact de la représentation de l'agent dans l'interaction sur l'efficacité des nudges ?

Un biais envers la manière dont sont perçues les actions des machines semble exister. Néanmoins, il n'y aurait pas toujours de différences significatives entre un agent robotique physique et un agent virtuel [Tussyadiah and Miller, 2019]. Il s'agira alors de confirmer ou non cette tendance dans nos expériences.

Comment créer un environnement de recherche pour tester les nudges linguistiques ?

Pour personnaliser les nudges, nous avons utilisé la détection du comportement émotionnel de la personne. Le chapitre 3 présente un état de l'art sur la détection des émotions.

L'analyse des différentes architectures des systèmes de dialogue du Chapitre 4 a permis de montrer les avantages de choisir un système de dialogue orienté tâche comme environnement de recherche. Il offre la possibilité de limiter le domaine sémantique et de permettre une évaluation plus simple du succès d'un nudge. Un système de génération de réponses déterministes (retrieval-based) permettra notamment d'étudier uniquement l'impact du choix de la stratégie de nudge linguistique.

Sur quelles données entraîner le système ?

Le Chapitre 4 a pu présenter différents corpus adaptés à l'entraînement des systèmes de dialogue et de leurs modules. Néanmoins, l'absence de données spécifiques aux nudges nous pousse donc à entreprendre des expériences de collecte qui seront développées dans la partie II.

Quel apprentissage est possible avec peu de données ?

Qu'il s'agisse des modules d'entrée (reconnaissance automatique de la parole, détection d'émotion, compréhension de la parole) ou des stratégies du DM pour s'adapter à l'utilisateur, nous avons vu différentes méthodes qui permettraient de réaliser leur apprentissage en utilisant la petite quantité de données collectées par nos expériences.

Dans le cas des modules ASR et SLU, des modèles pré-entraînés peuvent être utilisés comme ont pu le montrer les Chapitres 4 et 5. Pour l'apprentissage des émotions, le Chapitre 3 a exposé des techniques d'apprentissage *few-shot* pour la détection d'émotions. En ce qui concerne l'apprentissage des stratégies de nudge du système, le Chapitre 5 a présenté des algorithmes d'apprentissage par renforcement efficaces par échantillon.

Nous verrons alors comment implémenter ces méthodes dans un système de dialogue dans la section suivante et tenterons d'évaluer la faisabilité de l'ap-

prentissage en ligne de celles-ci.

Quelles métriques élaborer pour évaluer l'influence du nudge ?

Si le changement émotionnel peut aider à évaluer un système de dialogue, comme a pu le présenter le Chapitre 4, l'état de l'art sur les nudges (Chapitre 2) n'a pas révélé de métriques standard généralisables pour mesurer l'efficacité d'un nudge. En effet, leur évaluation est souvent dépendante de la tâche (slot-filling) et du contexte (interaction avec un chatbot ou un robot social). Nous introduirons donc des métriques numériques simples adaptées à une interaction orale dans nos expériences.

Comment évaluer les systèmes de nudges linguistiques ?

Les Chapitres 4 et 5 ont montré comment évaluer un système de dialogue avec apprentissage par renforcement. Nous utiliserons une modélisation similaire dans la partie suivante.

Est-ce éthique de faire des systèmes qui nudgent ?

Les principaux enjeux éthiques des nudges ont été exposés dans le Chapitre 2. En plus des contraintes de l'éthique du numérique, Nous précisons dans chaque expérience la démarche éthique suivie également. Celle-ci sera davantage discutée dans la Conclusion.

Deuxième partie

Expériences pour l'analyse de nudges linguistiques

Bien que de plus en plus de corpus de références pour le dialogue orienté tâche soient accessibles ces dernières années (MultiWoZ [Budzianowski et al., 2018], DSTCs), il n'existe pas à notre connaissance de corpus de dialogue dédiés aux nudges. La première étape de notre travail a donc été de collecter des données orales spécifiques aux nudges pour observer les facteurs utiles à leur prédiction et construire des corpus de référence adaptés. Nous présentons ensuite dans cette partie l'approche suivie pour construire notre système de dialogue capable de nudges fondé sur un apprentissage par renforcement. Si de nombreux systèmes de dialogue récents optent pour une approche *end-to-end*, une approche modulaire offre néanmoins davantage de visibilité sur les performances et les difficultés rencontrées des différents composants du système (cf section 4.2.5). Dans le cas de cette thèse cherchant à étudier les paramètres pertinents pour un agent influençant un utilisateur dans le dialogue, une architecture modulaire nous a ainsi semblé meilleure. Cela permet ainsi d'étudier l'importance et l'impact des différents modules, de les remplacer et les améliorer au fur et à mesure ainsi que de pouvoir les utiliser avant l'apprentissage pour automatiser les phases de collecte. En plus d'un corpus spécifique aux nudges, l'évaluation de notre agent nécessite la création de métriques spécifiques pour mesurer concrètement l'influence d'un nudge. En effet, la définition d'un nudge en économie étant assez large, leur évaluation est souvent très dépendante de la tâche demandée.

Nous proposons ainsi dans le Chapitre 7 deux expériences différentes conçues pour collecter des données de dialogues et mesurer numériquement le succès d'un nudge. Le Chapitre 8 décrit ensuite l'architecture entière du système choisi pour déclencher le nudge. L'émotion jouant une part centrale dans les nudges, le Chapitre 9 présente notre intégration d'un module pour la détection d'émotions dans le dialogue à partir de few-shot learning. Le Chapitre 10 montre enfin la modélisation du système comme problème d'apprentissage par renforcement.

Chapitre 7

Conception des nudges dans le dialogue

Notre travail s'est articulé autour de 2 expériences décrites ci-dessous, faisant l'objet d'une partie chacune. La première s'est faite en collaboration avec une équipe de chercheurs en économie du laboratoire du RITM et s'est déroulée sur le terrain, dans une école primaire. Elle s'intéresse en particulier aux différents facteurs (types d'examineurs, états mentaux relevés, etc.) qui permettraient de prédire un nudge. La seconde est entièrement en ligne et cherche à automatiser le processus de collecte et d'apprentissage dans une approche continue. Elle a notamment été montée en temps de Covid où il n'était pas possible de poursuivre des expériences en présentiel.

7.1 Objectif 1 : Expérience sur le terrain montrant l'efficacité des nudges des machines auprès d'enfants

Cette expérience se focalise en particulier sur les questions de recherche suivantes :

- Quels présupposés avoir sur les nudges ? Elle tente de répondre notamment aux hypothèses 1 et 4 relatives aux nudges : étudier les différences d'influence du type d'interlocuteur et déterminer les informations émotionnelles, paralinguistiques et contextuelles pertinentes pour prédire les nudges.
- Comment mesurer l'influence d'un nudge ? Quelles métriques utiliser ?

Son protocole et les résultats obtenus ont alors fait l'objet de 2 publications [[Ali Mehenni et al., 2020a](#)][[Ali Mehenni et al., 2020b](#)].

7.1.1 Préliminaires

Ne disposant pas de corpus spécifiques à l'étude du nudge, un premier enjeu pour cette thèse était de réaliser une expérience de collecte. Elle a ainsi été planifiée en collaboration avec une équipe de chercheurs en économie du RITM, appartenant également au projet DATAIA, dans le but de satisfaire un certain nombre de contraintes et d'objectifs des deux partis.

Objectifs

- *Corpus de données réelles sur le nudge* : Un premier objectif était la collecte de données notamment orales et françaises qui soient relatives aux nudges numériques. En effet, c'est le type d'entrées que recevra par la suite le SDS envisagé pour cette thèse.
- *Étude du nudge auprès d'une population sensible* : Cette expérience avait pour but également d'étudier l'influence du nudge auprès d'enfants d'école primaire, une population estimée plus facilement influençable et qui est pourtant particulièrement exposée aux nouvelles technologies.
- *Influence de l'examineur* : Un autre focus de cette expérience était de comparer l'efficacité des nudges en fonction de celui qui conduit l'expérience. A cette fin, 3 types d'examineurs ont été considérés : un robot humanoïde (Pepper de SoftBank Robotics), une enceinte (Google Home déconnecté) et un adulte.
- *Mesure numérique d'un nudge* : Les 2 équipes étaient intéressées par une mesure concrète et sans équivoque du succès d'un nudge. Le protocole choisi inclut ainsi un jeu en économie standardisé, aux résultats statistiquement bien connus : le jeu du Dictateur [[Benenson et al., 2007](#)]. Il s'agit d'un jeu très simple de mesure d'altruisme en Théorie des jeux (cf Protocole ci-dessous). Il offrait l'avantage d'être compréhensible pour de jeunes enfants, d'avoir une connotation éthique et de donner un résultat numérique.
- *Mesure de la confiance envers l'examineur* : Un autre objectif de cette expérience était de mesurer lors de l'interaction orale la confiance que l'enfant accordait à son examineur. Pour cela, différentes questions cherchaient à tester la crédulité de l'enfant et s'il était prêt à censurer des informations à l'examineur sur le comportement de ses camarades. Un des

enjeux sous-jacents de ces questions était également d'observer quelles stratégies pourraient potentiellement augmenter l'affection de l'enfant envers un robot.

Contraintes

- *Environnement* : L'expérience devait se dérouler sur le terrain, c'est-à-dire au sein de l'école primaire en quelques jours seulement. Les expériences avec chaque type d'examineurs (robot, enceinte, adulte) ont donc dû être conduites en parallèle ce qui entraînait certaines contraintes matérielles et logistiques, notamment pour obtenir des set-ups le plus possible identiques.

Cela posait certaines contraintes sur la collecte audio, en raison du bruit de l'environnement également (en particulier la récréation...).

- *Décision en temps réel* : L'interaction avec l'enfant est en temps réel. Pour conserver un échange naturel, l'examineur, s'il est une machine, doit donc être autonome ou du moins le simuler. Le système n'étant pas encore entraîné, nous avons eu recours au dispositif classique du WOZ (Wizard of Oz). L'idée était ainsi de laisser croire aux enfants qu'ils dialoguaient avec un robot ou une enceinte alors que celui ou celle-ci était en réalité contrôlé par un des chercheurs dans la salle.
- *Accessibilité* : Cette expérience s'adressant à de jeunes enfants (les plus jeunes avaient aux alentours de 5 ans), un effort de simplicité a été réalisé sur le protocole afin qu'il demeure bien compréhensible. De nombreuses étapes de confirmation pour vérifier sa compréhension ont notamment été ajoutées.
- *Éthique* : Cette expérience s'adressant à des enfants, une attention tout particulière s'est portée sur le protocole afin qu'il s'effectue dans des conditions agréables et éthiques (possibilité de faire une pause, humour, voire toucher le robot, etc.). L'expérience a notamment été validée en amont par un comité d'éthique. Elle a ensuite suivi la démarche éthique suivante : le consentement des parents a été recueilli, puis les participants ont été anonymisés et répartis aléatoirement et enfin la sauvegarde de leurs données s'est faite de manière anonyme selon des règles présentés en amont de RGPD. Une séance de sensibilisation sur les nudges et la confiance accordée aux technologies en général a également été réalisée a posteriori à l'école.

7.1.2 Protocole

Notre première expérience était une expérience sur le terrain auprès de jeunes enfants adaptée à une école primaire. Les enfants étaient répartis de manière statistiquement équilibrée, en fonction de leur âge et de leur genre, en 3 groupes. Un tiers des participants devaient ainsi interagir avec le robot social Pepper de SoftBank Robotics, un autre tiers avec une enceinte Google déconnectée et le dernier tiers avec une personne adulte (un doctorant en économie). Les interactions avec les examinateurs “machine” étaient réalisées sous WOZ. Une illustration du set-up de l’expérience est donnée Figure 7.1.

L’expérience durait en moyenne 8 minutes et étaient divisées en 2 parties :

1. La première concernait un jeu d’économie où l’enfant faisait un choix physique où l’enjeu était des billes à gagner.
2. La seconde était présentée comme un quizz relatif aux jeux vidéo, dans lequel était glissé des questions cherchant à mesurer la confiance de l’enfant envers l’examineur. Elle cherchait notamment à identifier les facteurs pertinents dans l’oral qui aideraient à prédire les résultats de nudges de la partie précédente.



FIGURE 7.1 : Enfant participant à l’expérience avec le robot Pepper

Mesure numérique des nudges avec le jeu d’économie du Dictateur

La première étape consistait en un jeu d’économie mesurant l’altruisme le jeu du Dictateur, adapté à des enfants d’école primaire [Benenson et al., 2007]. Le

Le jeu du Dictateur initial est réalisé pour des adultes avec de l'argent. L'expérience consiste alors à donner une somme d'argent au participant puis de voir quelle proportion il est prêt à redonner à une personne inconnue par altruisme. Par souci d'éthique, l'argent a été remplacé dans cette expérience par des billes que les enfants pouvaient gagner pour conserver un enjeu similaire. (En réalité, l'ensemble des billes a été distribué en totalité à chaque enfant à la fin de l'expérience.)

L'intérêt de cette expérience pour nous était d'introduire un nudge entre chaque choix de l'enfant et d'observer à quel point il était influencé. Le design de l'expérience est illustré dans la figure 7.2.

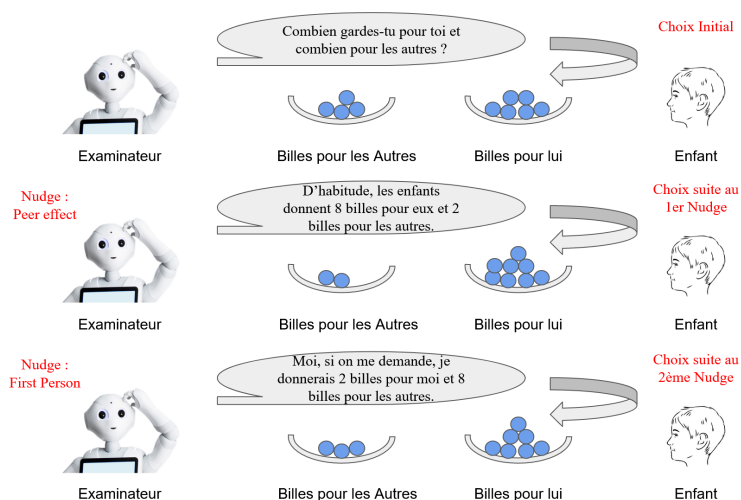


FIGURE 7.2 : Exemple de déroulé de l'expérience du Dictateur

Chaque enfant recevait initialement 10 billes qu'on lui demandait alors de répartir entre ce qu'il souhaiterait garder pour lui et ce qu'il souhaiterait donner à une autre personne inconnue. Une fois son choix fait, son interlocuteur le soumettait successivement à 2 nudges (dans un ordre aléatoire) en lui offrant la possibilité à chaque fois de changer son dernier choix.

Pour cette expérience, nous avons utilisé des stratégies de *normes sociales* (ou influence sociale selon les 6 catégories utilisées dans [Caraban et al., 2019], donc transparentes et invitant à un choix réfléchi. Les 2 nudges en question consistaient à ajouter un contexte au jeu en précisant soit ce qu'aurait choisi l'expérimentateur (robot/enceinte/humain), ce que nous avons appelé nudge **première personne**, soit ce que choisissent en général les gens de son âge, c'est-à-dire un nudge **peer effect**.

Le nudge *peer effect* était ainsi de la forme suivante :

D'habitude, les enfants gardent X billes pour eux et donnent Y billes pour les autres.

Les valeurs de X et Y étaient décidées en temps réel en fonction du choix précédent de l'enfant et d'une valeur seuil de 3 déterminée selon la littérature. En effet, selon une étude du jeu du Dictateur sur des enfants britanniques d'une école primaire (avec des gommettes en jeu) [Benenson et al., 2007], les enfants avaient en moyenne une mesure d'altruisme de 30%.

Ayant fixé cette valeur seuil, les valeurs de X et Y étaient alors déterminées ainsi :

- Si l'enfant avait choisi 3 billes ou plus pour les autres (valeur seuil), alors X et Y prenaient respectivement les valeurs de 8 et 2. L'examineur incitait l'enfant à prendre plus de billes pour lui.
- Si l'enfant avait choisi moins de 3 billes pour les autres, alors X et Y prenaient les valeurs de 2 et 8. L'examineur incitait l'enfant à être plus altruiste.

L'enfant était alors considéré comme nudgé à chaque fois qu'il modifiait son précédent choix. En comparant les valeurs des différentes réponses, nous avons alors construit une métrique simple pour mesurer quantitativement l'influence de ces biais d'ancrage et sociaux (cf section 7.1.4).

Interaction orale cherchant à identifier les facteurs prédictifs des nudges

La deuxième partie était une interaction orale, présentée comme un quizz relatif aux jeux vidéo, permettant de tester l'influence de différentes questions et leur lien potentiel avec les résultats de nudge obtenus durant le jeu du Dictateur. Les questions étaient glissées en alternance avec les questions du quizz et séparées par plusieurs tours de paroles.

L'objectif de ces questions était d'estimer la confiance de l'enfant envers l'examineur (à travers sa crédulité et son degré de censure envers celui-ci), son attention et sa mémoire.

Les deux questions d'influence ayant eu le plus de réactions étaient :

- une question *incongrue* : "Savais-tu que la lune a le goût de gruyère ?"
- une question de *censure/confiance* : "Est-ce que tu as déjà vu des camarades se battre ou se disputer pour des billes ?"

Cette partie a ainsi permis de chercher des indices paralinguistiques dans le dialogue qui aideraient à prédire un nudge. Les résultats de ces deux parties ont ainsi fait l'objet de 2 articles [Ali Mehenni et al., 2020a][Ali Mehenni et al., 2020b] et les conclusions de cette expérience sont discutées section 7.1.4.

7.1.3 Corpus

L'expérience a été conduite auprès de 91 enfants d'une école primaire âgés de 5 à 10 ans. Les différentes statistiques sont données dans la Table 7.1 suivante.

Nombre d'enfants	91
Nombre d'enfants garçons	53
Nombre d'enfants filles	38
Nombre d'enfants avec le Robot	29
Nombre d'enfants avec l'enceinte	31
Nombre d'enfants avec l'adulte	31
Age des enfants	5-10
Longueur moyenne des dialogues	8min10s
Durée totale du corpus	12h

TABLE 7.1 : Données du corpus collecté

7.1.4 Résultats

Influence plus forte des machine lors du jeu du Dictateur

Comme le révèle la Figure 7.3 sur les résultats du jeu du Dictateur, les nudges se sont avérés efficaces puisque chaque incitation utilisée pousse en moyenne **un enfant sur deux** à modifier son choix (51% des enfants modifient leur choix à la suite du premier nudge et 47% à la suite du second). Nous avons pu remarquer cependant que 30% des enfants ne sont jamais nudgés dans le sens où ils ne changent jamais de choix.

De plus, les enfants semblent davantage influencés par les examinateurs "non-humains". En effet, parmi les enfants qui ont changé leur choix lors du second nudge, environ 16% interagissait avec l'adulte tandis que 40% et 44% respectivement interagissait avec le robot et l'enceinte (cf Table 7.2 et figure 7.3). Ce résultat semble conforter notre Hypothèse 1 selon laquelle un **interlocuteur machine** est plus susceptible d'influencer à travers des nudges. Néanmoins, afin de généraliser cette observation, des données supplémentaires seraient nécessaires et devraient couvrir un plus grand nombre d'enfants, des

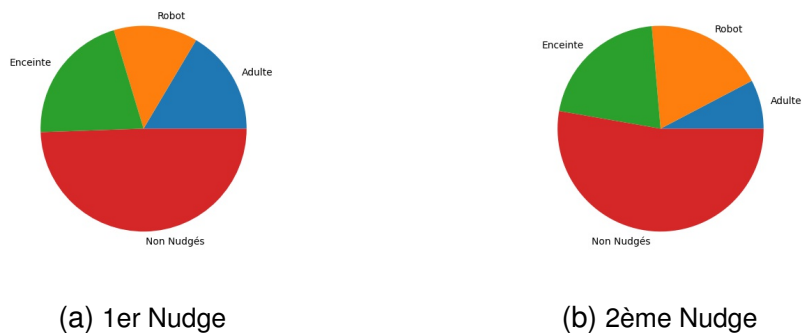


FIGURE 7.3 : Proportion des enfants ayant changé leur choix pour chaque examinateur suite à chaque Nudge.

groupes d'enfants plus variés ainsi qu'une population cible plus âgée (adultes, personnes âgées).

En termes d'altruisme cependant, l'expérience révèle peu de différences entre les différents expérimentateurs, même si la moyenne des billes données par altruisme, de 4.57, est un peu plus élevée que ce que rapporte la littérature [Benenson et al., 2007].

Métriques :

Pour comparer plus précisément l'influence de chaque nudge, nous avons également introduit une métrique de nudge, définit ainsi :

$$\text{Métrique de nudge} = |R_0 - R_n| - |R_1 - R_n|$$

où R_0 et R_1 sont les résultats donnés par l'enfant respectivement avant et après le nudge et R_n est la valeur suggérée par le nudge.

Le raisonnement derrière cette métrique est de mesurer numériquement si l'enfant s'est rapproché de la valeur suggérée par le nudge lors du choix suivant. Le signe de cette métrique indique notamment si l'enfant est allé dans le sens du nudge (valeur positive) ou dans la direction opposée suggérée par le nudge (valeur négative). La valeur de cette métrique est équivalente, quant à elle, au nombre de billes de différence entre les 2 choix de l'enfant.

Néanmoins, celle-ci ne permet pas de savoir dans quelle proportion l'enfant s'est rapproché de la valeur suggérée. En effet, en fonction de l'écart entre la valeur suggérée et la valeur initiale, le nombre de billes changées n'a pas la même signification. Pour un même nombre de billes changées, l'enfant a-t-il

Groupes	Adulte	Robot	Enceinte	Total
Nombre d'enfants	31	29	31	91
Enfants nudgés (1er Nudge)	48%	41%	61%	51%
Enfants nudgés (2e Nudge)	23%	59%	61%	47%
Métrique de nudge (1er Nudge)	0.68	1.10	1.26	1.01
Métrique de nudge normalisée (1er Nudge)	0.12	0.30	0.26	0.23
Métrique de nudge (2ème Nudge)	0.97	1.90	1.48	1.44
Métrique de nudge normalisée (2ème Nudge)	0.19	0.43	0.46	0.35

TABLE 7.2 : Données collectées durant le jeu du Dictateur

respecté à la lettre la valeur suggérée ou s'en est-il seulement approché ?

Pour prendre en compte cet aspect, la métrique du dernier nudge a été divisée par la différence en valeur absolue entre le résultat avant le nudge et la valeur suggérée par le nudge :

$$\text{Métrique de nudge normalisée} = \frac{\text{Métrique de nudge}}{|R_0 - R_n|}$$

Selon la table 7.2, la première métrique indique que les enfants changent en moyenne 1 bille dans la direction suggérée par l'examineur. Les enfants réagissent donc positivement aux nudges.

Les 2 métriques montrent en outre que l'influence des interlocuteurs non-humains (robot et enceinte) est également plus forte que celle de l'interlocuteur humain. Cela signifie que lorsque l'enfant est influencé, il a tendance à suivre plus fidèlement le conseil du robot ou de l'enceinte que celui donné par l'adulte.

Indices paralinguistiques pertinents pour prédire un nudge

L'expérience finie, les enregistrements ont été annotés à l'intérieur des tours de parole du locuteur selon plusieurs dimensions à l'aide du logiciel d'annotations multimodales ELAN¹ (cf figure 7.4)[Ali Mehenni et al., 2020a]. En plus d'une transcription, les 3 dimensions principales étaient : l'état mental estimé de l'enfant (émotions, attention, etc.), les affect bursts rencontrés ainsi que le registre de langage utilisé par l'enfant. L'objectif de ces annotations est d'une part, de pouvoir entraîner différents types de classifieurs utiles pour estimer l'état du dialogue, d'autre part, de chercher à relier certaines caractéristiques paralinguistiques aux nudges.

Etats mentaux :

Nous regroupons dans les "états mentaux" les émotions et autres états de l'enfant comme l'attention, la confiance ou la confusion qui sont présents dans les tours de paroles et qui permettraient d'estimer l'état actuel de l'enfant et sa susceptibilité d'être nudgé.

En ce qui concerne les émotions, la roue des émotions de [Plutchik, 1984] a servi de référence pour sélectionner les 6 émotions suivantes : l'amusement, le respect, la surprise, l'irritation, la nervosité et l'intimidation (les deux dernières se rattachant à deux niveaux faibles de peur sur la roue). En outre, des macro-étiquettes ont également été ajoutées pour séparer les émotions positives des émotions négatives, afin de pouvoir faciliter la tâche de classification étant donné le peu de données pour certaines classes.

¹<https://archive.mpi.nl/tla/elan>

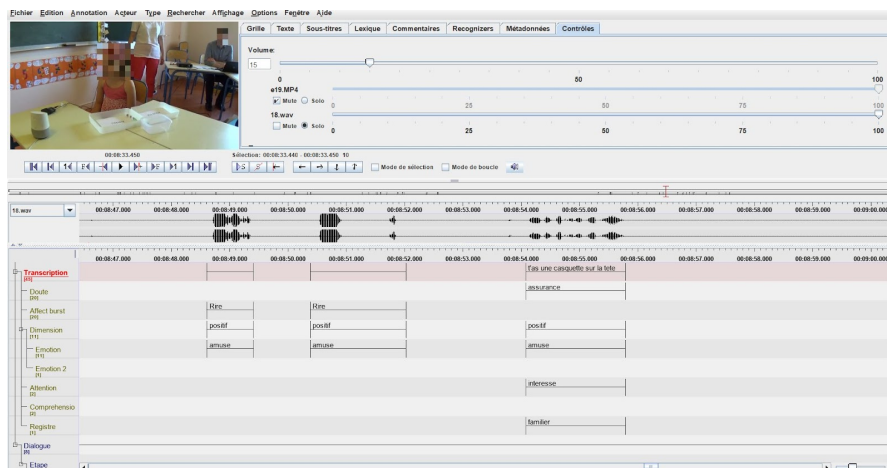


FIGURE 7.4 : Annotations des états de l'utilisateur (états mentaux, affect burst, registre de langue) avec Elan

L'attention ou l'intérêt étant un concept clé pour les nudges, des étiquettes d'annotation "Intérêt" et "Désintérêt" ont également été ajoutées. Dans la même optique, nous avons choisi les couples d'étiquettes "Confiance"/"Doute" et "Compréhension"/"Confusion".

Affect Bursts :

Les affect bursts suivants ont été annotés : "Rire", "Hésitation" et "Souffle". Néanmoins, le nombre d'annotations pour ceux-ci, en particulier le "Souffle", était assez faible.

Registre de langue :

Un autre indice qui nous a semblé pertinent pour prédire si l'enfant était susceptible de se faire influencer est le registre de langage. En effet, celui-ci peut refléter la proximité de l'enfant avec son examinateur. Il correspond aux étiquettes "Poli"/"Familier".

Enfin, une classe "Autre" a été construite contenant principalement des chunks neutres et ayant pour but de servir de classe de référence pour entraîner des classifieurs. La liste complète des étiquettes attribuées avec leur nombre de chunks de parole associés est donnée dans la table 7.3.

caractéristiques objectives au niveau du tour de parole :

En plus des états mentaux collectés, par nature subjectifs, des mesures ob-

Classes annotées	Nombre
Emotions Positives (amusé, surpris, respectueux)	354
Emotions Négatives (irrité, nerveux, intimidé)	59
Assurance	622
Doute	286
Intérêt	330
Désintérêt	49
Compréhension	68
Incompréhension	101
Poli	49
Familier	117
Hésitation	273
Rire	52

TABLE 7.3 : Annotations des états mentaux, affect bursts et registres de langue en classes et nombre de segments audio par classe (sur 91 dialogues)

jectives ont été calculées pour chaque tour de parole, telles que la durée de latence (le temps entre la fin de la question de l'interlocuteur et le début de la réponse de l'enfant), la durée du tour de parole, la durée et le nombre de pauses silencieuses ainsi que le nombre d'hésitations par tour de parole. Ces durées et caractéristiques paralinguistiques sont décrites dans la Table 7.4.

Le but de ces mesures est de mettre en évidence certaines tendances en termes de réactions aux nudges. En effet, la Table 7.4 montre que la durée de latence moyenne est plus longue lorsque l'examineur est non humain. Cela pourrait être lié au fait que les échanges avec le robot et l'enceinte étaient parfois moins réactifs et plus lents que la parole spontanée de l'adulte. (Le nombre moyen de tours de parole est également moins élevé pour l'adulte car quelques questions supplémentaires sur les impressions de l'enfant sur son examineur avaient été ajoutées à la fin de l'interaction pour l'enceinte et le robot.)

Prédiction des nudges :

Pour estimer le pouvoir prédictif des différents paramètres de chaque dialogue sur le succès d'un nudge, un vecteur pour chaque tour de parole a été

Caractéristiques (valeurs moyennes)	robot	enceinte	adulte
Nombre de tours	29.31	30	22.7
Durée d'un tours (s)	2.23	3.17	2.84
Durée de latence (s)	1.41	1.51	0.66
Nombre de pauses par tour	0.17	0.26	0.12
Durée d'une pause (s)	2.03	2.09	1.39
Nombre d'hésitations par tour	0.17	0.21	0.22
Durée d'une hésitation (s)	1.26	0.83	0.89

TABLE 7.4 : Caractéristiques au niveau du tour de paroles (valeurs moyennes, durées en s)

construit, rassemblant l'ensemble des caractéristiques considérées, dont les différentes catégories sont décrites dans la Table 7.5. Les réactions de l'enfant aux 2 questions d'influence ont également été annotés comme suit :

- de manière binaire pour la question *incongrue* en fonction de l'accord ou le désaccord exprimé de l'enfant avec le fait ridicule proposé ("la lune a le même goût que le fromage")
- "0", "1" ou "2" pour la question de *confiance* si, respectivement, l'enfant n'avait pas coopéré, avait seulement décrit les conflits en général ou avait décrit les conflits en donnant des noms.

Une base de données de près de 2500 vecteurs représentant les tours de parole obtenus à partir des 91 dialogues a ainsi été constituée dans le but de servir d'entrée à des modèles d'apprentissage automatique. Ces vecteurs n'étant pas indépendants, les vecteurs d'un même dialogue ont donc tous été groupés soit dans la base de données d'entraînement, soit dans celle de validation, soit dans celle de test. Ils ne prennent néanmoins pas en compte l'aspect séquentiel du dialogue et les dépendances qui pourraient en résulter. Ayant à l'esprit de conserver une base de données de taille suffisante, nous avons fait l'hypothèse que ces biais étaient négligeables pour une première analyse du lien entre ces caractéristiques et l'efficacité d'un nudge.

Afin de prédire le succès des nudges utilisés dans le jeu du Dictateur, différentes combinaisons de caractéristiques ont donc été envisagées. L'objectif

Catégories	Caractéristiques
Profil	âge, genre
Interlocuteur	adulte, enceinte, robot
Tour de parole	durée, latence
Annotations émotionnelles (perceptives)	doute/assurance, agressif/intimidé/nerveux, amusé/surpris/respectueux,intéressé/désintéressé, compréhension/incompréhension
Paralinguistique calculée (objectif)	nombre et durée des hésitations

TABLE 7.5 : Catégorisation des caractéristiques utilisées

était de déterminer les caractéristiques qui contiennent les informations les plus pertinentes sur le comportement des enfants et de les sélectionner comme entrée pour un classifieur binaire cherchant à prédire si l'enfant avait été nudgé ou non.

Trois sets ou ensemble de caractéristiques ont alors été sélectionnés. Ils correspondent aux caractéristiques les plus fréquemment observées dans le jeu de données et comprennent à la fois des caractéristiques perceptives et des caractéristiques mesurées objectivement :

Sets :

1. Caractéristiques les plus fréquentes calculées et/ou extraites des annotations : durée, durée de latence, doute, compréhension, dimension, émotion, attention
2. Set regroupant les 2 caractéristiques les plus importantes du Set 1 et les caractéristiques personnelles et situationnelles : durée, durée de latence, sexe, âge, interlocuteur
3. Set regroupant les 2 caractéristiques les plus importantes calculées à partir d'annotations au niveau des tours de parole (durée, durée de latence), le sexe, l'âge, l'interlocuteur et la réaction à d'autres stratégies d'influence (question incongrue, question ouverte).

Importance des caractéristiques selon un modèle Random Forest :

Pour cette expérience, le modèle de classification Random Forest (de scikit-learn, avec 100 classifieurs arbres de décision) avait été utilisé car il fournit des informations sur l'importance de chaque feature sur la prédiction.

Il s'agit d'une méthode d'apprentissage ensembliste, c'est-à-dire utilisant un grand nombre d'algorithmes d'apprentissage peu corrélés, ici des arbres de décision, pour obtenir une prédiction meilleure que celles des algorithmes pris séparément.

Les arbres de décision sont une méthode d'apprentissage supervisée non paramétrique permettant d'effectuer une régression ou classification. Dans le cas d'un classifieur classique, ils suivent un schéma en forme d'arbre (succession de nœuds jusqu'à des feuilles terminales) cherchant à prédire une classe cible. Pour cela, ils séparent à chaque nœud les exemples de la base d'apprentissage selon les différentes valeurs prises d'une feature jusqu'à représenter tous les cas de décision pertinents pour pouvoir classer ces exemples. La feature la plus informative est choisie en priorité pour chaque nœud au fur et à mesure (calculée selon son gain d'information, aussi appelée information mutuelle, similaire au concept d'entropie en théorie de l'information)(cf équation 7.1). En pratique, il s'agit de la feature qui sépare les différents exemples de manière la plus équitable possible dans chacune de ses branches, permettant ainsi de réduire les différents cas le plus vite possible.

Le *Gain d'Information* $GI(Y, X1)$ correspond alors à la réduction en entropie de la variable cible Y obtenue par la connaissance de la valeur de la variable $X1$:

$$GI(Y, X1) = H(Y) - H(Y|X1) \quad (7.1)$$

où H est la fonction d'entropie définie par $H(Y) = \sum_{i=1}^n -p_i * \log_2(p_i)$ avec p_1, p_2, \dots, p_n qui représentent les proportions de chacune des n classes qui seront séparées par le noeud (p_1, \dots, p_n sommant à 1)

Le gain d'information peut notamment être représenté par la divergence de Kullback-Leibler.

Un arbre de décision offre ainsi une bonne interprétabilité mais se révèle très sensible au changement de données. Le modèle Random Forest garantit alors une faible corrélation entre les différents classifieurs arbres de décision grâce à la technique du *bagging* : chaque arbre de décision est entraîné en sélectionnant de manière aléatoire (et avec remise) des exemples de la base de données, ce qui assure d'obtenir des arbres variés. L'algorithme Random Forest classe alors de nouvelles données en prenant la classe qui aura reçu

	1er Nudge	2ème Nudge
Set 1	0.53(\pm 0.05)	0.55(\pm 0.05)
Set 2	0.67(\pm 0.05)	0.67(\pm 0.05)
Set 3	0.85(\pm 0.04)	0.83(\pm 0.03)

TABLE 7.6 : Résultats de classification : score f1 obtenus par validation croisée (10-fold) avec les sets 1, 2 et 3

	Caractéristiques	1er Nudge	2ème Nudge
set 1	latence	44	45
	durée	41	38
set 2	latence	38	36
	durée	35	33
	interlocuteur	12	16
set 3	latence	20	20
	durée	19	20
	q. ouverte	20	19
	interlocuteur	10	15
	q. incongrue	10	6

TABLE 7.7 : Caractéristiques importantes en % pour chaque set et chaque nudge

le plus de votes de ses différents arbres de décision.

La Table 7.6 montrent les résultats de f1-score moyennés sur 20 runs de ce classifieur en fonction des différents sets utilisés avec validation croisée (10-fold). La Table 7.7, quant-à-elle, met en évidence les caractéristiques les plus importantes pour la prédiction retenues pour chaque set ("q." signifiant question).

Les résultats soulignent que les caractéristiques extraites au niveau du tour de parole (configuration du set 1) ne sont pas suffisantes pour prédire les résultats de nudges du jeu du Dictateur (f1-score et précision également à peine supérieure au hasard). De plus, comme le montre la Table 5, seules deux caractéristiques semblent réellement pertinentes : la durée et la durée de latence

des tours de parole.

En gardant ces deux caractéristiques et en ajoutant les caractéristiques personnelles et situationnelles (interlocuteur, sexe et âge de l'enfant) (configuration set 2), nous atteignons un score f1 d'environ 0.67 pour la prédiction des deux stratégies de nudging (Nudges 1 et 2). Enfin, en ajoutant les deux questions d'influence (incongrue et ouverte) pour construire le set 3, les résultats de classification montrent une bonne prédiction des réactions des enfants avec un score f1 de 0.85 et 0.83 pour le premier et second Nudge respectivement. Ceci semble indiquer une certaine corrélation entre les réponses aux deux questions d'influence et la propension à être nudgé.

Conclusion

En conclusion, nous pouvons retenir les points suivants :

- Une classification fondée sur les tours de parole comme unités d'analyse et ne prenant en compte que des mesures quantifiables et des annotations émotionnelles apparaît ici comme une tâche difficile, même si certains paramètres tels que la durée et la latence du tour de parole semblent être des caractéristiques pertinentes.
- Les caractéristiques personnelles (âge, sexe) et situationnelles (type d'interlocuteur) ainsi que les réponses aux deux questions d'influence semblent être des caractéristiques robustes pour prédire les réactions aux stratégies de nudge, car leur inclusion permet d'obtenir les meilleurs résultats.
- Les résultats ne mettent pas en évidence de différence notable entre le premier et le second nudge du jeu du dictateur indiquant que ces caractéristiques semblent être pertinentes indépendamment de l'ordre des nudges a priori.
- Bien que les états émotionnels puissent être considérés comme de bons prédicteurs des réactions par rapport aux nudges, leur impact a été assez faible ici. Une raison à cela vient peut-être de leur caractère épars dans les données, soulignant ainsi le besoin de corpus plus importants pour soutenir les résultats actuels. En outre, les résultats suggèrent qu'une analyse approfondie des caractéristiques correspondant à certains sous-ensemble spécifiques de tours de parole (par exemple, les tours de parole au niveau des questions d'influence) pourrait fournir des indices plus pertinents que des vecteurs sur tout le dialogue.

Étant donné la taille du corpus, ces résultats correspondent davantage à des

tendances mais fournissent des pistes intéressantes sur le choix de caractéristiques dans l'étude des nudges.

7.2 Objectif 2 : Expérience automatisée en continue auprès d'internautes montrant l'influence de stratégies de nudges linguistiques oraux, réalisée sur la plateforme du système de dialogue développé

Cette expérience cherche à répondre aux hypothèses de recherche suivantes :

- Comment créer un environnement de recherche pour tester les nudges ? Quelle tâche choisir pour évaluer les nudges oraux dans un système de dialogue ?
- L'Hypothèse 2 sur les nudges : un nudge oral est-il significativement plus efficace qu'une question simplement répétée ?
- Quel est l'impact de la représentation de l'agent sur les nudges ? Quelles différences observer entre un agent robot physique et imaginaire ?

7.2.1 Préliminaires

Pour enrichir la première collecte de données et pour obtenir une plateforme sur laquelle déployer le système de dialogue final, un site a également été développé à partir de zéro (serveur Apache, base de données, etc.). Cette expérience a été montée en temps de Covid où il n'était pas possible de poursuivre des expériences en présentiel. Le site déployé a pour but de recueillir d'autres profils plus nombreux et plus variés et ce, à travers une approche automatique pouvant être conduite dans la durée. C'est sur ce site également que sera déployé notre système de dialogue avec apprentissage par renforcement ce qui permettra de tester un apprentissage en ligne.

Les objectifs et contraintes de cette 2ème expérience sont développés ci-dessous.

Objectifs

- *Données réelles d'interaction sur les nudges* : Cette expérience s'adresse à une population différente, principalement adulte, dans un environnement

non contrôlé. Un objectif associé est ainsi de recueillir des profils plus variés, dans un environnement plus proche du cadre d'utilisation privé des agents conversationnels. En effet, [Berry et al. \[2009\]](#) ont observé une différence de comportements lorsque les participants étaient dans un laboratoire et lorsqu'ils étaient dans un cadre domestique. Cette expérience permettra alors de récolter des interactions plus spontanées (sans le stress de l'équipement et du regard des chercheurs en présentiel) et dans un cadre réaliste (bruits ambiants d'un foyer, distractions, voire même conversations parallèles parfois, etc.). C'était également l'occasion de tester différents nudges audio et leurs réponses spécifiques.

- *Comparer l'efficacité de nudges par rapport à l'absence de nudges* : Un objectif important du nouveau protocole est de comparer l'efficacité d'une question reformulée avec un nudge par rapport à une simple répétition de la question sans nudge. Les réponses à une simple répétition serviront ainsi de référence pour mesurer l'efficacité de stratégies de nudges davantage adaptées à un individu.
- *Continuité* : Un avantage de cette expérience est qu'elle peut être maintenue sur une longue durée, permettant alors de recueillir davantage de participants.
- *Autonomie* : Cette expérience a pour but d'être le plus autonome possible et de faciliter ainsi les annotations et traitements postérieurs. A cette fin, les données doivent donc être annotées et traitées automatiquement en temps réel et les étapes du protocole entièrement déterminées.

Les modules ASR et NLU doivent ainsi être fonctionnels. Pour cette expérience, la reconnaissance vocale de Google pour l'ASR a été choisie permettant une transcription rapide sur laquelle un traitement NLP peut être appliqué. Ces 2 modules et leurs choix associés sont détaillés au Chapitre suivant. Cet impératif de fonctionnalité constitue notamment une des raisons justifiant le choix d'une architecture modulaire : certains modules peuvent être utilisés à différentes étapes de la construction du système de dialogue.

- *Représentation de l'agent* : Afin de comparer l'agent de cette expérience aux différents expérimentateurs machines de la première expérience, celui-ci est représenté de manière figurative par une image du robot Pepper

(utilisé dans l'expérience 1). L'idée est ainsi de suggérer un robot imaginaire. L'agent peut ainsi être assimilé à une enceinte vocale mais cette fois-ci sous l'apparence d'un robot relativement anthropomorphe.

Contraintes

- *Environnement* : La collecte de données orales à distance fait face à un certain nombre de challenges relatifs aux aléas de l'environnement d'un foyer. Tout d'abord, le cadre de l'étude pouvant ne pas être isolé, l'utilisateur peut être sujet à des distractions ou bruits extérieurs. Le matériel utilisé pour l'expérience est également non contrôlé. La qualité de l'enregistrement peut donc être moins bonne que lorsque l'expérience est réalisée sur le terrain : non réglage du micro, bruits extérieurs, supports technologiques (téléphone, ordinateur, etc.) non complètement maîtrisés, connexion internet mauvaise ou interrompue, etc. L'utilisateur a aussi tendance à agir différemment lorsqu'il est dans le confort de son foyer, pouvant résulter en un engagement plus faible pour l'expérience. [Ritschel \[2022\]](#) note enfin que même en présence d'un problème stationnaire, les comportements et les réactions d'un utilisateur peuvent ne pas toujours être cohérents dans la durée, notamment dans des environnements non contrôlés, l'utilisateur pouvant être distrait ou influencé par une humeur antérieure.

- *Automatismes* : Contrairement à une expérience sur le terrain, cette expérience ne permet pas de flexibilité humaine pour résoudre un problème imprévu. La plupart de ces problèmes doivent donc être pensés en amont (incompréhensions de l'utilisateur, expérience interrompue au milieu, etc.) et être automatiquement résolus.

Néanmoins, il faut accepter une perte de qualité sur certaines données comparé au cadre contrôlé d'un WOZ sur le terrain. [Berry et al. \[2009\]](#) discute notamment du défi que représente la conception d'une évaluation et collecte de données adéquates sans un système pleinement fonctionnel déployé. Selon eux, la facilité d'adoption du système par les utilisateurs déterminera grandement le succès ou l'échec de la stratégie d'évaluation déployée.

- *Traitement et apprentissage en ligne* : Afin de permettre un traitement en temps réel des données, le site a été développé sur un GPU du laboratoire (carte graphique : Nvidia GeForce GTX 1080 Ti). La vitesse des

traitements des différents modèles doit donc prendre en compte les capacités de la carte graphique utilisée.

- *Fluidité* : L'expérience étant entièrement en ligne, elle a besoin d'une certaine fluidité et ergonomie pour maintenir l'utilisateur impliqué. En effet, les participants n'étant pas payés et simplement volontaires, certains peuvent abandonner l'expérience en cours de route. Afin de maintenir ceux-ci engagés, des efforts d'ergonomie ainsi qu'un design intuitif ont été considérés. Cette contrainte se traduit également dans le protocole : l'expérience a été "gamifiée" en proposant un score à l'utilisateur à l'issue de l'interaction.

Les traitements et modèles utilisés ont également essayé de prendre en compte le temps de chargement pour l'utilisateur. Néanmoins, si le temps de traitement des modèles sur le GPU peut être borné, l'envoi et la réception des données entre le serveur et l'utilisateur restent dépendants de la connexion internet de l'utilisateur. L'expérience est cependant conçue pour se dérouler correctement avec une connexion raisonnable, les données échangées étant relativement petites (fichier audio wav de moins de 15 secondes à chaque tour de parole).

- *Respect des données* : Les utilisateurs du site s'engagent à prendre connaissance de la politique de données du site en accord avec la RGPD. Un document précise notamment quelles données sont collectées, dans quel contexte et comment joindre l'équipe si la personne veut se retirer de l'expérience.

7.2.2 Protocole

Pour cette collecte automatique, nous avons choisi de nudger les participants sur leur comportement écologique.

Profil de l'utilisateur

L'expérience débute par une première phase d'inscription durant laquelle l'utilisateur renseigne quelques informations générales sur son profil (âge, genre, familiarité avec la technologie, végétarien). Ces informations sont stockées une seule fois dans la base de données mise en place pour le site. Cela permet ainsi de récupérer ces informations relativement statiques du profil pour d'autres expériences.

L'utilisateur est ensuite invité, à 2 reprises, à jouer quelques émotions ce qui permet d'obtenir des états mentaux individuels et déjà annotés. La première fois, les émotions à simuler sont entièrement scriptées tandis que la seconde fois, il est libre d'exprimer l'émotion demandée comme il le souhaite.

Orientation du comportement écologique

L'utilisateur est alors amené à répondre aux questions du chatbot oral sur son comportement écologique. Ce dernier va ensuite tenter de lui faire changer ses réponses initiales en reformulant certaines questions sous forme de nudges. Les questions portent en général sur l'effort que serait prêt à faire l'utilisateur pour un comportement écologique responsable (tri des déchets, alimentation, voyage, etc.) et pour lequel il s'auto-évalue. Pour pouvoir évaluer numériquement et de manière claire le succès d'une stratégie de nudge, l'utilisateur va être invité à s'attribuer une note de 1 à 5 pour chaque question, note qu'il sera amené à justifier. Suivant les principes des échelles de Likert, les notes suggérées sont en nombre impair, afin de laisser une valeur médiane. Les questions sont par exemple de la forme suivante :

A quel point serais-tu prêt à consommer moins de viande ou de produits laitiers ?

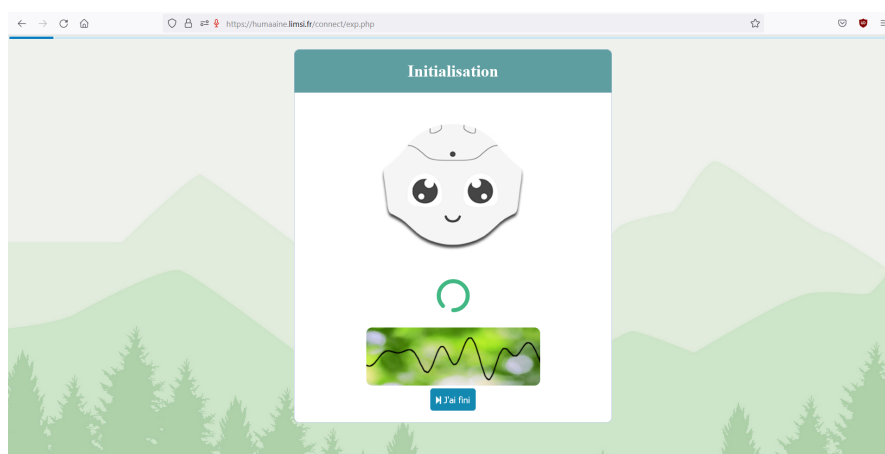


FIGURE 7.5 : Visualisation du site en train d'enregistrer la réponse de l'utilisateur

De manière similaire à la première expérience en école, les questions de nudges vont chercher à orienter le choix de l'utilisateur dans la direction opposée à celle qu'il avait précédemment donnée. Par exemple, s'il s'attribue une note faible (inférieure ou égale au seuil fixé) pour le fait de manger moins de viande, la stratégie de nudge sélectionnée l'incitera au comportement inverse :

faire davantage d'efforts pour consommer moins de viande. Pour la première expérience, le choix du seuil avait été guidé par les résultats de la littérature sur le jeu d'économie sélectionné. Dans celle-ci, le seuil a été déterminé à la valeur médiane de 3.

Pendant cette phase de collecte, l'enchaînement des questions de nudges est systématique. L'utilisateur est soumis à jusqu'à 2 relances sur chaque question de nudge, chaque relance testant une stratégie de nudges différente (sélectionnée aléatoirement).

Stratégies de nudge

Au niveau du tour de parole

Prenant en compte les résultats de la littérature (cf section 2.3.4), nous avons sélectionné 4 stratégies de nudge principales, utilisées pour reformuler les questions d'écologie et tenter d'influencer la réponse de l'utilisateur. Ces stratégies constituent une forme d'anchoring car elles viennent "ancrer" une question d'écologie dans un contexte particulier en attirant l'attention sur un point voulu.. Elles sont toutes plutôt "transparentes" et cherchent à inciter un choix "réfléchi" de l'utilisateur (quadrant droit haut selon la classification de [Hansen and Jespersen \[2013\]](#) :

- *peer-effect* (norme social) : Le comportement à nudger est présenté comme le choix de la majorité.
- *empathie* (émotionnel positif) : Le nudge exprime une émotion positive associée au comportement à nudger.
- *avertissement* (rationnel négatif) : Le nudge présente un risque associé au comportement précédemment choisi par l'utilisateur et indique comment le comportement à nudger permet d'éviter ce risque.
- *ikea effect* (personnel, intentionnel) : L'utilisateur est invité lui-même à trouver des arguments en faveur du comportement à nudger.

Une pause est également placée au milieu de l'interaction (deux moments différents sont testés) afin d'introduire aussi une stratégie de "distraction". La pause n'est pas une question de nudge en elle-même et aborde volontairement un sujet complètement différent de l'écologie. Elle consiste soit en une blague, soit en différentes questions sur les impressions de l'utilisateur à propos de son interlocuteur.

Au niveau du dialogue

En plus de ces stratégies de nudges au niveau du tour de parole, l'utilisateur est invité à s'autoévaluer sur son niveau d'écologie au début de l'expérience, puis à s'autoévaluer à nouveau à la fin, ce que nous appellerons la question d'évaluation globale. Son potentiel changement d'avis à ce sujet prend donc en compte l'ensemble des stratégies et actions du système durant l'interaction.

Le design de l'expérience fait également intervenir les stratégies suivantes :

- *gamification* : Le protocole en lui-même est présenté comme une interaction à l'issue de laquelle le participant obtient un score de "lutte pour la planète" qu'il peut ensuite comparer aux autres participants. L'idée est ainsi de gamifier l'expérience et de le motiver à finir.
- *repetition* : La plupart des questions d'écologie sont posées plusieurs fois pour observer l'impact du nudge utilisé. Néanmoins, l'aspect répétitif du protocole peut avoir un impact en lui-même, ce que nous investiguons plus précisément grâce à la création de 2 traitements différents.

Traitements

Afin de comparer l'efficacité des stratégies de nudge choisies par rapport à une simple répétition de la question sans nudge, 1 quart des participants reçoit un traitement différent des autres, appelé Traitement B :

- *Traitement A* : 5 questions de nudges sur l'écologie avec jusqu'à 2 relances par question (chaque relance testant une stratégie différente)
- *Traitement B* : Même traitement que le Traitement A, à l'exception que les 2 premières questions de nudges sont remplacées par une simple répétition de la question initiale (sans relance). En plaçant ainsi cette différence de traitement au début de l'interaction, dans la continuité des questions initiales, nous tentons de limiter la présence de biais extérieurs.

Prises de sons

La prise de son est un élément délicat à automatiser, d'autant plus lorsque le set-up ne peut pas être vérifié et se déroule à distance. Dans un environnement contrôlé comme un WoZ, l'expérimentateur est maître du matériel et peut s'adapter aux différents imprévus rencontrés (distractions, etc.). Dans le cas de

cette expérience en milieu naturel, certains choix d'enregistrements ont donc été faits.

Le temps pour chaque tour de parole est ainsi borné en fonction de la longueur de la réponse attendue (de 8 à 15 secondes). Cette longueur d'enregistrement permet d'appliquer des traitements rapides ensuite en temps réel sur la réponse de l'utilisateur pour générer la réplique suivante. La prise de son ainsi que le temps restant sont indiqués par des feedbacks visuels jugés intuitifs, par souci de simplicité.

Bien que les participants soient invités à développer leurs réponses, certains se contentent de quelques mots, puis attendent. Pour essayer de réduire la longueur des silences après la réponse du participant dans les fichiers audio, une condition a alors été ajoutée : si pendant 3 secondes le système détecte un arrêt de parole, il met fin à l'enregistrement automatiquement.

7.2.3 Corpus

Cette expérience a été maintenue ouverte pendant plusieurs mois. Les différentes statistiques des données collectées sont données dans la Table 7.8 suivante.

Tous participants confondus	
Nombre de comptes ayant commencé l'expérience	111
Durée totale enregistrée	3h19
Participants ayant fini l'expérience	
Nombre de participants finaux	54
Nombre d'adultes	44
Nombre de femmes	36
Nombre d'hommes	16
Score moyen de familiarité avec les technologies	3.80/5
Temps moyen total parlé par utilisateur	3 min 21

TABLE 7.8 : Caractéristiques du corpus collecté

7.2.4 Résultats

Engagement

Comme le montre la Table 7.8, environ la moitié des participants s'inscrivant à l'expérience ne la finissent pas en réalité. Le graphique Figure 7.6 donne alors un aperçu plus détaillé des différentes étapes où les participants abandonnent.

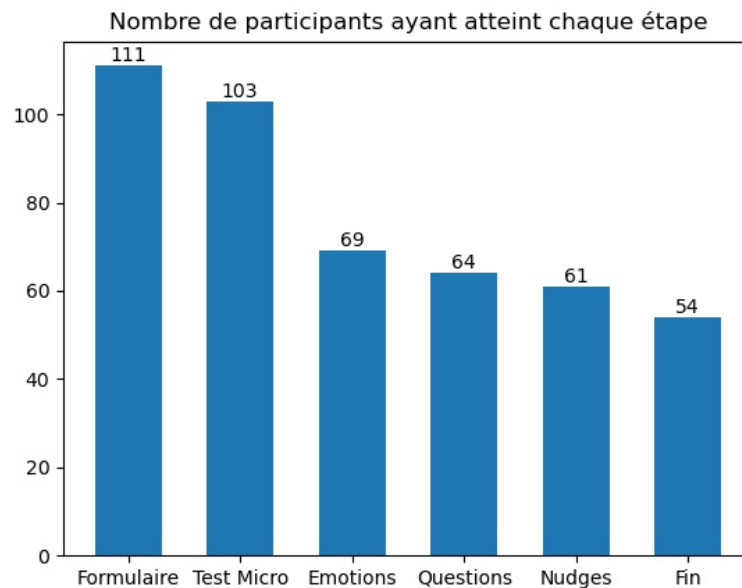


FIGURE 7.6 : Engagement des participants

Ce graphique montre ainsi un manque d'engagement des participants sur cette expérience à distance. Nous pouvons notamment observer un nombre élevé d'abandons entre le moment où le micro est testé et le moment où l'utilisateur enregistre sa première émotion. Le fait de devoir s'enregistrer à l'oral apparaît visiblement comme un facteur dissuasif important. Une autre raison pourrait également être que l'autorisation du micro sur téléphone peut nécessiter plusieurs étapes, l'abandon serait alors en partie lié au support utilisé.

Il semble également y avoir un abandon progressif à chaque étape. Bien qu'il soit difficile d'en estimer les causes précises, la longueur de l'expérience (8 min environ) et la présence de répétitions dans le protocole (questions reformulées avec nudges) ont pu jouer dans ces décisions.

Environnement et pertes

Un certain nombre de données a également été perdu en raison du bruit et des distractions liés à l'environnement. Le nombre total de segments muets ou inaudibles est estimé à 18%. Bien que nous n'excluons pas que la connexion puisse être un problème dans quelques cas, la majorité des segments audio inaudibles sont tout simplement muets, l'utilisateur n'ayant mal réglé son micro ou n'ayant pas répondu.

Étant donné la structure modulaire choisie (avec un traitement en pipeline pour l'audio), ces pertes de données se répercutent sur l'ensemble du traitement restant. En tout, nous estimons qu'un tiers des tours de paroles dans nos données ne sont pas exploitables pour estimer le succès d'un nudge à cause d'une des raisons suivantes :

- un des tours est muet (problème micro, de connexion ou abstention)
- un des tours de parole n'a pas réussi à être transcrit par l'ASR (inaudible ou erreur)
- une des réponses (réponse initiale ou réponse suite au nudge) n'a pas été comprise (sémantiquement) et empêche ainsi une comparaison.

Succès général des nudges

Le succès général des nudges dans cette expérience, indépendamment de la stratégie de nudge utilisée, a été calculé en moyenne selon deux mesures différentes :

- une mesure binaire indiquant si l'utilisateur avait modifié ou non son choix à la suite du nudge
- une mesure numérique mesurant dans quelle mesure le nouveau score donné par l'utilisateur allait dans la direction du comportement suggéré par le nudge. En effet, si le choix initial de l'utilisateur était au-dessus du score seuil (valeur médiane 3), les nudges sélectionnés l'inciteraient à baisser sa note en favorisant le comportement opposé. Cette mesure m est ainsi calculée selon la formule suivante, en notant x_i le choix initial de l'utilisateur et x_n son choix après le nudge :

$$- \text{ si } x_i > 3 : m = x_i - x_n$$

$$- \text{ sinon : } m = x_n - x_i$$

La mesure numérique permet ainsi de montrer que les nudges orientent bien la décision de l'utilisateur dans la direction souhaitée, bien que faiblement (moins d'1 point par nudge), avec environ 21% de nudges ayant eu l'effet opposé.

Ce déclenchement d'un comportement opposé peut être attribué à un phénomène de réactance : une réaction généralement énervée suite à un sentiment de perte de liberté. Certains participants ayant l'impression de se faire manipuler ont ainsi pu répondre par esprit de contradiction. La répétition des questions et les erreurs de compréhension du système ont également pu participer à un sentiment d'énervement similaire. Par exemple, quelques participants ont modifié leur réponse initiale pour le score "zéro", qui ne faisait pas partie des alternatives suggérées, afin d'exprimer leur désaccord.

Stratégies

La Table 7.9 donne le succès des différentes stratégies utilisées en moyenne sur l'ensemble du corpus. Elle montre notamment qu'une simple répétition semble être la stratégie la moins performante pour nudger les différents utilisateurs, ce qui semble donc valider notre Hypothèse 2. Ces résultats nécessiteraient néanmoins d'être confirmés avec davantage de données. La stratégie *warning* (Avertissement) semble être la meilleure stratégie au global pour ce corpus. Nous n'incluons pas les pauses comme stratégies ici car elles se sont révélées non statistiquement significatives pour influencer le résultat du nudge (pas de différence significative entre les résultats de nudge avant et après les pauses peu importe leurs positions).

Le pourcentage de réactance, c'est-à-dire les nudges ayant eu l'effet opposé de celui escompté parmi ceux qui ont influencé, est également donné pour chaque stratégie même si ce résultat est à relativiser étant donné le faible nombre d'échantillons. En plus de la mesure numérique moyenne des nudges, ce score permet d'étudier plus précisément l'efficacité exacte de chaque stratégie.

Bien que le domaine de la tâche, l'environnement et la population étudiée soient différents dans les 2 expériences, le succès des stratégies de nudges apparaît plus faible lorsque l'agent est imaginaire ici (figurée par une image). La stratégie *peer effect* par exemple, dont le succès était de 50% pour le robot et 61% pour l'enceinte en moyenne selon les résultats de la première expérience, n'influence l'utilisateur que dans 31% des cas ici (cf Table 7.9). Son succès est même légèrement inférieur à celui obtenu en moyenne par l'examineur humain de la première expérience de 35%.

	Peer-effect	Empathie	Avertissement	Ikea effect	Simple répétition
Occurrences	68	47	55	55	18
Succès (mesure binaire)	31%	26%	35%	27%	17%
Mesure numérique moyenne	0.25	0.28	0.38	0.22	0.28
Réactance pour les nudges influents	24%	17%	16%	20%	33%

TABLE 7.9 : Occurrences et succès des stratégies dans l'ensemble du corpus

L'amplitude du changement de choix semble également plus faible, bien que l'échelle de scores utilisés diffèrent pour les 2 expériences. Les participants modifient leur choix de 0.25 point en moyenne dans la direction du nudge pour le *peer effect* (sur une échelle de 5 points) dans l'expérience 2 (mesure numérique moyenne). En supposant que l'influence de l'échelle soit linéaire, cela correspondrait à une modification de score de 0.5 point sur 10. En effet, pour l'expérience 1, les enfants changeaient leur choix dans la direction du nudge de respectivement 0.8 bille pour l'adulte, 1.5 billes pour le robot et 1.4 bille pour l'enfant (métrique de nudge Table 7.2).

Conclusion

En conclusion, cette expérience automatisée montre les enjeux pour la mise en place d'un système de dialogue en ligne. Elle propose un cadre pour étudier et mesurer les nudges linguistiques oraux et semble valider notre Hypothèse 2 : des **stratégies de nudges oraux sont plus efficaces que de simples répétitions.**

Bien qu'il s'agisse d'un contexte différent de la première expérience, **les stratégies de nudges semblent moins efficaces ici, ce qui pourrait être expliqué par la représentation de l'agent qui est imaginaire** (figurée par une image) dans cette expérience à distance.

Ce Chapitre décrit ainsi les résultats associés à 2 expériences sur les nudges, réalisées dans l'objectif d'entraîner ensuite un SDS. Ceux-ci semblent en particulier valider les Hypothèses 1 et 2 sur leur efficacité formulées section 2.3.4. L'expérience 1 a également apporté des éléments de réponses sur l'Hypothèse 4, révélant certains paramètres paralinguistiques et contextuels pertinents pour une prédiction de nudges.

Ces expériences ont aussi offert la possibilité de comparer différents agents et représentations de ceux-ci montrant qu'elles pourraient jouer un rôle dans l'efficacité des nudges employés.

Les données collectées ont notamment permis d'étudier les émotions et états mentaux dans le contexte d'un dialogue avec nudges à travers de premiers modèles supervisés ce qui servira de point de départ la mise en place du module de détection d'émotions pour notre système décrit dans le chapitre 9.

Chapitre 8

Structure du système de dialogue développé

L'objectif du système de dialogue envisagé est d'adapter des stratégies de nudge en fonction du profil détecté de l'utilisateur notamment grâce à divers modules d'entrée.

Une architecture modulaire (en pipeline) a alors été choisie. Si une architecture end-to-end peut offrir de meilleures performances en évitant notamment la propagation des erreurs au fur et à mesure de la progression des modules, notre choix a cependant été motivé par les raisons suivantes :

- la possibilité d'utiliser certaines parties pendant la phase de collecte pour automatiser l'interaction, en particulier la reconnaissance vocale et l'interprétation sémantique de la réponse de l'utilisateur
- la flexibilité de l'agencement des modules permettant l'ajout ou le retrait d'un module selon sa pertinence dans la détection et l'utilisation des nudges dans le dialogue
- l'explicabilité des modules utilisés

Les sections suivantes décrivent tout d'abord la structure du système de dialogue dans son ensemble, puis détaillent ses différents composants.

8.1 Structure du système dans son ensemble

Une vue du système dans son ensemble est donnée dans la figure 8.1. Pour chaque tour de parole, le traitement du signal de l'utilisateur le long de la pipeline est ainsi montré, de sa réception par le système à la réponse vocale ren-

voyée ensuite par celui-ci. Les différentes étapes de ce processus sont alors détaillées dans les sections suivantes.

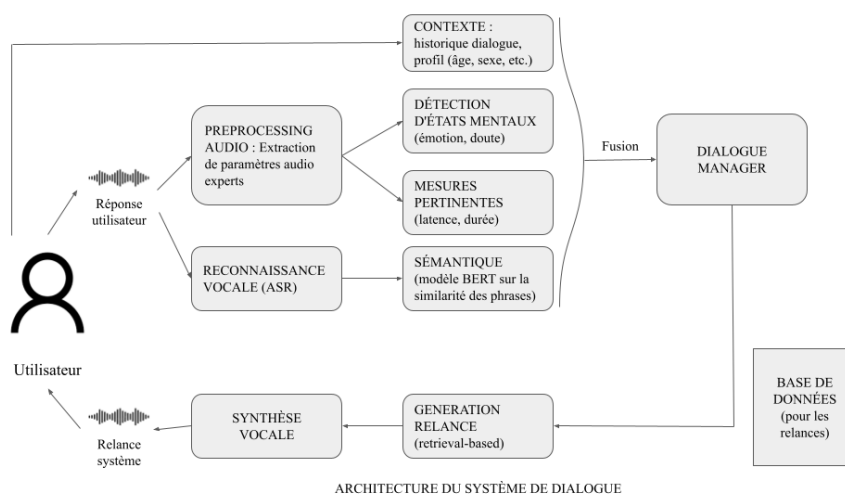


FIGURE 8.1 : Architecture du système

Le signal audio de l'utilisateur est ainsi d'abord enregistré puis prétraité. Les modules d'entrée se composent ensuite de modules de détections non-linguistiques (états mentaux, latence et durée de la réponse de l'utilisateur) et d'un module d'ASR suivi d'un module de NLU (*Natural Language Understanding*). Le module de détection non-linguistique est un classifieur en few-shot learning et le module de NLU utilise une représentation transformers (un modèle de BERT sur les paraphrases) pour prendre en compte l'information contextuelle sémantique. Les sorties de ces modules serviront à représenter l'état actuel du dialogue (ce que nous avons appelé le tour actuel pour reprendre la décomposition de l'état du dialogue présentée section 4.2.3), puis enrichir l'historique pour les tours suivants. La réplique du système est alors choisie par le Dialogue Manager. La génération de la réponse utilisateur est ensuite retrieval-based, construite à partir d'audio pré-enregistrés (synthèse vocale faite en amont).

8.2 Traitement de l'entrée audio

La voix de l'utilisateur est tout d'abord enregistrée sur le site à l'aide d'un code javascript puis envoyée au serveur sous forme de fichier audio "wav" (sampling rate : 48000Hz, 1 channel). Il suit alors 2 traitements parallèles :

- un preprocessing audio cherchant à extraire des paramètres acoustiques experts pour la détection d'émotions

- une transcription permettant des traitements de NLP ensuite

8.2.1 Preprocessing audio

L'aspect audio du signal contient des informations pertinentes pour la détection d'émotions. Il convient donc de préprocesser cette information pour l'adapter à l'entrée de notre modèle de détection d'émotions.

Descripteurs acoustiques

Bien que d'autres approches soient possibles, nous avons choisi d'utiliser des descripteurs acoustiques pour extraire les caractéristiques pertinentes de l'audio à fournir ensuite à un classifieur émotionnel. Les raisons de ce choix sont en particulier :

- Les descripteurs acoustiques peuvent être choisis pour pouvoir traiter des segments audio de différentes tailles, contrairement à d'autres approches qui nécessiteraient plus de prétraitements (approche spectrale par exemple).
- Le prétraitement est relativement court pour 1 audio et peut donc être utilisé en temps réel.

Les descripteurs acoustiques sont extraits du fichier audio de la réponse de l'utilisateur collecté par le serveur à l'aide du logiciel OpenSMILE [Eyben and Schuller, 2015] qui permet d'obtenir des ensembles standardisés de ceux-ci. Plusieurs configurations de paramètres sont proposées avec le logiciel et leur choix est très dépendant du corpus utilisé. Pour nos données, les caractéristiques emobase2010 (1582 caractéristiques) et IS10_compat_paraling (paralinguistique d'InterSpeech 2010 de 1582 caractéristiques) semblaient donner les meilleurs résultats et utilisaient des descripteurs fonctionnels permettant de traiter des signaux de taille variable. Un algorithme aurait également pu être utilisé pour sélectionner le set de paramètres optimal même si l'acquisition progressive des données audio rend son utilisation difficile.

Il est important ensuite de normaliser les paramètres audio obtenus. En effet, de nombreux algorithmes comme les SVM ont des performances dépendant du *scaling* et de la normalisation des données. La normalisation des données de préentraînement, entraînement et évaluation est séparée pour éviter tout biais.

8.2.2 Reconnaissance automatique de la parole : ASR

Afin d'obtenir plus de visibilité sur les données obtenues et permettre ensuite un traitement NLP (*Natural Language Processing*), une reconnaissance vocale est également conduite sur le fichier audio collecté de l'utilisateur. La transcription est ainsi réalisée une fois le segment audio entièrement enregistré. En effet, la réponse doit être enregistrée sur le serveur et celui-ci disposant d'un GPU peut alors effectuer des traitements plus lourds.

Pour des raisons de rapidité, nous avons eu recours à la reconnaissance vocale de Google disponible sur la librairie python SpeechRecognition. Bien qu'elle soit moins performante que d'autres options comme Whisper [Radford et al., 2022] d'OpenAI, elle permet un traitement d'environ 1 seconde sur les segments enregistrés.

Un score de WER (*Word Error Rate*) a alors été calculé pour la reconnaissance automatique utilisée. Sur les données spontanées, nous obtenons un WER de 15.4%. Ce score est notamment expliqué par la différence de qualité des micros, les bruits de l'environnement, etc. Pour diminuer l'impact des erreurs de ce module sur les autres modules, les 2 meilleures hypothèses de transcription sont sélectionnées pour les traitements suivants.

Étant donné le contexte de l'interaction, un lexique a également été ajouté pour pouvoir trancher à base de règles simples dans différents cas d'homonymie pour éviter certaines erreurs récurrentes. Par exemple, l'interaction portant sur l'écologie et le système attendant généralement un score de la part de l'utilisateur, la transcription "Troyes" a systématiquement été substituée en "trois".

8.3 Interprétation émotionnelle et sémantique

8.3.1 Module de détection émotionnelle

Ce module fait l'objet d'un chapitre entier et est développé au chapitre suivant. Il cherche à détecter les émotions suivantes chez l'utilisateur à chaque tour de parole : négative, doute, neutre et positive. Il utilise alors le preprocessing audio pour détecter les émotions à l'aide d'un réseau siamois en few-shot learning selon la méthode exposée dans Feng and Chaspari [2021].

8.3.2 Compréhension sémantique (SLU)

Pour la compréhension sémantique de la réponse utilisateur, des traitements NLP simples ont tout d'abord été appliqués dans les premiers essais. Un mo-

dèle de similarité de phrases *Sentence-Bert*, préentraîné avec des réseaux siamois, a ensuite été utilisé [Reimers and Gurevych, 2019]. Il s'agit d'un modèle multilingue distillé [Reimers and Gurevych, 2020] de manière à réduire la taille des paramètres du modèle. En effet, une des contraintes de notre dispositif est d'obtenir un temps de réponse rapide du système sans avoir à charger un modèle trop lourd. De manière similaire, des efforts ont été effectués en développement pour minimiser le temps de chargement du modèle grâce à l'utilisation de thread maintenant le modèle de langue en mémoire en attente de l'arrivée de transcriptions.

L'interaction avec l'utilisateur ayant été conçue comme relativement simple pour mettre clairement en valeur l'influence des nudges, le nombre d'actes de dialogue à identifier pour le système est relativement réduit. Nous avons ainsi retenu les actes de dialogue suivants pour représenter l'intention de l'utilisateur :

- Incompréhension
- Incertitude
- Accord
- Désaccord
- réponse à une question de nudges incluant un score (score prenant 5 valeurs possibles de 1 à 5)
- autre réponse sans score

L'intention de l'utilisateur est alors estimée en utilisant la similarité cosinus du modèle *Sentence-Bert* par rapport à des phrases de référence, ce qui donne des performances satisfaisantes dans le cadre spécifique de notre tâche. De manière similaire à l'ASR, les 2 meilleures hypothèses d'actes de dialogues détaillés renvoyées par le modèle *Sentence-Bert* sont alors gardées et utilisées pour représenter l'état actuel du dialogue. Avec davantage de données, un classifieur aurait peut-être été préférable en fine-tunant un modèle de langues sur nos classes d'actes de dialogue. Néanmoins, l'erreur estimée de ce module est négligeable et reproduit plutôt les erreurs du module ASR.

En plus de ces actes de dialogue, l'émotion de l'utilisateur est également prise en compte (cf section précédente) pour représenter l'état mental général de celui-ci.

8.4 Suivi de l'état du dialogue

8.4.1 Informations contextuelles

Le système récupère tout d'abord des informations générales sur le profil de l'utilisateur (âge, genre, familiarité avec les technologies) au travers d'un questionnaire préalable à l'interaction. Ces données sont ainsi supposées statiques pour l'étude. Elles seront redonnées à chaque tour de parole au système de dialogue en tant qu'historique.

8.4.2 Fusion

La fusion des informations de l'état du dialogue peut être précoce (fusion au niveau de l'entrée des modèles) ou tardive (fusion des sorties des modèles). Dans notre cas, nous optons pour une fusion tardive pour plus de visibilité.

Le système fusionne alors les sorties des modules de NLU et de détection d'émotions ainsi que les informations contextuelles.

8.4.3 Représentation probabiliste de l'état

En se plaçant dans le cadre du POMDP, une estimation probabiliste de l'état actuel de l'utilisateur est faite à chaque tour de parole. Pour conserver un espace d'état de taille raisonnable, seules probabilités des 2 meilleures hypothèses de NLU sont retenues, de manière similaire à la méthode utilisée dans les approches POMDP de la littérature [Daubigney et al., 2012][Ferreira, 2015].

Habituellement, la modélisation sous forme de POMDP peut être simplifiée en 3 composants : le *but recherché par l'utilisateur* dans ce dialogue, les *informations récupérées au tour actuel* du dialogue et l'*historique* résumant des informations importantes passées. Dans notre cas, le but recherché correspondrait plutôt à l'avancement des étapes de l'interaction (état des questions de nudges : répondues ou non). Les étapes suivant un déroulement très déterministe et l'interaction étant davantage centrée sur les stratégies de nudges, nous l'avons alors simplifié en représentant l'ensemble de l'avancement par une seule variable correspondant au numéro de la question actuelle.

Au final, notre état du dialogue est représenté par les paramètres suivants :

- le but ou plutôt l'état actuel de l'interaction : numéro de la question de nudge actuelle
- les informations du tour de parole : émotion prédite par les modules émotionnels, probabilité des 2 meilleures hypothèses de NLU

- l'historique du dialogue. Il regroupe les informations contextuelles statiques : âge, genre, familiarité avec les technologies et végétarisme. Il comprend également des informations dynamiques correspondant à l'estimation du dernier acte de dialogue et le nombre de pauses effectuées. Ces variables cherchent notamment à décourager le système à trop se répéter.

8.5 Dialogue Manager

Le DM ici a pour but de sélectionner la stratégie de la réplique à retourner à l'utilisateur à partir de l'estimation de l'état du dialogue. Il prend ainsi en entrée la représentation du suivi du dialogue et renvoie une action. Dans notre cas, les actions considérées sont les suivantes :

- Répéter une question (en cas d'incompréhension par exemple)
- Effectuer une stratégie de nudges (4 stratégies au choix utilisées pour la 2ème expérience : *peer effect*, *empathy*, *warning*, *ikea effect*)
- Effectuer une pause

Le choix des actions se fait alors selon l'algorithme de RL sélectionné (cf Chapitre 10). Notre DM suit plus précisément une approche hybride puisque les actions sont choisies entièrement par l'algorithme, à l'exception des règles suivantes :

- le nombre de stratégies de nudges par question est limité à 2 et le nombre de pauses sur le dialogue entier à 2
- une même stratégie ne peut pas être répétée

Des récompenses diffuses sont ajoutées à chaque tour de parole grâce au résultat des modules de détection d'émotions, pour pallier au manque de récompense habituel avant la fin du dialogue.

8.6 Génération et synthèse de la réplique du système

La génération des répliques du système a été faite de manière retrieval-based, c'est-à-dire que ces répliques ont été construites à partir de blocs de réponses

prédéterminés. Pour cela, différents segments audio ont été enregistrés en amont, simplifiant ainsi la tâche de synthèse vocale. La construction de la réplique du système dépend alors du contexte et de l'action sélectionnée par le DM.

La raison de cette simplification de la partie génération et synthèse est de nous concentrer sur le choix d'une stratégie, par exemple utiliser l'humour ou un nudge Peer-effect et de limiter le nombre de formulations possibles de cette stratégie. En outre, une erreur de formulation pourrait avoir des effets imprévus et peut-être se rattacher à une autre stratégie de nudge ce qui rendrait l'expliquabilité du système difficile.

Un paramètre qui pourrait cependant être davantage expérimenté dans notre approche est la voix utilisée pour la synthèse. Nous avons utilisé un outil de transcription en ligne (naturalreader) et avons enregistré toutes les répliques avec une seule voix française de femme. Cependant, l'utilisation de différentes voix selon l'utilisateur pourrait être une forme de nudges à tester. Des travaux se sont notamment intéressés à la modification du timbre de la voix de synthèse pendant l'interaction pour s'adapter à l'utilisateur (ref à retrouver).

8.7 Entraînement et évaluation du système

Le système est ensuite entraîné grâce aux données collectées durant la deuxième expérience dans laquelle la tâche du système est d'inciter l'interlocuteur à modifier ses choix initiaux sur un sujet d'écologie. Les métriques d'évaluation sont alors le taux de réussite ainsi que la moyenne des récompenses cumulées sur chaque dialogue (cf Chapitre 10).

Chapitre 9

Mise en place d'un modèle d'analyse émotionnelle

Les émotions ayant un lien étroit avec nos jugements et prises de décision [Ange et al., 2011][Damasio, 1994], nous avons voulu les prendre spécifiquement en compte dans notre modèle grâce à un module de détection. Notre objectif était d'obtenir une estimation rapide de l'état émotionnel de l'utilisateur à chaque tour de parole. Il nous fallait alors répondre à certaines problématiques. Quelles émotions rechercher en particulier ? Comment les représenter ? Comment entraîner notre modèle en ligne avec peu de données ? Les sections suivantes tentent de répondre à ces questions dans cet ordre.

9.1 Etats mentaux retenus pour la prédiction de nudges

Afin de choisir quels états mentaux détecter, nous nous sommes inspirés des résultats empiriques de la première expérience, pour laquelle les annotations ont été présentées dans la section 7.1.4. Étant donné le peu d'annotations recueillies pour les différentes étiquettes fines émotionnelles, nous avons décidé d'utiliser seulement des macro-étiquettes par la suite : **négatif**, **positif** ou **neutre**. En effet, le fait que ces émotions fines aient été peu labellisées peut signifier qu'elles ne sont pas très présentes dans le dialogue et que de ne pas les distinguer précisément ne devrait pas beaucoup impacter les performances. Nous pouvons noter également que les émotions négatives sont très sous-représentées par rapport aux émotions positives (59 contre 354) dans ce corpus (cf Table 7.3).

En plus du caractère positif de l'émotion, nous avons choisi de prendre en

compte aussi le **doute** dans nos modèles suivants. D'une part, cette classe (Assurance/Doute) était la plus annotée dans le premier corpus. D'autre part, la latence des tours de parole ainsi que la réponse à la question incongrue, deux caractéristiques prédictives des nudges importantes, pourraient être reliées à des marques de doute.

Une estimation de la difficulté à classifier les différentes dimensions annotées (émotionnelles et affect bursts en particulier) sur le corpus de la 1ère expérience est alors donnée à travers les performances de modèles simples de classification (cf Table 9.1). Les scores de f1, avec une nested cross-validation (10-fold), sont donnés pour deux modèles :

- un classifieur SVM (Support Vector Machine) avec un kernel RBF (Radial Basis Function) et un paramètre C de régularisation de 1 choisi selon les résultats de validation.
- un classifieur RF (Random Forest) avec 100 classifieurs arbres de décision.

Ceux-ci ont été entraînés en extrayant pour chaque audio un set de paramètres acoustiques experts : le set de caractéristiques "emobase2010" d'OpenSMILE (comprenant 1582 caractéristiques).

Classes	SVM	R.F.
Emotions Positives / Négatives / Autre	0.56 ± 0.01	0.45 ± 0.03
Intéressé / Désintéressé / Autre	0.56 ± 0.01	0.49 ± 0.01
Doute / Assurance / Autre	0.59 ± 0.01	0.48 ± 0.01
Compréhension / Incompréhension	0.65 ± 0.04	0.66 ± 0.02
Poli / Familier	0.82 ± 0.01	0.79 ± 0.03
Hésitation / Rire / Autre	0.71 ± 0.01	0.64 ± 0.01

TABLE 9.1 : Scores F1 moyens pour les classificateurs d'états mentaux, d'affect bursts et de registres de langue (*10-fold nested cross-validation*) avec les caractéristiques du set *emobase2010* d'OpenSMILE (1582 caractéristiques)

9.1.1 Accord Inter-annotateur

L'annotation étant manuelle, une mesure de kappa a également été calculée sur 15 dialogues pour mesurer l'accord entre différents annotateurs et estimer

la qualité des annotations. Seuls les segments sélectionnés à la fois par les 2 annotateurs ont été pris en compte. Les scores de kappa obtenus sur les émotions (macro-étiquettes) et sur le doute (assurance/doute) sont respectivement de 0.76 et 0.68 qui correspondent à un accord fort. Les divergences d'annotations observées sont notamment davantage discutées dans l'article [Ali Meheni et al., 2020a].

9.2 Test perceptif

Pour étudier plus précisément comment représenter les émotions dans notre modèle, un test perceptif a été réalisé sur le site utilisé pour la deuxième expérience (cf Figure 9.1).

9.2.1 Objectifs

Les objectifs de ce test étaient de :

- comparer l'apport de l'audio par rapport au texte seul
- tester un grand nombre de catégories émotionnelles plus fines pour analyser sous quelles formes les macro-émotions se retrouvent le plus
- comparer une représentation catégorielle et continue (bi-dimensionnelle : valence et activation)
- comparer la reconnaissance d'émotions actées et spontanées dans notre corpus
- analyser les émotions mélangées
- prendre en compte la personnalité de l'annotateur à l'aide d'un test de personnalité préalable : test OCEAN [Wigging, 1996].

9.2.2 Protocole

Avant de commencer le test perceptif, l'annotateur réalise un test de personnalité OCEAN (O : ouverture aux autres, C : caractère consciencieux, E : extraversion, A : amabilité, N : neuroticisme) [Wigging, 1996] constitué d'une quarantaine de questions.

Pour chaque audio ensuite, l'annotation perceptive se fait en 2 étapes :

1. d'abord avec la transcription uniquement
2. puis en ajoutant l'audio

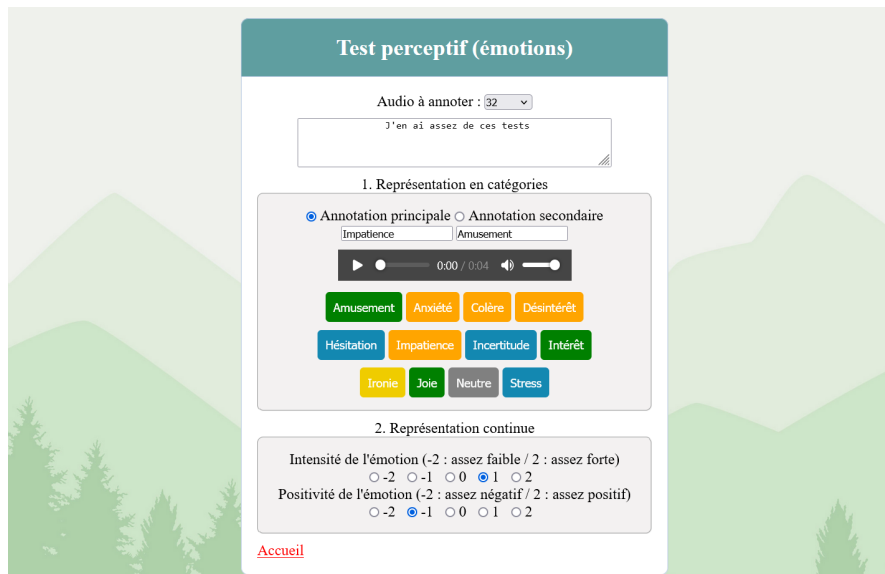


FIGURE 9.1 : Visualisation du test perceptif proposé (lorsque la transcription est disponible)

Choix de l'audio

Le test est réalisé sur 32 audio qui ont été annotés initialement dans la 1ère expérience selon 4 classes : négatif, doute, neutre et positif. La moitié des audio était actée et l'autre spontanée, les 4 classes étant représentées de manière équilibrée sur chacune.

Annotations catégorielles

Douze annotations fines sont proposées à l'annotateur, chacune rattachée à une macro-étiquette (cf Table 9.2). L'ironie est une émotion difficile à interpréter positivement ou négativement car elle nécessite de connaître quel aspect de la phrase a été nié ironiquement, selon le concept d'*ideational reversal irony* [Dyrel, 2014]. Nous avons jugé cependant qu'une annotation ironique dans nos fichiers audio pouvait être interprétée comme une réponse négative aux questions du robot.

Pour prendre en compte la présence d'émotions mélangées, l'annotateur peut attribuer une émotion majeure (principale) et une émotion mineure (secondaire).

Une fois qu'il a attribué une émotion principale pour la transcription, l'audio devient alors disponible et il est libre de changer son choix. L'évolution de ses choix est gardée en mémoire.

Macro-étiquettes	Annotations plus fines
négatif	anxiété, colère, désintérêt, impatience, (ironie)
doute	hésitation, incertitude, stress
neutre	neutre
positif	amusement, intérêt, joie

TABLE 9.2 : Catégories d'étiquettes macro et micro testées

Annotations continues

L'annotateur donne également pour chaque audio une valeur pour la valence (négatif-positif) et l'activation (faible-forte). Pour simplifier la collecte, les valeurs à choisir sont sous forme discrète : valeurs entières sur l'intervalle [-2 ;2].

9.2.3 Résultats

Quinze annotateurs se sont prêtés au test. Tout d'abord, une mesure d'accord général a été calculée à l'aide d'un coefficient alpha de Krippendorff, qui permet notamment ici de prendre en compte les notations manquantes de certains annotateurs (cf Table 9.3).

Annotations	Alpha de Krippendorff
A.P.I. (Annotation Principale Initiale) (sans audio)	0.11
A.P.I. regroupée en macro-classes	0.36
A.P.F. Annotation Principale Finale (avec audio)	0.31
A.P.F. regroupée en macro-classes	0.55
Annotation secondaire (finale)	0.15
Valence	0.58
Activation	0.35

TABLE 9.3 : Accord inter-annotateurs (alpha de Krippendorff)

Néanmoins, pour pouvoir rendre compte de l'écart de chaque annotateur à l'émotion de référence (émotion actée ou spontanée annotée en amont), des mesures de kappa (de Cohen) ont aussi été calculées. L'annotation de référence était composée des 4 macro-classes : négative, neutre, doute et positive.

Une équivalence a alors été faite lorsque les catégories plus fines ont été comparées aux macro-étiquettes. A titre de comparaison, une équivalence entre les macro-classes et la dimension de valence a aussi été effectuée même si elle est légèrement biaisée car elle ne prend pas en compte la deuxième dimension (ni la classe doute).

Pour vérifier la cohérence d'un annotateur pour différentes représentation d'émotions, des kappa intra-annotateur, pour un même annotateur donc, ont également été utilisés.

Ces mesures de kappa (de Cohen) ont alors été moyennées sur l'ensemble des annotateurs et sont présentées dans la Table 9.4.

Annotations comparées	Types d'annotations	Kappa moyen
la V.R. (valeur de référence) et l'A.P.I. (Annotation Principale Initiale) (sans audio)	macro-classes	0.41
la V.R. et l'A.P.F. (Annotation Principale Finale) (avec audio)	macro-classes	0.60
la V.R. et l'Annotation secondaire	macro-classes	0.49
la V.R. (sans le doute) et la valence	classes négatif-neutre-positif	0.18
l'A.P.I. et l'A.P.F.	catégories	0.36
	macro-classes	0.42

TABLE 9.4 : Accord intra-annotateurs (kappa de Cohen moyen)

Conclusions

Nous pouvons alors tirer les conclusions suivantes sur les émotions représentées dans le corpus de la 2ème expérience :

- **L'audio semble améliorer la perception de l'émotion** car les scores d'alpha de Krippendorff augmentent significativement lorsqu'on ajoute l'audio. Au niveau de l'annotateur, le score de kappa moyen mesurant l'accord entre l'annotation macro-classe avec l'annotation de référence augmente également avec l'audio. Cela justifie le choix de détecter l'émotion à l'aide de paramètres acoustiques.
- L'accord inter-annotateurs (alpha de Krippendorff) pour la valence est relativement bon, contrairement aux scores relatifs à l'activation, ce qui nous a dissuadé d'utiliser une représentation dimensionnelle. Le fort accord sur la valence peut alors être interprété comme le fait qu'elle relève d'une dimension similaire aux macro-classes. L'accord entre annotateurs étant trop faible lorsque les classes ne sont pas regroupées en macro-classes, nous les utiliserons donc pour la suite de nos traitements.
- Les annotateurs changent significativement leur avis durant le test perceptif. En effet, l'accord intra-annotateur est relativement faible entre la première annotation et l'annotation finale (0.36 avec les 12 catégories et 0.42 avec les macro-classes), ce qui souligne la difficulté de la tâche et l'apport de l'audio pour la décision.

Ces observations nous ont conduit à préférer un modèle prenant en entrée un signal audio, plutôt que l'information textuelle dérivée de l'ASR pour la détection d'émotions.

Néanmoins, un modèle *transformers* préentraîné pour de l'analyse de sentiment, puis *fine-tuné* sur nos transcriptions pourrait capturer des informations complémentaires à l'audio, ce qui permettrait d'améliorer les performances finales en fusionnant ces 2 modèles, comme certains travaux ont pu le montrer [Deschamps et al., 2022] (fusion d'un modèle Wave2vec2 et FlauBERT).

Nous avons fini par abandonner cette approche afin de garder un système modulaire suffisamment léger. Un autre problème était le peu de données annotées pour un fine-tuning et le fait que les données automatiquement annotées dans l'expérience 2 étaient actées, conduisant à des transcriptions fixes ou peu réalistes.

9.3 Module de détection d'émotions avec apprentissage Few-shot

9.3.1 Choix du modèle

Différents modèles pour la détection d'émotions en particulier dans le cas d'un apprentissage few-shot ont été proposés. Nous avons opté pour une approche à partir de réseaux siamois reprenant la méthode présentée dans [Feng and Chaspari, 2021]. Celle-ci repose sur l'entraînement en amont d'un réseau siamois apprenant à extraire une métrique de similarité entre les différentes classes. Des corpus, dont les domaines sont différents de l'expérience 2, sont alors utilisés pour ce préentraînement (cf section suivante). Celui-ci se fait en sélectionnant aléatoirement des couples d'échantillons de même classe émotionnelle (négatif, neutre, doute, positif) et des couples d'échantillons de classes différentes et en leur attribuant une étiquette binaire : 1 pour la même classe et 0 pour des classes différentes. Les nombres d'échantillons de même classe et de classes différentes sont pris de manière égale. Le réseau siamois est alors entraîné de manière supervisée.

Une fois le réseau siamois entraîné, il s'agit alors de faire un **transfert de connaissance** de ce réseau pour notre tâche de classification. Les poids de ce réseau siamois préentraînés sont donc copiés pour entraîner en *few-shot learning* un réseau *feedforward* sur la base de données d'intérêt (corpus de l'expérience 2). Il s'agit ainsi d'une forme de *transfer learning* où l'apprentissage d'une métrique de similarité codée par le réseau siamois est transféré à une tâche similaire pour un autre domaine.

Le modèle siamois implémenté possède 3 couches cachées linéaires (avec fonction d'activation ReLU entre les couches et une fonction d'activation sigmoïde en sortie) de tailles respectives 128, 64 et 16. Une fonction de perte de *cross-entropy binaire* est utilisée permettant d'obtenir en sortie une probabilité binaire de la classe prédite :

De manière similaire, son équivalent pour la classification multi-classes, la fonction de perte *cross-entropy* (avec une fonction d'activation softmax en sortie) est utilisée pour le modèle *feed-forward* FNN permettant d'obtenir une probabilité des 4 classes en sortie. Une 4ème couche cachée linéaire de taille 8 (avec fonction d'activation ReLU) est également ajoutée à la fin du réseau *feed-forward* afin de permettre l'apprentissage supervisé d'une classification multi-classe à partir de la représentation binaire apprise par les couches précédentes (copiées du réseau siamois).

9.3.2 Préentraînement

Base de données utilisées

Pour le préentraînement et l'entraînement du modèle, 3 corpus de voix françaises ont été utilisés (voir Table 9.5). Ces corpus ont des caractéristiques très différentes (environnement, type de données, âge, etc.) ce qui apportera de la robustesse à nos modèles.

	JEMO	Expérience 1	Expérience 2
Environnement	Laboratoire	Ecole primaire	Site internet (foyers)
Qualité sonore	bonne (micro champ proche)	variable (micro cravate et de caméra)	médiocre (support non contrôlé)
Age	24-50	5-10	10 enfants et 44 adultes
Genre	27H et 35F	53H et 38F	16H et 36F
Type de données	Actées	Spontanées	Actées et Spontanées
Unité d'annotation	tour de parole	chunks	tour de parole
Nombre de segments	832	799	738
Kappa	-	≈ 0.7	0.6
Emotions	colère, joie, tristesse, neutre	positif, négatif, neutre, doute, etc.	positif, négatif, neutre, doute

TABLE 9.5 : Corpus utilisés pour l'entraînement de modules de reconnaissance émotionnelle

Parmi les différences importantes, nous pouvons noter :

- *le type de données* : Comme l'ont montré plusieurs études [[Batliner et al., 2003](#)][[Vogt and Andre, 2005](#)], les modèles entraînés sur des données ac-tées obtiennent de meilleures performances que sur des données spon-tanées.
- *l'âge et le sexe* : Les voix d'enfants sont en moyenne plus hautes que celles des femmes, elles-même plus aiguës que celles des hommes, avec une fréquence fondamentale aux alentours de 200-300, 200 et 100 Hz res-pectivement. Selon [Tahon \[2012\]](#), la variabilité du locuteur (enfant, adulte, personnes âgées) dans l'analyse émotionnelle est un facteur important pour le choix de descripteurs acoustiques robustes.
- *l'environnement et les supports utilisés pour l'enregistrement* : Les bruits de l'environnement ainsi que la qualité et le type du support d'enregistre-ment influent également sur les performances finales du modèle. [Wöllmer et al. \[2011\]](#) ont notamment comparé les performances d'apprentissage en enregistrant un corpus avec deux supports audio différents : un micro-cravate (bonne qualité, proche) et un micro interne de caméra (mauvaise qualité et éloigné). Ils ont noté que si les données à tester sont bruitées, les performances seront meilleures en incluant ce type de bruit dans l'ap-prentissage.

Un autre aspect rendant l'apprentissage compliqué pour des données col-lectées dans un environnement non contrôlé comme un foyer est la pré-sence de distractions orales. Certains audio comportent ainsi les émo-tions d'autres individus dans le background, différentes de celle exprimée par l'utilisateur. Dans d'autres audio, l'utilisateur dialogue pendant l'expé-rience avec un autre interlocuteur en exprimant parfois des émotions dif-férentes, comme un agacement envers un ami ou membre de la famille.

D'autre part, le corpus JEMO ne comporte pas les mêmes catégories d'émo-tions que les deux autres corpus. Il permet néanmoins d'augmenter la taille des données et d'apporter une meilleure robustesse au système. Les interac-tions avec le robot comportant très peu de tristesse, nous avons alors conservé seulement les émotions de colère, neutre et joie en leur attribuant respective-ment les macro-étiquettes négatif, neutre et positif.

Le préentraînement du modèle siamois est ainsi effectué sur les corpus de la première expérience et celui de JEMO tandis que l'apprentissage few-shot est réalisé dans un deuxième temps sur le corpus de l'expérience 2 à partir des

	Préentraînement	Entraînement	Evaluation
Corpus	JEMO et Expérience 1	Expérience 2 (données actées)	Expérience 2 (données spontanées)
Nombre de Segments	1631	410	328

TABLE 9.6 : Répartition des corpus pour le préentraînement, l’entraînement et l’évaluation

émotions actées demandées en début d’interaction (cf table 9.6). Ce préentraînement est exécuté jusqu’à ce que la fonction de perte (*cross-entropy binaire*) cesse de diminuer sur le set de validation, ce qui semble atteint pour environ 32 000 couples d’exemples (générés aléatoirement sur les corpus JEMO et de l’expérience 1). Un taux d’apprentissage est alors déterminé à 0.0005 pour l’optimizer Adam avec des batchs de taille 64.

9.3.3 Entraînement en ligne

A l’initialisation, les poids d’un réseau siamois pré-entraînés sur les corpus JEMO et de l’expérience 1 sont copiés dans un réseau *feed-forward*. L’initialisation est ainsi la même pour chaque utilisateur. Dans le cas d’une utilisation en ligne, il s’agit alors de maintenir chargé en mémoire ce modèle.

Pour adapter ce modèle neuronal à un utilisateur en vue d’une interaction, nous entraînons alors celui-ci sur les données actées disponibles déjà collectées par le site (2 échantillons pour chaque émotion par utilisateur). Pour évaluer cette approche, des émotions d’interactions de l’expérience 2 ont été annotées. Les données pour l’entraînement et l’évaluation du modèle sont présentées dans la table 9.7. Il s’agit donc d’un apprentissage *few-shot* avec 2 exemples d’émotions par émotion par utilisateur.

Pour évaluer ce que donne un apprentissage en ligne, nous testons l’entraînement du modèle à différentes phases de collecte. Nous ne gardons alors dans le set d’entraînement que les données des utilisateurs présents dans le set de test à chaque fois. Si les données actées d’entraînement sont naturellement équilibrées, celles spontanées prises pour l’évaluation ne le sont et comportent comme beaucoup de corpus une sur-représentation d’émotions neutres. Plusieurs méthodes existent alors pour y remédier : ajouter un poids

Données	Types de données	Nombre
Entraînement : répartition équilibrée des 4 émotions	actées (50% avec un texte à répéter, 50% sans)	410
Evaluation : 4 émotions	spontanées	328
Evaluation : sous-échantillonnage	idem	≈ 148
Emotion négative	id.	37
Emotion doute	id.	45
Emotion neutre	id.	186
Emotion positive	id.	60

TABLE 9.7 : Données émotionnelles annotées de l'Expérience 2 disponibles pour l'entraînement et l'évaluation du modèle *feed-forward*

aux données, les échantillonner (sous-échantillonnage ou sur-échantillonnage), etc. Nous effectuons ici un sous-échantillonnage (les classes sur-représentées sont échantillonnées aléatoirement) afin de ne pas diminuer la variance des données d'évaluation (ce que pourrait induire un sur-échantillonnage).

Les scores f1 (moyennés sur 10 runs) équilibrés sur les 4 émotions doute, négatif, neutre et positif sont présentés dans la table 9.8 pour les modèles SVM, *Feed-forward* FNN-SNN avec préentraînement par réseau siamois et *Feed-forward* FNN sans. Le modèle SVM ici utilise un noyau gaussien et a un paramètre C de pénalisation fixé à 10 et est utilisé avec le modèle FNN comme *baseline*.

Nous pouvons alors constater peu d'amélioration avec pré-entraînement à l'aide d'un réseau siamois pour l'apprentissage en *few-shot*. La tâche de classification ici demeure compliquée en raison de la taille limitée du corpus de préentraînement de données françaises ainsi que des données réelles bruitées et non contrôlées collectées sur le site. Celles-ci comportent en effet les caractéristiques négatives suivantes : la présence de long silence, une mauvaise qualité de micro, un bruit en arrière-plan perturbateur. La variance élevée des scores f1 peut être due notamment à l'inégalité dans la qualité des audio utilisés. A titre de comparaison, est aussi donné le f1-score en cross-validation (5-fold) sur les données d'entraînement actées, relativement meilleur, ce qui souligne la difficulté de la tâche.

	SVM	FNN	FNN-SNN
Evaluation pour 2 utilisateurs	0.25 ± 0.2	0.20 ± 0.14	0.32 ± 0.19
Evaluation (set entier)	0.30 ± 0.3	0.29 ± 0.03	0.30 ± 0.04
Validation croisée (5-fold) sur les données actées d'entraînement	0.41 ± 0.04	0.37 ± 0.03	0.38 ± 0.04

TABLE 9.8 : Résultats (f1-score moyennée sur 10 runs) des modèles entraînés et évalués sur les données de l'expérience 2 (table 9.7)

L'évolution du score f1 en fonction du nombre d'utilisateurs, pour simuler un accès progressif aux données, est présenté dans le graphique 9.2 pour le modèle *Feedforward* étudié et pour un modèle SVM à titre de comparaison (mêmes paramétrisation que précédemment). Augmenter le nombre d'utilisateurs sans augmenter le nombre d'exemples d'émotions par utilisateurs ne semblent pas améliorer la performance de notre modèle FNN-SNN ici avec les données et paramétrisation utilisées.

Avec plus de données, il pourrait être intéressant de tester l'apprentissage de plusieurs modèles simultanément en fonction du genre et de l'âge de l'utilisateur, ces données ayant été indiquées avant l'expérience par l'utilisateur. Une approche par clusterisation pourrait ainsi être pertinente pour une meilleure adaptation aux caractéristiques de la voix (par rapport à la fréquence fondamentale en particulier).

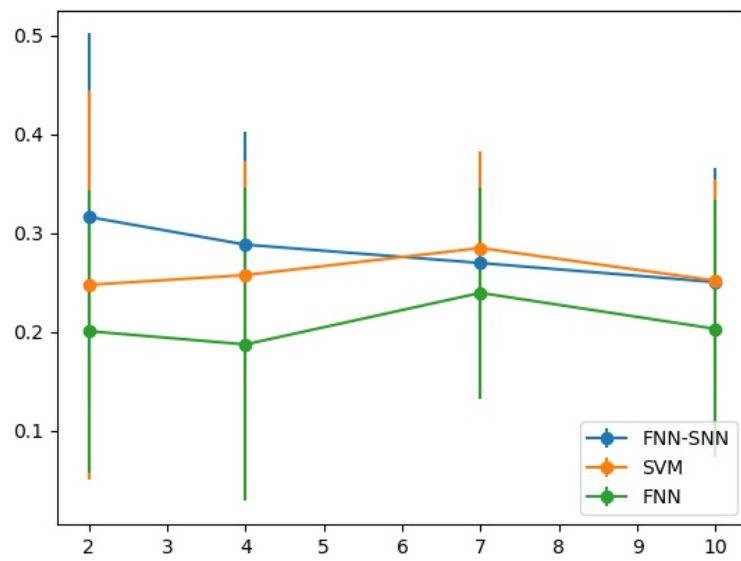


FIGURE 9.2 : Evolution de la f1-score en simulant l'accès progressif de données en ligne avec le nombre d'utilisateurs

Chapitre 10

Apprentissage par renforcement

Afin d'apprendre à notre système à adapter ses stratégies de nudges à l'utilisateur, nous discutons tout d'abord de la modélisation de cette tâche en temps que problème d'apprentissage par renforcement, puis évaluons la faisabilité d'un apprentissage en ligne à partir de nos données. Nous cherchons alors à répondre aux questions de recherche relatives à l'apprentissage avec peu de données, l'apprentissage personnalisé de nudges ainsi que l'évaluation d'un système de nudges. Plus précisément, cette partie s'intéresse aux questions suivantes :

- Quelles récompenses doit-on choisir pour optimiser l'apprentissage ?
- Comment construire une simulation de l'utilisateur avec peu de données d'entraînement ?
- Hypothèse 3 sur les nudges : Les stratégies de nudges sont-elles significativement plus efficaces lorsqu'elles sont personnalisées ?

10.1 Modélisation du problème

10.1.1 Modélisation de la récompense

La modélisation de la récompense du système de dialogue s'articule autour de 2 points : la récompense reflétant l'état estimé du dialogue et les récompenses supplémentaires obtenues par *reward shaping*.

La récompense selon l'état du dialogue rend compte généralement du succès ainsi que de l'efficacité (rapidité d'exécution) de la tâche. Ce premier aspect peut être estimé assez facilement dans notre cas en fonction du succès ou non du nudge (estimation faite par le module de SLU). A cette fin, une récompense positive forte est donnée selon la métrique utilisée en cas de succès et une récompense négative faible (fixée à -1) à chaque tour sinon en cas d'échec pour

encourager le système à s'adapter rapidement. Pour rappel, en cas de succès (ou suite à 2 échecs), le système passe à la question de nudge suivante. En fonction de la métrique utilisée, la récompense en cas de succès est exprimée ainsi :

- si la métrique ne mesure que le succès binaire du nudge (changement de score dans la bonne direction) : une récompense constante (fixée à +5) est attribuée
- si la métrique prend en compte l'écart par rapport au score initial : la récompense est pondérée par la valeur de la métrique (+3 x la métrique)

Etant donné que les récompenses ne sont pas aussi espacées que dans un système de dialogue classique (où le succès est généralement déterminé seulement à la toute fin), le besoin de récompenses additionnelles pour guider l'apprentissage est moins présent. Il peut néanmoins être effectué en fonction du résultat du module de détection émotionnelle avec par exemple +0.75, -0.75 et 0 respectivement selon si l'émotion est positive, négative ou autre (neutre, doute).

10.1.2 Modélisation de l'utilisateur et de l'environnement

Qu'il s'agisse d'un entraînement ou pré-entraînement hors-ligne, ou même d'une première estimation pour un algorithme apprenant en zero ou *few-shot learning*, une simulation de l'utilisateur est une méthode souvent bénéfique pour initialiser le *Dialogue Manager* [Ferreira, 2015][den Hengst et al., 2019]. En l'absence de larges corpus d'interactions, une simulation permet d'évaluer le système en limitant le recours à de véritables utilisateurs (généralement coûteux en temps, moyens, etc.).

Facteurs considérés

Pour évaluer la convergence du système, nous nous intéressons à un modèle utilisateur de buts. Il doit alors prendre en compte les facteurs suivants :

- *l'environnement* : Une erreur est généralement ajoutée artificiellement pour représenter les erreurs de compréhension. Dans notre cas, les données utilisées sont obtenues à partir des premiers modules déployés et comportent ainsi déjà une part d'erreur. Néanmoins, pour l'analyse, les réponses ambiguës ont été retirées donc un taux d'erreur doit être ajouté artificiellement.

- *l'intégration des informations statiques et dynamiques de l'utilisateur* : Le modèle doit cumuler des informations fluctuantes selon le tour de parole comme les émotions de l'utilisateur ou l'historique du dialogue ainsi que d'autres non-fluctuantes liées au profil (âge, sexe, etc.). Nous avons choisi de les représenter au même niveau dans l'état du dialogue.
- *la non-stationnarité des utilisateurs* : L'expérience comportant certaines répétitions dans son protocole, un modèle idéal devrait représenter les potentiels changements de comportements de l'utilisateur durant l'interaction (en raison d'une distraction ou d'un agacement plus ou moins exprimé). Cet aspect est a priori contenu dans les données collectées.

Corrélation des paramètres

Pour estimer les paramètres pertinents pour un premier modèle de l'utilisateur, il est important de vérifier une éventuelle corrélation des variables. Nous avons alors découpé les tours de paroles en vecteurs comportant les informations suivantes :

- des informations contextuelles statiques : le numéro d'identification de l'utilisateur, la position de la pause dans l'interaction (déterminée pour chaque utilisateur), ses informations de profil (âge, sexe, familiarité avec les technologies)
- des informations fluctuantes sur chaque tour de dialogue ou pendant le dialogue : le numéro de question, le numéro de l'essai c'est-à-dire la tentative de nudge pour cette question, la stratégie utilisée, le score initial (1er score) donné pour cette question et le résultat du nudge

La figure 10.1 montre alors peu de corrélation entre ces paramètres à part un léger biais entre l'âge et le sexe des participants. Il y a également une certaine corrélation entre l'utilisateur et la position de la pause dans le dialogue puisque celle-ci est fixée arbitrairement pour chaque utilisateur et ne varie donc pas durant le dialogue.

Analyse univariable et importance des paramètres des tours de parole

Afin d'estimer quels paramètres du tour de parole mettre en avant dans notre simulation, nous effectuons une analyse univariable (test χ^2) ainsi qu'un calcul de l'importance des caractéristiques pour les modèles *ExtraTrees* et *Random Forest* de scikit-learn selon une démarche similaire à celle réalisée pour l'expérience 1 (cf section 7.1.4). A cette fin, la métrique numérique du nudges pour

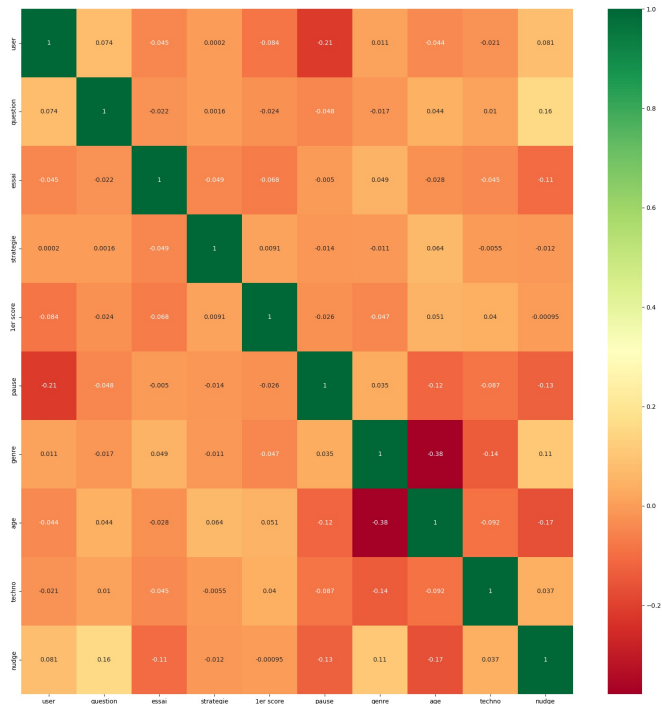
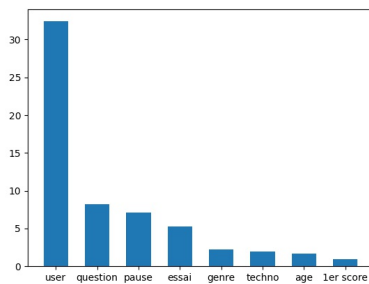


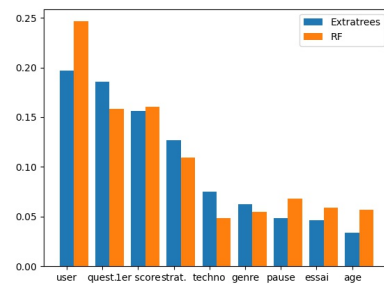
FIGURE 10.1 : Matrice de corrélation des principaux paramètres d'un tour de parole

chaque tour de parole est utilisée comme variable cible et est labellisée par -1, 0, 1, 2 respectivement selon que le score soit négatif, nul, de 1 ou de plus de 1. Les classifieurs RF et ExtraTrees sont alors entraînés en prenant comme caractéristiques en entrée les mêmes paramètres que ceux de la section 10.1.2 précédente. Les résultats sont donnés dans la figure 10.2.

Nous pouvons alors observer que le succès des nudges semblent bien dépendant de l'utilisateur, puisqu'il s'agit de la variable la plus importante selon chaque modèle (cf figure 10.2, Figure (a) et (b)). Le choix de la stratégie en revanche semble moins important pour prédire le succès d'un nudge que la question pour laquelle elle est utilisée. Cela souligne la forte influence du contexte dans lequel un nudge est employé.



(a) Analyse Univariable (χ^2)



(b) Importance (%) selon les modèles ExtraTrees et RF

FIGURE 10.2 : Importance des paramètres pour prédire le score de nudges au tour de parole

Modèle simplifié de l'utilisateur

Comme il est difficile de simuler dans notre cas l'ensemble des facteurs amenant l'utilisateur à être influencé ou non, une solution est de récupérer l'acte de dialogue de ces réponses directement depuis le corpus. Nous entendons par corpus ici l'ensemble des tours de paroles de l'expérience 2, dont l'acte de dialogue est extrait directement à l'aide des modules d'entrée (ASR, NLU). La taille du corpus ne permettant pas un entraînement suffisant, nous simulons alors le déroulé de l'expérience pour chaque utilisateur et récupérons la réponse utilisateur dans le corpus associée à un état du dialogue similaire à l'état actuel simulé.

Pour évaluer la faisabilité d'un apprentissage rapide sur des utilisateurs en ligne, nous simulons donc le comportement d'un utilisateur de manière simplifiée.

Focus sur les stratégies :

Notre objectif étant de tester l'influence de stratégies selon l'utilisateur, nous faisons l'approximation qu'un utilisateur répondra de manière similaire pour une même stratégie sur une question différente afin de prendre en compte les nombreux cas non représentés du corpus. 8 utilisateurs ayant répondu de manière pertinente à chaque stratégie (présence de succès et d'échecs de nudges) ont alors servi d'exemples pour un modèle utilisateur. Une randomisation partielle du score initial pour chaque question du corpus est également ajoutée pour prendre en compte le taux d'erreur du module SLU. En s'appuyant sur l'importance des paramètres prédictifs du succès d'un nudge (cf figure 10.2), l'utilisa-

teur simulé récupérerait alors dans le corpus une réponse privilégiant dans cet ordre les paramètres suivants : la stratégie utilisée, le numéro de la question, la similarité avec le score initial de la base de données et enfin le nombre d'essais pour cette question. Si nous suivons ici un ordre de priorité déterministe, plusieurs calculs de similarité seraient néanmoins possibles. Nous cherchons également ici à privilégier une adaptation individuelle mais une approche à partir de clusters pourraient être envisagée par exemple.

Focus sur la longueur du dialogue :

L'apprentissage du DM se fait donc en entraînant un algorithme de RL sur un modèle utilisateur simplifié renvoyant l'acte de dialogue du corpus le plus similaire à l'état du dialogue simulé. Une autre solution pour favoriser la convergence du modèle RL serait de diviser chaque dialogue en 5 mini dialogues, chacun correspondant à une question de nudge. L'avancement du dialogue (numéro de question) étant compris dans l'état du dialogue, cela permettrait de multiplier artificiellement par 5 le nombre d'épisodes tout en divisant leur longueur du même chiffre. Cela offrirait une plus grande robustesse également en augmentant la variance dans l'enchaînement des épisodes. Dans la pratique néanmoins, l'apprentissage se révèle moins stable.

10.2 Evaluation sur un modèle utilisateur pour l'apprentissage en ligne

10.2.1 Simulation de l'apprentissage du DM en ligne

Pour simuler l'apprentissage du DM en ligne, nous nous intéressons aux variables suivantes :

- Le type de récompense : selon la métrique utilisée (binaire ou numérique) (cf section 10.1.1). Nous ne considérons pas les récompenses additionnelles ici pour davantage de visibilité sur les calculs des récompenses obtenues.
- Le type d'algorithmes : 2 algorithmes de RL en particulier ont été étudiés ici. L'un est l'algorithme KTD-Q [[Geist and Pietquin, 2010](#)] avec une modélisation paramétrique (fonctions RBF) et l'autre l'algorithme DQN [[Mnih et al., 2013](#)] avec une modélisation neuronale.
- l'optimisation du corpus d'apprentissage (nombre d'utilisateurs, longueur du dialogue) : Le modèle utilisateur simplifié présenté section 10.1.2 est

gardé pour les simulations suivantes pour des soucis de stabilité.

Modèle RL

Deux algorithmes de RL ont été étudiés ici afin de déterminer lequel intégrer au système pour l'apprentissage en ligne de ce dernier sur de véritables utilisateurs selon sa vitesse de convergence et sa stabilité. Ces algorithmes étant préentraînés sur les données du corpus, ils devaient être tous deux *off-policy*.

KTD-Q

L'algorithme KTD-Q offre l'avantage d'être efficace par échantillon et de prendre en compte la non-stationnarité des données. Il peut néanmoins souffrir d'instabilité en particulier lorsqu'une modélisation neuronale est utilisée [Daubigney et al., 2012]. Nous avons donc choisi une modélisation paramétrique à l'aide de fonctions RBF pour prendre l'état du dialogue en entrée. Un critère de décision bonus-glouton est utilisé. Les hyperparamètres du modèle sont les mêmes que ceux de l'article [Daubigney et al., 2012].

DQN :

L'algorithme DQN est un algorithme classique de RL [Mnih et al., 2013], utilisé notamment au début sur des tâches de jeux (comme Atari) et qui a depuis connu de nombreuses améliorations. Il utilise une modélisation neuronale pour représenter les valeurs de la fonction Q (au lieu de la représentation tabulaire limitée dimensionnellement du Q-learning). Notre modèle neuronal comporte 2 couches cachées (taille 32 et 64) avec fonction d'activation ReLu et un optimizer Adam avec un taux d'apprentissage de 0.001.

Nous utilisons ici un Double DQN, c'est-à-dire que 2 réseaux sont utilisés : l'un pour déterminer l'action (critère ϵ -glouton) et l'autre pour estimer la valeur Q associée. Cela permet alors d'éviter des valeurs Q trop optimistes. Le 2ème réseau reste ainsi en mode évaluation et ses poids sont mis à jour avec ceux du premier réseau réalisant l'exploration tous les 10 épisodes (hyperparamètre fixé par la simulation). Le modèle intègre également la méthode *experience replay* offrant plus de robustesse (taille des batch : 32). L'hyperparamètre γ est fixé à 0.2 pour encourager une convergence rapide et la recherche de récompenses à court terme. L'exploration se fait selon le critère ϵ -glouton décrémente exponentiellement de $\epsilon_0 = 0.9$ à $\epsilon_1 = 0.01$ selon la formule :

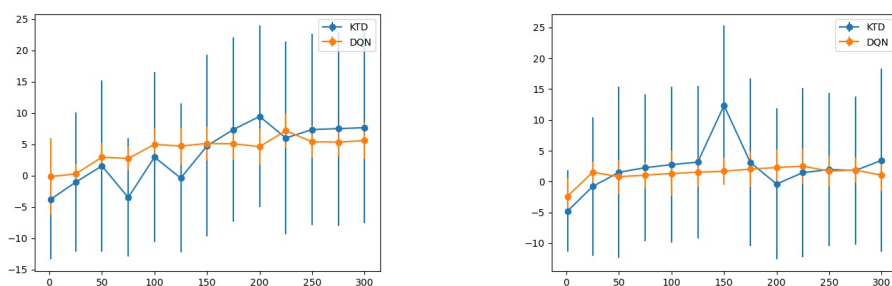
$$\epsilon = \epsilon_1 + (\epsilon_0 - \epsilon_1) * \exp(-n_{\text{etapes}} * \epsilon_d)$$

où n_{etapes} correspond au nombre d'actions déjà réalisées et ϵ_d est un hyperparamètre contrôlant

Résultats

Pour évaluer la faisabilité d'un entraînement en ligne du DM, nous évaluons les 2 modèles de RL sur notre modèle utilisateur selon 2 types de récompenses (cf figure 10.3). La récompense binaire semble alors donner les résultats les plus stables. La figure montre notamment qu'un apprentissage plus personnalisé serait possible en quelques centaines de dialogues. L'algorithme KTD semble donner les meilleures performances mais se révèle moins stable que l'algorithme DQN.

Les modèles convergent notamment vers une récompense de 5 ou plus pour la figure 10.3 (a). Cela signifie qu'un peu plus de 2 nudges sont réussis par dialogue, ce qui correspond à un succès d'au moins 40%. Il semble ainsi plus performant qu'une utilisation non adaptative des stratégies de nudges (plutôt de 30% en moyenne, cf section 7.9), ce qui validerait notre Hypothèse 3 sur les nudges. Ces tendances gagneraient cependant à être vérifiées sur de véritables utilisateurs en ligne avec davantage de participants.



(a) Récompense pour succès binaire

(b) Récompense (succès numérique)

FIGURE 10.3 : Evolution des récompenses cumulées pour 2 algorithmes (moyennée sur 10 simulations)

En conclusion, le succès d'un nudge semble bien dépendre de l'utilisateur mais pourrait dépendre davantage du contexte de son utilisation que de la stratégie de nudge employée. Un apprentissage par renforcement est alors possible pour adapter les stratégies de nudge à l'utilisateur même s'il peut souffrir d'instabilité. Cet apprentissage permettrait alors un nudge plus efficace qu'une utilisation non personnalisée. Une évaluation sur une grande quantité d'utilisateurs permettrait enfin de confirmer l'efficacité d'une adaptation personnalisée.

Conclusion

Chapitre 11

Conclusions

Dans cette partie, nous concluons sur l'agent capable de nudges proposé et sur son efficacité à s'adapter à l'utilisateur. La section 11.1 conclut sur les questions de recherche soulevées surant la thèse, la section 11.2 se focalise sur les conclusions tirées à propose des nudges étudiés et discute des implications éthiques dans l'étude des nudges. Enfin, la section 11.3 conclut sur le système proposé et discute des perspectives de recherche après la thèse.

11.1 Réponses aux questions de recherche

Comment étudier les nudges linguistiques et leur personnalisation dans le dialogue ? Comment tester l'impact de la représentation de l'agent dans l'interaction sur l'efficacité des nudges ?

Nos conclusions au sujet des nudges sont toutes regroupées dans la section suivante.

Comment créer un environnement de recherche pour tester les nudges ?

Nous proposons donc comme environnement de recherche un système de dialogue *retrieval-based*, orienté tâche, permettant une évaluation numérique des nudges sur des questions d'écologie.

Les avantages de cette approche sont :

- permettre une évaluation numérique et objective des nudges
- tester en priorité l'impact du choix des stratégies de nudges
- engager les participants sur un sujet ordinaire mais politique et tester leur propension à modifier leurs intentions sur ce sujet
- évaluer l'impact de répétitions de nudges sur les participants

Les défauts de cette approche étaient cependant :

- Le grand nombre de répétitions pouvaient créer un agacement chez certains utilisateurs et participer à créer un effet de réactance
- Le caractère très spécifique du domaine choisi pour la tâche peut rendre difficile la généralisation des résultats à d'autres domaines
- L'aspect conversationnel de l'agent n'est pas évalué

Sur quelles données entraîner le système ?

Cette thèse propose 2 expériences pour collecter des données orales spécifiques aux nudges :

- une expérience sur le terrain (école primaire), avec un dispositif de WoZ, auprès d'une population *sensible* d'enfants
- une expérience automatique en ligne, auprès d'internautes dans un environnement donc non contrôlé

Ces 2 expériences permettent d'illustrer les avantages et difficultés rencontrés dans leurs 2 cadres différents. Nous montrons alors comment un apprentissage est possible à partir de celles-ci pour :

- prédire le succès d'un nudge durant l'expérience (section 7.1.4)
- aider l'apprentissage d'un module de détection émotionnelle (chapitre 9)
- apprendre des stratégies de nudges adaptées (chapitre 10)

Ces données gagneraient néanmoins à être augmentées afin de pouvoir correctement confirmer les tendances observées.

Quel apprentissage est possible avec peu de données ?

Deux mécanismes de notre système reposaient sur un apprentissage avec peu de données :

- l'apprentissage des stratégies de nudge du DM : Nous montrons qu'il est possible à l'aide d'algorithmes efficaces par échantillon (cf chapitre 10).

- l'apprentissage *few-shot* du module de détection émotionnel : Cet apprentissage nécessite des connaissances a priori sur les classes émotionnelles, réalisées à partir d'un préentraînement d'un réseau siamois. Les performances de ce module restent cependant très modestes en raison probablement des données bruitées collectées sur le site, peut-être trop différentes des données utilisées pour le préentraînement, enregistrées dans un environnement maîtrisé.

Comment évaluer l'influence du nudge ?

Afin de mesurer numériquement et quantitativement l'impact du changement de comportement ou de décision que génère un nudge, nous avons utilisé des questions faisant intervenir le choix explicite d'un score, dont nous pouvons analyser l'évolution en introduisant des métriques simples. De telles métriques ont été utilisées pour les deux expériences. Les résultats de l'expérience 1 ont également permis de révéler certains paramètres contextuels et linguistiques pertinents pour l'analyse des nudges.

Le changement émotionnel de l'utilisateur suite à un nudge a également été pris en compte dans l'état du dialogue et sous forme de faibles récompenses implicites dans l'apprentissage par renforcement.

Conclusion sur la prise en compte des émotions pour l'apprentissage par renforcement

Comment évaluer les systèmes de nudges linguistiques ?

Pour évaluer le système de nudge, un simulateur d'utilisateurs est utilisé, modélisé à partir des interactions collectées durant l'expérience 2. Le système est entraîné par apprentissage par renforcement sur ces interactions mettant en scène une série de questions de nudges sur l'écologie, chaque question permettant de tester l'influence d'une stratégie choisie par le système.

Est-ce éthique de faire des systèmes qui nudgent ?

Nous montrons dans cette thèse la démarche éthique suivie pour étudier les nudges, qui a déjà permis de sensibiliser les participants sur leur influence et les risques potentiels de leur utilisation. Cet aspect est discuté davantage dans la section 11.2.3.

11.2 Bilan sur les nudges

11.2.1 Vérification des hypothèses de travail

Les expériences réalisés, ainsi que l'évaluation ensuite du système de dialogue a permis de confirmer les hypothèses avancées au sujet des nudges :

1. L'expérience 1 semble apporter une réponse claire. Auprès de populations sensibles comme les enfants, celles-ci sont davantage influencés par une machine (enceinte ou robot) que par un humain.
2. Cette hypothèse est vérifiée dans l'expérience 2. L'utilisation d'un nudge linguistique semble plus efficace pour changer le choix d'un utilisateur lors d'une interaction avec un agent virtuel que la simple répétition d'une question. Ces résultats sont observés pour 4 stratégies de nudges linguistiques contextualisées au domaine de l'interaction : le comportement écologique. Cette tendance bénéficierait d'être confirmée en étendant l'analyse à d'autres types de nudges linguistiques mais également d'interface (cf 2.3.3), celle-ci pouvant constituer un facteur important, ainsi qu'à d'autres domaines.
3. Un apprentissage personnalisé est possible et serait plus performant, selon les utilisateurs simulés à partir des données collectées de l'expérience 2.
4. L'expérience 1 présente des résultats encourageants. Certains paramètres paralinguistiques et contextuels se révèlent de bons prédicteurs de nudge même si d'autres études aideraient à confirmer ces résultats.

Les 2 expériences présentées Chapitre 7 ont également permis de tester différents agents et représentations pour ceux-ci. L'agent imaginaire utilisé dans l'expérience 2 semblerait alors moins efficace pour nudger que les agents machines (enceinte et robot) physiques. Cela pourrait être dû à un engagement différent des participants dans les 2 expériences. En effet, la moitié des inscrits à l'expérience 2 ne la finissent pas tandis que l'encadrement sur le terrain de l'expérience a permis d'accompagner chaque enfant jusqu'à la fin de l'expérience dans de bonnes conditions. Néanmoins, un grand nombre de facteurs séparent ces 2 expériences et pourraient influencer ces résultats : la population cible, le domaine, l'environnement.

11.2.2 Limites des nudges

Les travaux dans cette thèse ont permis d'observer des effets de réactance en particulier lors de la 2ème expérience. Si ces effets restent minoritaires, soit environ 20% des nudges ayant influencé (cf Table 7.9), ils ne sont néanmoins pas négligeables. Cet effet a l'air d'avoir été amplifié par le caractère répétitif du protocole.

Les protocoles proposés font également intervenir d'autres limites des nudges qui sont cependant plus difficiles à mettre en évidence :

- *des préférences ou caractéristiques personnelles fortes* : Le régime végétarien de certains participants a pu créer un vrai biais sur leur inclinaison à se faire influencer. En pratique, le nombre de végétarien était cependant trop faible pour obtenir des résultats significatifs. Un autre critère pouvant introduire un biais était l'âge des participants (les plus jeunes n'ayant jamais conduit par exemple, ils avaient a priori un rapport aux voitures différents d'autres participants plus âgés).
- *un manque de compréhension des intentions du designer du nudge* : Les nudges étant des incitations 'douces' par définition, leur utilisation peut ne pas être remarquée ou être mal interprétée. Tous les candidats ne semblaient pas prendre conscience dans une même mesure de la répétition des questions par exemple. Nous avons pour cela borné le plus possible l'expression des nudges mais une formulation davantage personnalisée et flexible pourrait réduire des potentiels manques de compréhension.
- *un effet à courte durée* : Les nudges dans ces expériences n'ont pas d'objectifs sur la durée puisqu'ils servent davantage de sensibilisation. Dans le but de changer durablement des habitudes écologiques, d'autres paramètres devraient alors être pris en compte. Des stratégies de *pré-engagement*, d'*intentions de mise en œuvre* et de *rappels* (cf section 2.3.4) seraient alors préférables. L'efficacité des nudges présentés ici est ainsi dépendante de la tâche sélectionnée. Adapter l'ensemble de stratégies de nudge parmi lesquelles choisir selon la tâche pourrait alors être pertinent dans le cadre d'un système de dialogue multi-tâche.

11.2.3 Utilisation éthique des nudges

Si le nudge est un concept apparu à l'origine en économie comportementale, cette thèse l'étudie davantage dans une perspective de machine learning. Elle fait cependant partie du projet interdisciplinaire Bad Nudge Bad Robot de l'Institut DATAIA rassemblant informaticiens, linguistes, économistes et juristes. Une

collaboration avec une équipe d'économistes du RITM (expérience 1) a ainsi offert des approches différentes, a ancré l'expérience dans un contexte d'économie comportementale connu et a permis d'insister sur les contraintes éthiques de l'expérience 1, celles-ci devant ainsi satisfaire ceux des 2 domaines.

Respect des données

Ces travaux se placent dans le cadre général de l'éthique du numérique, dont l'une des thématiques phares est la protection des données à la fois dans la conception et la mise en pratique d'outils numériques. Notre étude ayant pour cadre le projet Bad Nudge Bad Robot, cherchant à étudier l'influence des nudges notamment sur des populations sensibles, une attention particulière a été portée au respect et à l'anonymat des données des participants.

La première expérience a ainsi été soumise à un comité d'éthique qui a alors validé le protocole de l'expérience. Un consentement parental a notamment été recueilli. Le stockage de ces données a également été anonymisé et les participants ont été répartis si possible de manière aléatoire dans chacun des groupes de l'expérience.

Pour les expériences suivantes conduites sur le site, la consultation d'une mention légale indiquant la RGPD suivie était nécessaire pour pouvoir participer. Les données collectées restaient également anonymes.

Sensibilisation

Les démarches de sensibilisation durant cette thèse incluent :

- des séances de démonstration et d'explications en classe au sujet de l'influence des machines notamment.
- la communication des résultats de l'expérience à distance (à travers le mail renseigné)

Dans les 2 cas, nous avons testé l'influence de nudges dans un contexte où les participants ignoraient le but véritable de l'interaction. Cela soulève néanmoins le besoin de pouvoir détecter lorsqu'un système tente d'influencer un utilisateur à son insu. Des propositions pour réguler l'utilisation de nudges et les risques liés à l'utilisation de systèmes automatiques sont en cours à ce sujet, comme la loi européenne EU AI ¹, la première loi sur l'IA par un régulateur majeur.

¹<https://oecd.ai/fr/wonk/ai-act-manipulation-methods>

11.3 Perspectives et Travaux futurs

Cette thèse propose une première modélisation d'un agent capable de nudges dans le dialogue oral. Nous avons fait le choix pour cela d'un apprentissage adaptatif personnalisé à partir de données collectées spécifiques aux nudges qui nous ont permis de valider les hypothèses de notre étude. Il s'agit néanmoins d'un sujet encore ouvert dont les utilisations en informatique bénéficieraient à être encore étudiées.

11.3.1 Perspectives à court terme

Différentes approches sont possibles pour améliorer le système proposé en termes de performance, robustesse et transparence.

Performance

La performance des différents modules pourrait être améliorée par :

- le *fine-tuning* de modèles *transformers* préentraînés pour les modules d'ASR et de détection émotionnelle. Si les tests réalisés ont montré l'importance de l'information audio dans la détection d'émotions sur notre tâche ; le module de détection émotionnelle pourrait également profiter d'un modèle *transformers* préentraîné pour capturer l'information sémantique. Il pourrait alors être entraîné à partir de la transcription ou directement à partir de l'audio comme Wav2vec2 [Baevski et al., 2020] (ou une version plus légère comme Audio-AIBERT [Chi et al., 2020]). L'accès à davantage de données spontanées annotées rendraient alors possible un *fine-tuning* offrant de meilleures performances. En effet, l'entraînement sur des données actées rend l'apprentissage difficile sur les données spontanées.
- davantage préprocesser les fichiers audio enregistrés par le site : reconnaissance du locuteur (pour des locuteurs multiples), reconnaissance de voix humaine (limiter automatiquement l'impact des silences et bruits)
- augmenter le nombre d'exemples par émotion disponibles pour l'entraînement *few-shot* d'émotions.
- l'amélioration de la vitesse d'apprentissage par renforcement avec de l'*active learning* par exemple, en indiquant les utilisateurs les plus pertinents (interaction sans perturbations, utilisateurs engagés) en début d'apprentissage.

Robustesse

Les approches suivantes peuvent augmenter la robustesse de l'apprentissage :

- l'ajout de feedbacks humains : En plus d'accélérer l'apprentissage, l'utilisation de feedbacks humains pourrait offrir une meilleure estimation de l'effet des nudges a posteriori. Cette approche serait d'autant plus pertinente dans le cas d'un domaine de dialogue plus ouvert où la tâche ne permettrait pas d'estimer directement la modification d'un comportement ou d'une décision.
- utiliser un modèle par groupes d'utilisateur : Pour simplifier l'apprentissage de profils internautes très variés, une solution serait de séparer cet apprentissage en plusieurs modèles en fonction des classes d'utilisateurs. Il serait notamment pertinent de séparer le traitement audio enfant et adulte (en particulier masculin) car leur voix ont fréquences fondamentales assez éloignées en moyenne (cf section 9.3.2).

11.3.2 Poursuite de l'obtention de données

Un des *bottlenecks* évidents de notre étude est le manque de données. S'il s'agit d'une contrainte supplémentaire pour cette thèse (due en partie aux conditions sanitaires du Covid), des travaux suivants pourraient profiter de données supplémentaires qui permettraient notamment de confirmer les résultats présentés. La plateforme en l'état offre déjà cette possibilité.

Bien que certains modules comme le module émotionnel puissent être adaptés au fur et à mesure des utilisateurs, d'autres modules comme l'ASR demeurent fixes. Ces modules pourraient être mis à jour par la suite. Davantage de données annotées aideraient notamment à les fine-tuner, particulièrement les modules suivants : ASR, NLU.

Une manière d'obtenir davantage de données serait par exemple de filtrer des données d'autres domaines proches du domaine des nudges et possédant des termes caractéristiques à notre tâche (comme cela a été fait sur des données conversationnelles de Reddit [[Hung et al., 2022](#)]). Cette approche serait notamment adaptée au fine-tuning d'un modèle de langue *transformers* qui ne nécessite pas de données supervisées. Cela pourrait permettre d'aligner davantage le module NLU à la tâche.

11.3.3 Adaptation à d'autres domaines

Si les principaux biais cognitifs qui influencent les décisions de chacun sont communs à de nombreux contextes, il est clair cependant que le cadre dans

lequel ils interviennent peut jouer un rôle. Dans nos expériences, nous avons borné notre protocole à un cadre très restreint et notre système de dialogue final aborde ainsi seulement certaines questions d'écologie. Néanmoins, il serait intéressant de tenter de reproduire ces résultats sur d'autres domaines. En effet, le contexte d'utilisation d'un nudge, dans notre expérience 2, semble non négligeable dans la prédiction de son succès. La plateforme développée a justement été conçue dans l'optique de faciliter et d'automatiser tout travail similaire avec une collecte automatisée en ligne. Elle permettrait ainsi de varier facilement le domaine du dialogue et plus généralement le protocole de futures expériences.

Étant donné le cadre très général des chatbots actuels, une future piste de recherche serait également d'évaluer un agent capable de nudges dans un système de dialogue à domaine ouvert et de le rendre davantage conversationnel. A cette fin, la mesure du changement de comportement ou de décision pourrait être indépendante de la tâche. Les performances récentes des LLM suggèrent qu'ils pourraient alors être adaptés en filtre sémantique pour estimer la mesure de ce changement.

11.3.4 Axes d'étude à long terme

Cette thèse propose un premier système de dialogue avec apprentissage sur les nudges qui s'est concentré sur la prise en compte d'états mentaux et de certaines informations du profil de l'utilisateur. Différents axes d'études pourraient encore être étudiés dont :

- La génération de la réponse : Afin qu'elle n'influe pas sur les résultats des choix de stratégies de nudges, nous avons choisi une génération *retrieval-based*, les réponses étant reconstituées à partir d'audio pré-enregistrés selon l'état du dialogue estimé. Néanmoins, étant donné le succès récent de large modèles de langues (LLM) davantage alignés aux tâches voulues comme ChatGPT, une perspective future serait de les utiliser pour obtenir une réponse plus dynamique. Cela nécessiterait d'assurer néanmoins un contrôle de la réponse
- La synthèse vocale : L'apport pour les nudges de la variation de la synthèse vocale pourrait également davantage être étudié. Par exemple, le genre, l'intensité de la voix ou la présence d'émotions dans celle-ci pourraient être des pistes futures d'autres dimensions de nudges. Cela pourrait inclure des approches comme le préenregistrement de plusieurs voix, l'ajout d'un filtre vocal, la modification dynamique de l'interface audio, l'entraînement de plusieurs modèles de synthèse vocale pour chaque type de voix, etc.

- La dimension multimodale : Durant cette étude, nous nous sommes concentrés principalement sur le caractère vocal de l'interaction. Un autre axe d'étude serait d'étudier les nudges de manière multimodale aussi bien au niveau des informations perçues (expressions du visage, regard, etc.) qu'au niveau des retours du système. Beaucoup de nudges étant visuels, un système mettant à profit ceux-ci augmenterait probablement en efficacité, même si cela sort de notre champ d'étude de chatbots vocaux.
- L'ajout de caractéristiques de personnalité à l'agent : Nous avons choisi de modéliser un agent non personnifié pour des raisons à la fois éthique et de généralisabilité mais il serait également possible d'adapter la personnalité et la manière de s'exprimer (registre de langues, humour, persona) de celui-ci à l'utilisateur.

11.3.5 Applications réelles

Dans cette thèse, j'ai simulé des nudges linguistiques et étudié leurs effets à travers des expériences très spécifiques montrant l'influence du nudge sur les décisions d'enfants mais aussi d'adultes. Il reste de nombreuses questions de recherche sur les systèmes de dialogue capables de nudges, qui gagneraient à être explorés dans des applications réelles. Par exemple, dans le cas d'une utilisation du robot Moxie, un robot pour aider le développement d'enfants souffrants de troubles neurologiques (tel que l'autisme), comment évaluer l'influence du nudge et de ses dérives possibles ?

Bibliographie

- Hugues Ali Mehenni, Sofiya Kobylanskaya, Ioana Vasilescu, and Laurence Devillers. Nudges with conversational agents and social robots : a first experiment with children at a primary school. *IWSDS*, 2020a.
- Hugues Ali Mehenni, Sofiya Kobylanskaya, Ioana Vasilescu, and Laurence Devillers. Children as candidates to verbal nudging in a human-robot experiment. *SAMIH*, 2020b.
- Amanda D. Angie, Shane Connelly, Ethan P. Waples, and Vykinta Kligyte. The influence of discrete emotions on judgement and decision-making : A meta-analytic review. *Cognition and Emotion*, 25 :8, 1393-1422, 2011. doi : 10.1080/02699931.2010.550751.
- A. Baevski, H. Zhou, A. Mohamed, and M. Auli. Wav2vec 2.0 : A framework for self-supervised learning of speech representations. 2020.
- P. Bard. A diencephalic mechanism for the expression of rage with special reference to the sympathetic nervous system. *American Journal of Physiology*, 84, 490–515., 1928.
- J.A. Bargh, P.M. Gollwitzer, A. Lee-Chai, K. Barndollar, and R. Trötschel. The automated will : nonconscious activation and pursuit of behavioral goals. *Journal of personality and social psychology* 81, 2001.
- A. Batliner, V. Warnke V., E. Nöth, J. Buckow, R. Huber, and M. Nutt. How to label accent position in spontaneous speech automatically with the help of syntactic-prosodic boundary labels. *Technical Report*, 1998.
- A. Batliner, K. Fisher, R. Huber, J. Spilker, and E. Noth. How to find trouble in communication. *Speech Communication*. 40 : p. 117-143, 2003.
- L.E. Baum and T. Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, 37(6), 1554–1563, 1966.

- R. Bellman. A markovian decision process. *In : Journal of Mathematics and Mechanics*, 1957.
- J.F. Benenson, J. Pascoe, and N. Radmore. Children's altruistic behavior in the dictator game. *Evolution and Human Behavior*, 2007.
- P.M. Berry, T. Donneau-Golencer, K. Duong, M.T. Gervasio, B. Peintner, and N. Yorke-Smith. Evaluating user-adaptive systems : Lessons from experiences with a personalized meeting scheduling assistant. *Proceedings of the Twenty-First Conference on Innovative Applications of Artificial Intelligence, July 14-16, 2009, Pasadena, California, USA*, 2009.
- B. Bogert, M. Healy, and J. Tukey. *Symposium in time series analysis*. Chapitre The quefrequency analysis of time series for echoes : cepstrum, pseudo-autovariance, cross-cepstrum and saphe cracking, pages 209-243. John Wiley & Sons, 1963.
- P. Budzianowski, T-H. Wen, B-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, and M. Gasic. Multiwoz - a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, p. 5016–5026*, 2018.
- R. Cadario and P. Chandon. Which healthy eating nudges work best ? a meta-analysis of field experiments. 2019.
- W.B. Cannon. The james-lange theory of emotions : A critical examination and an alternative theory. *The American Journal of Psychology, vol. 39, no. 1/4, pages 106–124*, 1927.
- A. Caraban, E. Karapanos, D. Gonçalves, and P. Campos. 23 ways to nudge : A review of technology-mediated nudging in human-computer interaction. *Proceedings of CHI Conference on Human Factors in Computing Systems, Glasgow, Scotland UK, May, 2019*. doi : <https://doi.org/10.1145/3290605.3300733>.
- N. Carrara. *Reinforcement learning for Dialogue Systems optimization with user adaptation*. PhD thesis, Université Lille Nord-de-France, 2018.
- I. Casanueva, T. Hain, H. Christensen, R. Marxer, and P. Green. Knowledge transfer between speakers for personalised dialogue management. *Proceedings of the SIGDIAL 2015 Conference, pages 12–21, ACL*, 2015.

- I. Casanueva, P. Budzianowski, P-H. Su, N. Mrkšić, T-H. Wen, S. Ultes, L. Rojas-Barahon, S. Young, and M. Gašić. A benchmarking environment for reinforcement learning based task oriented dialogue management. *Deep Reinforcement Learning Symposium, 31st Conference on Neural Information Processing Systems*, 2017.
- S. Chandramohan, M. Geist, F. Lefevre, and O. Pietquin. User simulation in dialogue systems using inverse reinforcement learning. *Interspeech*, 2011.
- Z. Chen, L. Chen, X. Zhou, and K. Yu. Deep reinforcement learning for on-line dialogue state tracking. 2020.
- P-H. Chi, P-H Chung, T-H. Wu, C-C. Hsieh, Y-H. Chen, S-W. Li, and Y-H. Lee. Audio albert : A lite bert for self-supervised learning of audio representation. *IEEE Spoken Language Technology Workshop 2021*, 2020.
- J. Chung, C. Gülçehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *NIPS 2014 Deep Learning and Representation Learning Workshop, Dec*, 2014.
- R.B. Cialdini. *Influence*. Vol. 3. A. Michel Port Harcourt, 1987.
- A. Cockburn, P. Quinn, and C. Gutwin. Examining the peak-end effects of subjective experience. *In Proceedings of the 33rd annual ACM conference on human factors in computing systems. ACM*, 357–366., 2015.
- J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measures, vol. 20, pages 3746*, 1960.
- C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning, 20(3)*, 273–297, 1995.
- G. Cybenko. Approximation by superpositions of a sigmoidal function. *Math. Control Signal Systems, 2* :303–314, 1989.
- Yinpei Dai, Huihua Yu, Yixuan Jiang, Chengguang Tang, Yongbin Li, and Jian Sun. A survey on dialog management : Recent advances and challenges. 2021.
- A. Damasio. *L'erreur de Descartes : la raison des émotions*. New York : Grosset/Putnam, 1994.
- C. Darwin. *The expression of the emotions in man and animals*. Ed. John Murray, London, 1872.

- L. Daubigney, M. Geist, and O. Pietquin. Apprentissage off-policy appliqué à un système de dialogue basé sur les pdmpo. *Actes du 18ème congrès francophone sur la Reconnaissance de Formes et l'Intelligence Artificielle (RFIA 2012)*, 2012.
- Floris den Hengst, Mark Hoogendoorn, Frank van Harmelen, and Joost Bosman. Reinforcement learning for personalized dialogue management. *IEEE/WIC/ACM International Conference on Web Intelligence*, 2019.
- Florisa den Hengst, Eoin Martinob Grua, Alic el Hassouni, and Mark Hoogendoorn. Reinforcement learning for personalization : A systematic literature review. *Data Science*, vol. 3, no. 2, pp. 107-147, 2020.
- Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echevoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. Survey on evaluation methods for dialogue systems. *Artificial Intelligence Review*, 2020.
- T. Deschamps, L. Lamel, and L. Devillers. Investigating transformer encoders and fusion strategies for speech emotion recognition in emergency call center conversations. *ICMI '22 Companion, November 7–11, 2022, Bengaluru, India, 2022 Association for Computing Machinery. ACM ISBN 978-1-4503-9389-8/22/11.*, 2022.
- David DeSteno, Richard E Petty, Derek D Rucker, Duane T Wegener, and Julia Braverman. Discrete emotions and persuasion : the role of emotion-induced expectancies. *Journal of personality and social psychology* 86(1) :43-56, 2004.
- L. Devillers, L. Vidrascu, and L. Lamel. Challenges in real-life emotion annotation and machine learning based detection. *Neural networks*, May ;18(4) :407-22., 2005.
- J. Devlin, M-W. Chang, K. Lee, and K. 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)*, p. 4171–4186, 2019.
- M. Dynel. Isn't it ironic? defining the scope of humorous irony. *Humor* 27, 4 (2014), 619–639, 2014.
- D. Eckles, D. Wightman, C. Carlson, A. Thamrongrattanarit, M. Bastea-Forte, and B.J. Fogg. Social responses in mobile messaging : influence strategies,

- self-disclosure, and source orientation. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM*, 1651–1654, 2009.
- J. Egebark and M. Ekström. Can indifference make the world greener? *Journal of Environmental Economics and Management* 76, 1–13., 2016.
- P. Ekman. *Basic Emotions*. Wiley, New-York, pp. 301–320, 1999.
- C. Etienne. *Apprentissage profond appliqué à la reconnaissance des émotions dans la voix*. PhD thesis, Université Paris-Saclay, 2019.
- F. Eyben and B. Schuller. opensmile : The munich open-source large-scale multimedia feature extractor. *SIGMultimedia Rec.*, vol. 6, no. 4, pages 4–13, January, 2015.
- F. Eyben, K.R. Scherer, B.W. Schuller, J. Sundberg, E. André, C. Busso, L. Devillers, J. Epps, P. Laukka and S.S. Narayanan, and K.P. Truong. The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing. *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pages 190–202, April, 2016.
- H. Fan and M.S. Poole. What is personalization? perspectives on the design and implementation of personalization in information systems. *Journal of Organizational Computing and Electronic Commerce* 16 : (3–4) ((2006)), 179–202, 2006.
- K. Feng and T. Chaspari. Few-shot learning in emotion recognition of spontaneous speech using a siamese neural network with adaptive sample pair formation. *IEEE Transactions on Affective Computing*, 2021.
- E. Ferreira. *Apprentissage automatique en ligne pour un dialogue homme-machine situé*. PhD thesis, Université d’Avignon, 2015.
- M. Gasic and S. Young. Gaussian processes for pomdp-based dialogue manager optimisation. *IEEE Transactions on Audio, Speech and Language Processing*, Jan 2014, 2013.
- M. Geist and O. Pietquin. Kalman temporal differences. *Artificial Intelligence Research* 39(1), 483–532, 2010.
- G. Gordon, S. Spaulding, J.K. Westlund, J.J. Lee, L. Plummer, M. Martinez, M. Das, and C. Breazeal. Affective personalization of a social robot tutor for children’s second language skills. *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

- Gaël Guibon, Matthieu Labeau, H el ene Flamein, Luce Lefeuvre, and Chlo e Clavel. Few-shot emotion recognition in conversation with sequential prototypical networks. *The 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, Nov, Punta Cana, Dominican Republic, 2021.
- K.L. Gwet. Testing the difference of correlated agreement coefficients for statistical significance. *Educational and Psychological Measurement*, Aug;76(4) :609-637., 2016.
- P.G. Hansen and A.M. Jespersen. Nudge and the manipulation of choice : A framework for the responsible use of the nudge approach to behaviour change in public policy. *European Journal of Risk Regulation* 4, 1, 3–28, 2013.
- H.B. Hashemi, A. Asiaee, and R. Kraft. Query intent detection using convolutional neural networks. *ACM*. ISBN 978-1-4503-2138-9, 2016.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Tech Report*, 2015.
- Jeff Heaton. *Introduction to Neural Networks for Java, 2nd Edition*. Heaton Research, Inc., 2nd edition, 2008. ISBN 1604390085.
- C. Heilmann. Success conditions for nudges : a methodological critique of libertarian paternalism. *European Journal for Philosophy of Science* 4 (1) :1-20, 2014.
- M. Henderson, I. Vulic, D. Gerz, I. Casanueva, P. Budzianowski, S. Coope, G. Spithourakis, T-H. Wen, N. Mrksic, and P-H. Su. Training neural response selection for task-oriented dialogue systems. *ACL 2019*, 2019.
- C.A. Hidalgo. *How Humans judge machines*. MIT Press, 2021.
- S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8) :1735–1780, 1997.
- R.A. Howard. *Dynamic Programming and Markov Processes*. MIT Press, 1960.
- Y. Hu, C-H. Lee, T. Xie, T. Yu, N.A. Smith, and M. Ostendorf. In-context learning for few-shot dialogue state tracking. *EMNLP, Abu Dhabi, United Arab Emirates, Dec*, p. 2627–2643, 2022.
- Chia-Chien Hung, Anne Lauscher, Simone Paolo Ponzetto, and Goran Glavas. Ds-tod : Efficient domain specialization for task oriented dialog. *ACL 2022*, 2022.

- A.J. Hunt and A.W. Black. Unit selection in a concatenative speech synthesis system using a large speech database. *ICASSP*, 1996.
- Fatimah Ishowo-Oloko, Jean-François Bonnefon, Zakariyah Soroye, Jacob Crandall, Iyad Rahwan, and Talal Rahwan. Behavioural evidence for a transparency–efficiency tradeoff in human–machine cooperation. *Nature Machine Intelligence*, vol. 1, November, p. 517–521, 2019.
- W. James. What is an emotion? *Mind*, Apr., Vol. 9, No. 34, pp. 188-205, 1884.
- D. Kahneman and P. Egan. *Thinking, fast and slow*. 2011.
- Y. Kammerer and P. Gerjets. The role of search result position and source trustworthiness in the selection of web search results when using a list or a grid interface. *International Journal of Human-Computer Interaction* 30, 3(2014), 177–191., 2014.
- Maurits Kaptein, Joyca Lacroix, and Privender Saini. Nudge for deliberativeness : How interface features influence online discourse. *In Persuasive Technology*, 2010. doi : 10.1007/978-3-642-13226-1_11.
- Diederik P. Kingma and Jimmy Lei Ba. Adam : a method for stochastic optimization. *ICLR 2015*, 2015.
- G.A. Kleef, H. Berg, and M.W. Heerdink. The persuasive power of emotions : Effects of emotional expressions on attitude formation and change. *Journal of Applied Psychology*, 100, 4, 1124, 2015.
- K. Krippendorff. Computing krippendorff ‘s alpha-reliability. 2011.
- O. Kuchaiev and B. Ginsburg. Factorization tricks for lstm networks. *ICLR 2017 Workshop*, 2017.
- J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. *ICML*, 2001.
- C. Lange. On emotions : A psycho-physiological study. 1885.
- R.S. Lazarus. Emotions and adaptation : Conceptual and empirical relations. *Nebraska Symposium on Motivation*, 16, 175–266, 1968.
- Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoît Crabbé, Laurent Besacier, and Didier Schwab. Flaubert : Unsupervised language model pre-training for french. *LREC 2020*, 2020.

- Lihong Li, Jason D. Williams, and Suhrid Balakrishnan. Reinforcement learning for dialog management using least-squares policy iteration and fast feature selection. *Interspeech, Brighton, UK*, 2009.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta : A robustly optimized bert pretraining approach. 2019.
- S. Lukin, P. Anand, M. Walker, and S. Whittaker. Argument strength is in the eye of the beholder : Audience effects in persuasion. *In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers, volume 1, pages 742–753*, 2017.
- T. Mahmood, G. Mujtaba, and A. Venturini. Dynamic personalization in conversational recommender systems. *Inf Syst E-Bus Manage 12*, 213–238, 2014.
- Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT : a tasty french language model. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020. doi : 10.18653/v1/2020.acl-main.645.
- Gonçalo S. Martins, Luís Santos, and Jorge Dias. User-adaptive interaction in social robots : A survey focusing on non-physical interaction. *In : I. J. Social Robotics 11.1, pp. 185–205*, 2019.
- R.M. Mehmood and H.J. Emotion classification of eeg brain signal using svm and knn. *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2015.
- Sanju Menon, Weiyu Zhang, and Simon T. Perrault. Nudge for deliberativeness : How interface features influence online discourse. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, April, Pages 1–13*, 2020.
- M. Minsky. Steps toward artificial intelligence. *IRE, MIT*, 1961.
- V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. *NIPS Deep Learning Workshop 2013*, 2013.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda

- Askill, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. 2022.
- Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2231–2240, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi : 10.18653/v1/D17-1237. URL <https://aclanthology.org/D17-1237>.
- Baolin Peng, Xiujun Li, Jianfeng Gao, Jingjing Liu, Kam-Fai Wong, and Shang-Yu Su. Deep dyna-q : Integrating planning for task-completion dialogue policy learning. *acl 2018*, 2018.
- R.E. Petty and J.T. Cacioppo. The elaboration likelihood model of persuasion. In *Communication and persuasion*. Springer, 1–24, 1986.
- R. Plutchik. *Approaches to emotion*. Chapter : A General Psychoevolutionary Theory. Erlbaum, Hillsdale, NJ, 1984.
- D. Premack and G. Woodruff. Does the chimpanzee have a theory of mind? *Behavioural and Brain Sciences*, 1 : 515-526, 1978.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. 2022.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR 2020*, 21(140) :1-67, 2020.
- I. Rahwan, M. Cebrian, N. Obradovich, J. Bongard, JF.Bonnefon, C. Breazeal, JW. Crandall, NA. Christakis, ID. Couzin, MO. Jackson, NR. Jennings, E. Kamar, IM. Kloumann, H. Larochelle, D. Lazer, R. McElreath, A. Mislove, DC. Parkes, A. Pentland, ME. Roberts, A. Shariff, JB. Tenenbaum, and M. Wellman. Machine behaviour. *Nature*, 568 :477–486, 2019.
- C. Raymond and G. Riccardi. Generative and discriminative algorithms for spoken language understanding. *INTERSPEECH*, 1605–1608, 2007.
- Nils Reimers and Iryna Gurevych. Sentence-bert : Sentence embeddings using siamese bert-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

- Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. *EMNLP 2020*, 2020.
- M. Riou, B. Jabaian, S. Huet, and F. Lefèvre. Joint on-line learning of a zero-shot spoken semantic parser and a reinforcement learning dialogue manager. *Journal of Personality and Social Psychology* 76(5) :805-19, 2018.
- H. Ritschel. *Real-Time Generation and Adaptation of Social Companion Robot Behaviors*. PhD thesis, Augsburg University, 2022.
- J. Rong, G. Li, and Y-P. P. Chen. Acoustic feature selection for automatic emotion recognition from speech. *Information Processing and Management*, vol. 45, pages 315-328, 2009.
- Nicholas Roy, Joelle Pineau, and Sebastian Thrun. Spoken dialogue management using probabilistic reasoning. *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*. Hong Kong, Oct, 2000.
- R. Ruiz, P.P. de Hugues, and C. Legros. Acoustic feature selection for automatic emotion recognition from speech. *Acta Acustica United with Acustica*, vol. 96, n°3, pages 567-579, 2010.
- J. Russel. Reading emotions from and into faces : Resurrecting a dimensional-contextual perspective. *Cambridge University Press, U.K.*, pp. 295–360, 1997.
- J.A. Russel and L. Feldman-Barrett. Core affect, prototypical emotional episodes, and other things called emotion : Dissecting the elephant. *Journal of Personality and Social Psychology* 76(5) :805-19, 1999.
- A. Satt, S. Rozenberg, and R. Hoory. Efficient emotion recognition from speech using deep learning on spectrograms. *INTERSPEECH 2017, August 20–24, Stockholm, Sweden*, 2017.
- S. Schachter and J.E. Singer. Cognitive, social and physiological determinants of emotional state. *Psychological Review*, 69, 379-399, 1962.
- J. Schatzmann, B. Thomson, K. Weilhammer, H. Ye, and S. Young. Agenda-based user simulation for bootstrapping a pomdp dialogue system. *The Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 149–152, 2007.
- S. Scher. L'expérience ordinaire de la méfiance. *Tracés, Revue de sciences humaines*, p. 151-167, 2016.

- K.R. Scherer. Vocal affect expression : a review and a model for future research. *Psychological bulletin*, vol. 99 2, pages 143–65, 1986.
- K.R. Scherer. *Affect bursts*. chapitre Emotions, pages 161-193. Hillsdale, NJ : Lawrence Erlbaum, 1994.
- K.R. Scherer and M. Peper. Psychological theories of emotion and neuropsychological research. *Handbook of Neuropsychology*, vol. 5 (*Emotional behavior and its disorders*), pages 1748, 2001.
- B. Schuller, G. Rigoll, and M. Lang. Hidden markov model-based speech emotion recognition. *IEEE, International Conference on Multimedia and Expo. ICME '03. Proceedings (Cat. No.03TH8698)*, 2003.
- B. Schuller, R. Zaccarelli, N. Rollet, and L. Devillers. Cinemo - a french spoken language resource for complex emotions : facts and baselines. *In LREC, Valetta, Malta*, 2010.
- J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. 2017.
- I. Shahin, A.B. Nassif, and S. Hansa. Emotion recognition using hybrid gaussian mixture model and deep neural network. *IEEE*, 2019.
- Weiyang Shi and Zhou Yu. Sentiment adaptive end-to-end dialog systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1509–1519, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi : 10.18653/v1/P18-1140. URL <https://aclanthology.org/P18-1140>.
- Weiyang Shi, Tiancheng Zhao, and Zhou Yu. Unsupervised dialog structure learning. 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)*, pages 1797–1807, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi : 10.18653/v1/N19-1178. URL <https://aclanthology.org/N19-1178>.
- David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dhharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. 2017.

- P. Slovic, M.L. Finucane, E. Peters, and D.G. MacGregor. The affect heuristic. *European journal of operational research* 177, 3 (2007), 1333–1352, 2007.
- E.J. Sondik. *The Optimal Control of Partially Observable Markov Processes*. PhD thesis, Stanford University, 1971.
- P-H. Su, P. Budzianowski, S. Ultes, M. Gasic, and S. Young. Sample-efficient actor-critic reinforcement learning with supervised data for dialogue management. 2017.
- S. Sullivan and T. Ruffman. Social understanding : How does it fare with advancing years ? *British Journal of Psychology* ; 95 : 1-18, 2004.
- F. Sung, Y. Yang, L. Zhang, T. Xiang, P.H. Torr, and T.M. Hospedales. Learning to compare : Relation network for few-shot learning. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208., 2018.
- C. R. Sunstein. Nudging : A very short guide. *Journal of Consumer Policy* 37, 583–588, 2014.
- C. R. Sunstein. Nudges that fail. *Discussion Paper No. 879, Harvard Law School*, 2016.
- R.S. Sutton and A.G. Barto. *Reinforcement Learning : An Introduction*. MIT Press, 2014.
- A. Syssau and C. Monnier. L'influence de la valence émotionnelle positive des mots sur la mémoire des enfants. 2012.
- M. Tahon. *Analyse acoustique de la voix émotionnelle de locuteurs lors d'une interaction humain-robot*. PhD thesis, Université Paris-Sud, 2012.
- M. Tan and Q.V. Le. Efficientnet : Rethinking model scaling for convolutional neural networks. *ICML 2019*, 2019.
- R. H. Thaler and C.R. Sunstein. *Nudge : Improving decisions about health, wealth, and happiness*. 2008.
- R.H. Thaler and C. R. Sunstein. Nudge, not sludge. *Science*, 2019.
- D.R. Traum. Speech acts for dialogue agents. *Foundation of rational agency*, 169–201. Springer, 1999.

- G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M. Nicolaou, B. Schuller, and S. Zafeiriou. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, pages 5200–5204, 03 2016*, 2016.
- I. Tussyadiah and G. Miller. Nudged by a robot. *Annals of Tourism Research*, 2019.
- A. Tversky and D. Kahneman. Judgment under uncertainty : Heuristics and biases. *Science*, 185, 1124-1131, 1974.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. 2017. doi : 10.48550/ARXIV.1706.03762.
- Oriol Vinyals and Quoc Le. A neural conversational model. 2015.
- T. Vogt and E. Andre. Comparing feature sets for acted and spontaneous speech in view of automatic emotion recognition. *ICME*, 2005.
- Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, and Alicia Abella. Paradise : a framework for evaluating spoken dialogue agents. 1997.
- Weikang Wang, Jiajun Zhang, Han Zhang, Mei-Yuh Hwang, Chengqing Zong, and Zhifei Li. A teacher-student framework for maintainable dialog manager. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3803–3812.*, 2018.
- Xuwei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. Persuasion for good : Towards a personalized persuasive dialogue system for social good. *ACL*, 2019.
- Y. Wang, P.G. Leon, A. Acquisti, L.F. Cranor, A. Forget, and N. Sadeh. A field trial of privacy nudges for facebook. *In Proceedings of the SIGCHI conference on human factors in computing systems. ACM, 2367–2376.*, 2014.
- Y. Wang, G. Lucas, P. Khooshabeh, C. De Melo, and J. Gratch. Effects of emotional expressions on persuasion. *Social Influence* 10, 4, 236–249, 2015.
- C.J.C.H. Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge, England, 1989.

- Klaus Weber, Kathrin Janowski, Niklas Rach, Katharina Weitz, Wolfgang Minker, Stefan Ultes, and Elisabeth Andre. Predicting persuasive effectiveness for multimodal behavior adaptation using bipolar weighted argument graphs. *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems, May 2020, Pages 1476–1484, 2020.*
- G. Weisz, P. Budzianowski, P-H Su, and M. Gasic. Sample efficient deep reinforcement learning for dialogue systems with large action spaces. 2018.
- Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers*, pages 438–449, Valencia, Spain, April 2017. Association for Computational Linguistics. URL <https://aclanthology.org/E17-1042>.
- J. Wiggings. *The Five-factor models of personality*. Guilford, 1996.
- Jason D. Williams, Antoine Raux, and Matthew Henderson. The dialog state tracking challenge series : A review. *Dialogue Discourse*, 7 :4–33, 2016.
- Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. TOD-BERT : Pre-trained natural language understanding for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 917–929, Online, nov 2020. Association for Computational Linguistics. doi : 10.18653/v1/2020.emnlp-main.66. URL <https://aclanthology.org/2020.emnlp-main.66>.
- M. Wöllmer, F. Weninger, S. Steidl, A. Batliner A., and B. Schuller. Speech-based non-prototypical affect recognition for child-robot interaction in reverberated environments. In *Interspeech, Firenze, Italy*, 2011.
- Steve Young, Milica Gasic, Simon Keizer, François Mairesse, Jost Schatzmann, Blaise Thomson, and Kai Yu. The hidden information state model : A practical framework for pomdp-based spoken dialogue management. *Computer Speech and Language* 24(2), 150–174., 2010.
- Tiancheng Zhao and Maxine Eskenazi. Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 1–10, Los Angeles, September 2016. Association for Computational Linguistics. doi : 10.18653/v1/W16-3601. URL <https://aclanthology.org/W16-3601>.