Approche stochastique bayésienne de la composition sémantique pour les modules de compréhension automatique de la parole dans les systèmes de dialogue homme-machine

Marie-Jean Meurs

▶ **To cite this version:**

Marie-Jean Meurs. Approche stochastique bayésienne de la composition sémantique pour les modules de compréhension automatique de la parole dans les systèmes de dialogue homme-machine. Autre [cs.OH]. Université d'Avignon, 2009. Français. NNT : 2009AVIG0177 . tel-00634269

HAL Id: tel-00634269

https://theses.hal.science/tel-00634269v1

Submitted on 20 Oct 2011

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.

UNIVERSITÉ D'AVIGNON
ET DES PAYS DE VAUCLUSE
MINISTÈRE DE L'ENSEIGNEMENT
SUPÉRIEUR ET DE LA RECHERCHE

ACADÉMIE D'AIX-MARSEILLE
UNIVERSITÉ D'AVIGNON ET DES PAYS DE VAUCLUSE

THÈSE

présentée à l'Université d'Avignon et des Pays de Vaucluse
pour l'obtention du grade de Docteur

SPÉCIALITÉ : Informatique

École Doctorale 166 « Information Structures Systèmes »
Laboratoire d'Informatique (EA 4128)

Approche stochastique bayésienne de la composition sémantique pour les modules de compréhension automatique de la parole dans les systèmes de dialogue homme-machine

par

Marie-Jean Meurs

Soutenue publiquement le 10 décembre 2009 devant un jury composé de :

M^{me}	Lori LAMEL	Directrice de Recherche, LIMSI-CNRS, Paris	Rapporteur
M.	Jérôme BELLEGARDA	Apple Distinguished Scientist, Apple Inc, Cupertino USA	Rapporteur
M.	Laurent BESACIER	Professeur, LIG, Grenoble	Examinateur
M.	Philippe BRETIER	Docteur, Dir. R&D, France Telecom, Lannion	Examinateur
M.	Hermann NEY	Professeur, RWTH, Aachen, Allemagne	Examinateur
M.	Renato DE MORI	Professeur, LIA, Avignon	Directeur de thèse
M.	Fabrice LEFÈVRE	Maître de Conférences HDR, LIA, Avignon	Directeur de thèse

---

Laboratoire Informatique d'Avignon

Table des matières

Remerciements

Je souhaite remercier Lori Lamel et Jérôme Bellegarda, rapporteurs de ce manuscrit, pour leur lecture attentive. Merci également à Laurent Besacier, président du jury, pour son enthousiasme et sa gentillesse ainsi qu'à Philippe Bretier et Hermann Ney, examinateurs, pour toutes leurs remarques. Merci à Renato De Mori et Fabrice Lefèvre pour l'encadrement de ces trois années de thèse.

Merci à Jean-François Bonastre d'avoir rendu cette aventure possible et merci à Denis Allard d'y avoir cru avant moi.

Merci à Catherine et Dominique pour leur soutien indéfectible. Merci à Diego et Titi, nos plantes étaient plus heureuses grâce à vous et moi aussi. Merci à Simone pour sa fiabilité et ses attentions délicates. Merci à Anthony, Bérénice, Carole, Christophe, Marie, Jeff, Flo, Georges, Rémi, Marius, Agnès, Denis, Alice, Hannah pour tous les moments partagés et les témoignages d'affection. Merci à Christian pour ses encouragements et sa confiance tout au long du chemin parcouru ensemble.

Merci enfin à ceux dont l'amour m'a soutenu pendant ces années : mes fils Charles et Antoine dont l'enthousiasme et la patience ont toujours été au rendez-vous ; Eric, dont la présence à mes côtés donne sens à toute l'histoire.

Résumé

Les *systèmes de dialogue homme-machine* ont pour objectif de permettre un échange oral efficace et convivial entre un utilisateur humain et un ordinateur. Leurs domaines d'applications sont variés, depuis la gestion d'échanges commerciaux jusqu'au tutorat ou l'aide à la personne. Cependant, les capacités de communication de ces systèmes sont actuellement limités par leur aptitude à **comprendre** la parole spontanée.

Nos travaux s'intéressent au *module de compréhension de la parole* et présentent une proposition entièrement basée sur des approches stochastiques, permettant l'élaboration d'une hypothèse sémantique complète. Notre démarche s'appuie sur une représentation *hierarchisée* du sens d'une phrase à base de **frames sémantiques**. La première partie du travail a consisté en l'élaboration d'une base de connaissances sémantiques adaptée au domaine du corpus d'expérimentation MEDIA (information touristique et réservation d'hôtel). Nous avons eu recours au formalisme FrameNet pour assurer une généricité maximale à notre représentation sémantique. Le développement d'un système à base de règles et d'inférences logiques nous a ensuite permis d'annoter automatiquement le corpus.

La seconde partie concerne l'étude du module de composition sémantique lui-même. En nous appuyant sur une première étape d'interprétation littérale produisant des unités conceptuelles de base (non reliées), nous proposons de générer des fragments sémantiques (*sous-arbres*) à l'aide de réseaux bayésiens dynamiques. Les fragments sémantiques générés fournissent une représentation sémantique partielle du message de l'utilisateur. Pour parvenir à la représentation sémantique globale complète, nous proposons et évaluons un algorithme de composition d'arbres décliné selon deux variantes. La première est basée sur une heuristique visant à construire un arbre de taille et de poids minimum. La seconde s'appuie sur une méthode de classification à base de séparateurs à vaste marge pour décider des opérations de composition à réaliser.

Le module de compréhension construit au cours de ce travail peut être adapté au traitement de tout type de dialogue. Il repose sur une représentation sémantique riche et les modèles utilisés permettent de fournir des listes d'hypothèses sémantiques scorées. Les résultats obtenus sur les données expérimentales confirment la robustesse de l'approche proposée aux données incertaines et son aptitude à produire une représentation sémantique consistante.

Introduction

Les travaux présentés dans ce document ont été réalisés dans le cadre du développement des *systèmes de dialogue homme-machine*. Ces systèmes ont pour objectif de permettre un échange oral efficace et convivial entre un utilisateur humain et un ordinateur. L'emploi du langage naturel oral comme vecteur de communication entre l'humain et la machine limite l'effort à fournir par l'utilisateur en déplaçant la charge cognitive vers la machine.

Structure d'un système de dialogue

Le dialogue en langue naturelle autorise les auto-références, les ruptures et donc les modifications dynamiques de la situation de communication. Il devient réellement coopératif dès lors que le système interlocuteur est capable de s'adapter à ces caractéristiques. Le système doit comprendre les propos de l'utilisateur, trouver une réponse et la formuler vocalement. En pratique, dans les systèmes de l'état de l'art, ces tâches sont gérées de manière séquentielle par des modules dédiés, comme cela est illustré par la figure 1.

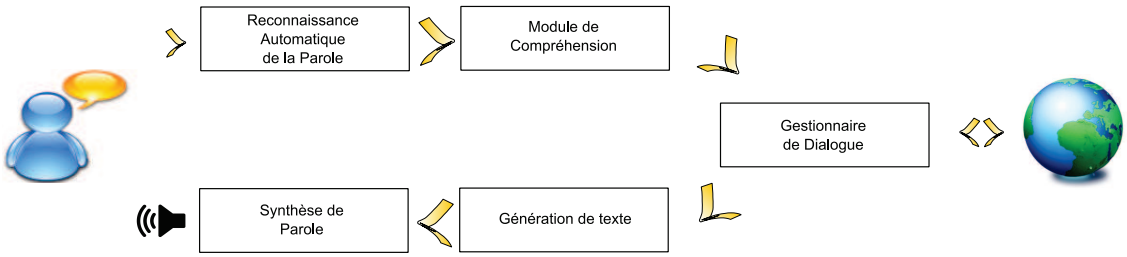


FIGURE 1 – *Schéma d'un système de dialogue.*

Les modules intervenant tour à tour dans un système de dialogue sont les suivants :

– **Reconnaissance automatique de la parole** (*Automatic Speech recognition*, ASR)
Ce module convertit le signal de parole d'entrée en une chaîne de mots. Les techniques de l'état de l'art sont basées sur des approches probabilistes. Bien que leurs performances aient été considérablement améliorées ces dernières années, les systèmes d'ASR continuent à produire des résultats comportant de nombreuses er-

reurs en raison des conditions difficiles dans lesquelles ils sont utilisés. Les systèmes traitent en effet des conversations réelles pour lesquelles sont courants un bruit environnant important, une faible articulation du locuteur ou encore des phrases incomplètes et agrammaticales. Ainsi même après qu'un décodeur ait été adapté à une tâche de dialogue particulière, un mot proposé sur cinq reste érroné.

– **Compréhension automatique de la Parole** (*Spoken Language Understanding*, SLU)
Ce module traduit généralement une entrée textuelle (transcription de l'oral) en une représentation sémantique abstraite de la requête utilisateur. La relation entre un texte et sa représentation sémantique étant éminemment ambiguë, les hypothèses de représentation sémantique étant élaborées à partir des transcriptions incorrectes produites par le module d'ASR, le module de SLU est également prompt à fournir des sorties contenant des erreurs.

  La construction de la représentation sémantique se base sur les mots transcrits mais également, selon les systèmes, sur d'autres connaissances éventuellement disponibles telles le déroulement du dialogue, les actes de dialogue identifiés ou encore la prosodie. Le choix des connaissances disponibles exploitées est fonction des structures sémantiques qui composent la représentation. Il est également dépendant des algorithmes de décision sur lesquels repose la construction de cette représentation et donc des capacités de calcul et de stockage des machines supportant les systèmes.

  Il est important de noter que la définition d'une représentation sémantique est un réel défi. Cela explique que les solutions retenues varient grandement d'un système à l'autre et qu'aucune représentation réellement standard n'existe à ce jour.

– **Gestionnaire de dialogue** (*Dialogue Manager*, DM)
Composant central d'un système de dialogue, le gestionnaire de dialogue en est le cerveau. Il coordonne le flux du dialogue en décidant des actions à réaliser par le système. Le gestionnaire de dialogue est aussi responsable de l'interaction avec le monde extérieur via diverses interfaces (bases de données, sites Web ou toute source d'information à laquelle la machine a accès). Décider de la meilleure action à réaliser est rarement facile car plusieurs actions sont généralement pertinentes. De plus, la prise de décision est d'autant plus complexe qu'elle est fondée sur des informations incertaines provenant des sorties erronées des modules d'ASR et de SLU.

– **Génération** (*Natural Language Generation*, NLG) et **synthèse de parole** (*Text-to-Speech Synthesis*, TTS)
Tout système de dialogue doit disposer d'un module lui permettant de communiquer l'information qu'il détient à l'utilisateur. Cette étape est cruciale dans la mesure où le système peut être perçu comme confus, inefficace, voire stupide si l'information est mal présentée. La transmission de l'information sera généralement effectuée par un module de génération en langage naturel suivi d'un mod-

ule de production de parole. La génération en langage naturel peut être obtenue en utilisant simplement des patrons textuels prédéfinis ou par le moyen de techniques plus complexes (Walker et al., 2007). La production de parole peut s'appuyer sur de la synthèse de parole ou des échantillons de parole pré-enregistrés (Taylor, 2009).

La plupart des systèmes de dialogue sont construits comme des cascades de modules. Chaque module est interfacé avec le suivant au moyen de résultats intermédiaires (séquences de symboles, liste de n-meilleures ou même treillis d'hypothèses). Cette approche est perfectible car elle ne permet pas facilement la combinaison des modules entre eux. Une telle architecture a cependant pour avantage de favoriser le développement et l'optimisation séparés des modules. Elle permet en outre à chaque module d'emprunter largement aux techniques utilisées dans d'autres domaines du traitement automatique de la parole.

Le niveau de compréhension par le système de dialogue des messages de l'utilisateur est donc lié à la qualité des transcriptions fournies par le module d'ASR et à la capacité d'exploitation de ces transcriptions par le module de compréhension. Ainsi, la phase de compréhension automatique nécessite la conception d'un module apte à réaliser des inférences complexes à partir de données incertaines pour produire une représentation sémantique fine du message traité. L'objet des travaux présentés dans ce document est le développement de modèles stochastiques adaptés à la construction de la représentation sémantique par le module de compréhension.

Applications

Les domaines d'application des systèmes de dialogue sont variés. Depuis le système ATIS (*Air Travel Information Service*) (Price, 1990), dédié à la consultation d'horaires d'avions aux Etats-Unis (projet ARPA), l'essor des services basés sur la téléphonie interactive a motivé le développement de nombreux systèmes parmi lesquels "How May I Help You" (Gorin et al., 1997) d'AT&T qui s'intéresse au routage d'appels téléphoniques, "ARISE" (*Automatic Railway Information Systems for Europe*) développé au LIMSI (Lamel et al., 2000), dédié à la recherche d'information et d'horaires ferroviaires ou encore le service "3000", évolution du serveur "AGS" (Sadek et al., 1995), réalisé et exploité commercialement par France Télécom (Damnati et al., 2007) qui permet aux utilisateurs de gérer leur compte client, de contrôler leur consommation ou même de payer leur facture.

Les applications dédiées au tutorat, notamment dans le contexte de l'apprentissage des langues, sont nombreuses à utiliser des systèmes de dialogue. Développées dans l'esprit des tuteurs intelligents des années 70-80, les applications du domaine sont encore souvent basées sur le dialogue textuel mais de nombreux projets académiques récents intègrent avec succès l'oral comme modalité d'échange (ITSPOKE (Litman et Silliman, 2004), SCoT (Pon-Barry et al., 2006)...).

Les systèmes de dialogue peuvent également être utilisés dans le cadre de l'aide

aux personnes handicapées (voir (Hockey et Miller, 2007) qui présentent le pilotage d'un fauteuil électrique par le biais d'un dialogue oral entre l'utilisateur et la machine) ou aux personnes âgées (voir l'adaptation du système d'information LET'S GO ! aux personnes âgées (Raux et al., 2003)).

Les systèmes de dialogue peuvent être classés selon le degré de naturel qu'ils acceptent de la part de l'utilisateur. Le style de l'échange peut être limité à de simples questions fermées, des mots clés choisis dans un court menu, un nombre restreint de locutions ou admettre la parole continue, sans contrainte sur le vocabulaire utilisé. Limiter le style des échanges facilite la reconnaissance de la parole et rend aisée la tâche de compréhension. Cependant, dans ce contexte, le nombre de tours de parole permettant d'atteindre un objectif est élevé car l'expressivité du système est faible. L'utilisateur est limité par les capacités d'échange d'un tel système de dialogue.

Un système efficace et agréable à utiliser doit donc supporter le style d'expression le plus élaboré possible. Le niveau de complexité idéal reste cependant fonction des tâches assignées au système considéré et de son domaine d'application. Par exemple, le contrôle d'un appareil électronique pourra être réalisé par le biais d'un système de dialogue ne supportant que les questions fermées. En revanche, seul un système admettant un style d'expression élaboré pourra être intégré avec profit dans un outil d'aide à l'apprentissage (tutoring).

Le niveau de liberté laissée à l'utilisateur dans la conduite du dialogue est également un critère d'évaluation important des systèmes de dialogue qui peuvent permettre la prise de contrôle totale ou partielle par l'utilisateur au cours de l'échange. Dans les systèmes de base, l'utilisateur doit uniquement répondre aux questions posées en donnant l'information demandée. Le déroulement du dialogue est le fait unique du système qui garde l'initiative durant tout l'échange. Cette approche, quoique très pratique du point de vue du développement, conduit à des dialogues sémantiquement pauvres et donc peu utilisables.

L'alternative consiste à permettre à l'utilisateur de prendre l'initiative en fournissant des informations non demandées qui complètent ou modifient l'information souhaitée par le système. Une telle approche "mixte" est généralement bénéfique car elle permet de raccourcir substantiellement la longueur des dialogues selon l'expertise de l'utilisateur. Toutefois, la stratégie mixte peut se rapprocher de la stratégie guidée par le système lorsque l'utilisateur est novice et découvre le service (voire les interactions vocales homme-machine en général). Enfin, la stratégie mixte peut devenir une stratégie guidée par l'utilisateur si celui-ci, utilisant le système fréquemment, est capable de fournir toute l'information requise en un seul tour de parole. La complexité du contenu sémantique des échanges est alors considérablement accrue et doit être abordée avec des approches d'autant plus performantes.

Projet LUNA

Les travaux présentés dans cette thèse ont été réalisés dans le cadre du projet européen LUNA [1]. Ce projet est inclus dans le sixième programme-cadre de recherche (FP6) de l'Union Européenne (EU), centré sur Les Technologies de la Société de l'Information (IST). Il s'intéresse au problème de la compréhension en temps réel de la parole spontanée dans le cadre des services de télécommunication avancés. Le principal objectif du projet LUNA est la création d'outils robustes de compréhension automatique de la parole pour les services de dialogue multilingues. Ces outils doivent être capables de gérer la communication homme-machine en apportant toute satisfaction aux utilisateurs.

D'un point de vue technologique, les objectifs de LUNA sont de proposer de nouvelles méthodes, de nouveaux algorithmes et de nouveaux outils pour développer des composants robustes de systèmes de compréhension orale pour les services téléphoniques multilingues. Dans cette perspective, LUNA s'intéresse à plusieurs problèmes scientifiques dont les thématiques principales sont la modélisation du langage et la modélisation sémantique pour la compréhension orale, l'apprentissage automatique, la robustesse des systèmes de compréhension orale et la portabilité multilingue. Les modèles de compréhension sont entraînés et appliqués à des systèmes de dialogue en français, italien et polonais. Les résultats des recherches LUNA sont validés à l'aide de différents scenarii liés à des services de dialogue téléphoniques de complexités variables (depuis l'acheminement automatique d'appels par classification de phrases types jusqu'aux systèmes de dialogue adaptés aux champs sémantiques complexes).

Le projet LUNA définit trois niveaux de complexité pour les applications de dialogue. Le premier niveau inclut *l'interprétation littérale*, c'est à dire le processus de traduction des mots en unités conceptuelles de base (génération de concepts sémantiques). Ce niveau de détail est suffisant pour des applications telles que la classification d'appel (call-routing) ou la classification de phrases en catégories disjointes. Le deuxième niveau réalise *la composition sémantique* des constituants de base pour des applications de type classification d'appel avec caractérisation de phrases (compréhension plus fine), questions/réponses ou détermination de requêtes. Au troisième niveau, un contexte plus large est considéré pour obtenir une *validation dépendante du contexte de dialogue* dans les systèmes de dialogue complexes. Les différents modèles sémantiques et les processus d'interprétation associés étudiés dans ce travail seront plus particulièrement focalisés sur le second niveau. Les approches sous-jacentes pour le premier niveau (décodage d'unités conceptuelles simples) seront décrites. Toutes les approches proposées prennent en compte les nécessités imposées par le dernier niveau (dépendance au contexte large).

1. http ://www.ist-luna.eu/

Problématique

Les systèmes de dialogue industriels actuels, tels ceux mis en œuvre par France-Télécom pour ses services aux clients, sont construits autour d'approches codées manuellement de type VoiceXML comme technique standard de gestion du dialogue (ou toute autre méthode comparable pour spécifier des automates à états finis). Ces systèmes font l'hypothèse de la connaissance totale des états courants du dialogue lors de l'exécution et sont limités à la complexité de dialogues pour le renseignement de champs de formulaires, avec des transitions de type état fini entre les formulaires. De tels systèmes n'incluent pas de mécanisme d'optimisation tenant compte de données réelles (mise à part la révision manuelle de la stratégie de dialogue après le constat d'erreurs ou de problèmes récurrents). Toutefois, l'avantage d'un modèle aussi simple est qu'il est très prévisible et peut être rendu relativement robuste aux erreurs, même si les systèmes obtenus sont assez inefficaces et frustrants à l'usage pour les utilisateurs. Il est aussi facile pour les développeurs d'encoder des déroulements de dialogues simples dans un tel modèle.

Traditionnellement, les approches pour la gestion du dialogue ont été classées selon trois grandes catégories : les grammaires de dialogues, les approches à base de plan et les approches collaboratives. Ces approches ne sont pas mutuellement exclusives et sont souvent utilisées simultanément. Un inconvénient des approches à états finis est que le gestionnaire de dialogue doit "manuellement" développer le dialogue (i.e. développer l'état descriptif de la machine, les cadres ou les actions avec leur pré-conditions, mise à jour des états de dialogue, post-conditions...). Aussi, les structures du dialogue sont rigides, les dialogues actuels ne permettant pas d'améliorer les dialogues futurs.

Le développement de méthodes d'apprentissage automatique appliquées à l'optimisation de stratégies de dialogue est devenu un axe de recherche prometteur depuis le milieu des années 90 (Levin et al., 1997; Walker, 1998; Singh et al., 2002). Plus récemment des expériences ont été réalisées sur l'application de l'apprentissage par renforcement à l'optimisation de systèmes d'interrogation de bases de données par requêtes (Williams et Young, 2007; Lefèvre et de Mori, 2007; Young et al., 2010; Laroche et al., 2009). L'approche propose un modèle de l'incertitude dans les systèmes de dialogue rendant explicite précisément ce qui est optimisé et applique des techniques mathématiques rigoureuses pour obtenir un stratégie optimale. Avec les évolutions récentes de l'approche et un couplage renforcé avec les modules d'ASR et de SLU, il est très probable que les performances augmentent et qu'il devienne possible de généraliser l'approche à tous les composants des systèmes de dialogue.

La complexité des systèmes de dialogue oral homme-machine dépend essentiellement de la complexité de la tâche visée. Ainsi les systèmes dédiés au routage d'appels téléphoniques ou à la recherche d'information, pouvant se représenter à l'aide de formulaires (répertoires téléphoniques, recherche d'horaires par exemple), reposent généralement sur une représentation des connaissances sémantiques relativement simple qui permet l'utilisation d'approches globales (comme par exemple (Bellegarda et

Silverman, 2003; Potamianos et al., 2005)).

A contrario dès lors qu'un système doit pouvoir gérer plusieurs demandes conjointes ou intégrer des phases de négociation, une représentation sémantique de haut niveau est requise. De tels systèmes doivent disposer de connaissances sémantiques permettant d'interagir avec des informations complexes sur l'état courant du dialogue tout en supportant les erreurs introduites dans la chaîne de traitement du signal de parole par les modules d'ASR et de SLU. La difficulté à établir une telle représentation de haut niveau, qui soit à la fois fonctionnelle et robuste, explique que les systèmes de dialogue actuels soient limités à des espaces sémantiques assez restreints.

Développer un système de dialogue capable de se remettre en cause, de proposer des alternatives et de s'adapter aux phases de négociation présentes dans le dialogue naturel suppose la construction d'une représentation des connaissances qui permette la composition sémantique au sein d'un tour de parole comme au cours du déroulement du dialogue. Parallèlement, quelle que soit l'expressivité du modèle de représentation sémantique sur lequel il s'appuie, le module de compréhension d'un système de dialogue ne sera réellement efficace que s'il est capable de s'adapter aux erreurs commises par le module d'ASR.

L'introduction de processus stochastiques dans chaque composant des systèmes permet d'améliorer leurs performances en augmentant leur robustesse aux variabilités de la parole. Si la plupart des modules d'ASR reposent actuellement sur des approches probabilistes, les modules de SLU stochastiques sont encore peu nombreux. Ainsi, même s'il est maintenant admis que les méthodes stochastiques sont d'efficaces alternatives aux méthodes à base de règles pour la compréhension de l'oral (Levin et Pieraccini, 1995; He et Young, 2006; Lefèvre, 2007), le développement de modules de SLU stochastiques est freiné par la disponibilité encore limité de corpus de dialogue annotés sémantiquement. De plus, les avantages de ces modules ne sont pas pris en compte par les gestionnaires de dialogue déterministes, inaptes au traitement d'hypothèses sémantiques multiples et valuées. Le développement récent de gestionnaires de dialogue probabilistes ouvre donc de nouvelles perspectives au déploiement des modules de SLU stochastiques.

Dans ce travail, la tâche de compréhension est envisagée par étapes. Une première étape d'interprétation littérale s'intéresse à la génération séquentielle de concepts élémentaires associés aux segments lexicaux qui composent le message de l'utilisateur. Une seconde étape réalise la composition de ces concepts de base pour obtenir une représentation sémantique hiérarchique du message. Cette étape de composition considère le message dans sa globalité et constitue le cœur de l'étude présentée ici. Ces travaux ont porté sur le renforcement de deux caractéristiques indispensables à un système de compréhension performant : la richesse de la *représentation des connaissances* et la robustesse face aux *données incertaines*.

Organisation du document

La première partie de ce document présente les connaissances actuelles en **compréhension du dialogue** et quelques systèmes de référence. Elle comporte également une description du corpus de dialogues oraux support des travaux réalisés.

Le chapitre 1 rappelle les fondements de l'approche linguistique du problème de la compréhension et décrit quelques systèmes de référence basés sur cette approche. Le chapitre 2, construit de façon identique, s'intéresse à l'approche stochastique du problème. Un aperçu non-exhaustif des représentations sémantiques adaptées à la compréhension orale est proposé chapitre 3. Le corpus MEDIA qui a servi de matériau d'expérimentation et d'évaluation à ce travail est décrit dans le chapitre 4.

La seconde partie du document détaille **les contributions** proposées selon trois axes : la production des données d'apprentissage dont dépendent les modèles stochastiques utilisés, la génération de fragments sémantiques par ces modèles et enfin la composition des fragments en une structure globale.

Les chapitres 5 et 6 décrivent respectivement la base de connaissances sémantiques construite pour le corpus MEDIA et le système d'annotation déterministe utilisé pour produire des annotations de référence sur ce corpus.

Le chapitre 7 rappelle les fondements théoriques des réseaux bayésiens dynamiques utilisés dans notre travail. L'application de ces modèles stochastiques à la génération de fragments sémantiques est présentée chapitre 8. Les expérimentations réalisées et les résultats obtenus sont détaillés dans le chapitre 9.

Le chapitre 10 propose une présentation de la notion d'*arbre* employée pour représenter les relations sémantiques dans le contexte de notre travail. Il rappelle ensuite les fondements théoriques des modèles de classification basés sur les séparateurs à vaste marge utilisés dans l'une des stratégies étudiées pour la composition des fragments sémantiques. Les algorithmes de recomposition d'arbres utilisés pour finaliser le processus de compréhension sont présentés chapitre 11. Les expérimentations réalisées et les résultats obtenus sont détaillés dans le chapitre 12.

Première partie

COMPRÉHENSION du DIALOGUE : théories, systèmes, matériau expérimental

Introduction

Les travaux présentés dans ce document s'inscrivent dans un cadre applicatif. Les approches privilégiées sont essentiellement pragmatiques et motivées par les performances des systèmes qui les utilisent. Bien qu'éloignés de toute recherche d'universalité, ces systèmes ont pour objectif ultime la *compréhension* de la langue naturelle. A ce titre leur développement s'inscrit dans le prolongement d'un questionnement philosophique dont les premières traces écrites remontent à l'Antiquité. Une part importante de ce questionnement porte sur la relation entre la valeur sémantique d'un énoncé complet et ses constituants (*compositionnalité*) et mérite donc qu'on s'y attarde quelques instants avant de reprendre notre cheminement technologique.

La philosophie s'intéresse au langage depuis le *Cratyle* de Platon, dialogue dans lequel Socrate, Cratyle et Hermogène s'interrogent sur l'arbitraire ou le naturel des signes de la langue. Socrate y combat les deux thèses opposées, concluant sa démonstration sur la seule exigence du sens. Depuis lors, la philosophie a toujours réfléchi à un grand nombre de questions autour du langage : son origine, ses fonctions, sa capacité à exprimer des significations, des relations sémantiques et en cela, le rapport entre langage et pensée, langage et société humaine. Pour autant, la *philosophie du langage* contemporaine est assez détachée des philosophes classiques.

Née en même temps que la logique formelle inspirée au philosophe et mathématicien allemand Gottlob Frege (1848-1925) par Leibniz (1646-1716) et sa caractéristique universelle [2], la philosophie du langage contemporaine est étroitement liée à la linguistique depuis le milieu du XXème siècle.

Selon le philosophe Diego Marconi, le but de la sémantique, pour le paradigme dominant, est de déterminer systématiquement les conditions de vérité des énoncés d'un langage, indépendamment de la manière dont elles sont déterminées elles- mêmes pas tel ou tel locuteur, et finalement tout aussi indépendamment du fait qu'elles lui soient connues, ou même accessibles (Marconi, 1995). Héritière de l'approche décrite par Marconi, une bonne partie de la recherche sémantique contemporaine est fondée sur le principe de compositionnalité de Frege.

2. La caractéristique universelle -*characteristica universalis*- est une langue universelle et formelle, imaginée par Leibniz, qui serait capable d'exprimer aussi bien les concepts mathématiques, scientifiques ou métaphysiques. C'est aussi bien une lingua universalis (langue universelle) qu'une lingua rationalis : la possibilité d'une communication universelle est fondée sur l'universalité de la raison. Leibniz espérait ainsi créer une langue utilisable dans le cadre d'un calcul logique universel ou *calculus ratiocinator*.

Ce principe stipule que la valeur sémantique (sens ou dénotation) de toute expression complexe est fonction des valeurs sémantiques de ces constituants. L'une de ses justifications premières est la suivante : il serait difficile de concevoir, sans admettre un principe de compositionnalité de la signification, que l'on puisse comprendre des phrases que nous n'avons jamais entendues - sans qu'elles nous soient expliquées -, à la seule condition qu'elles soient constituées de mots que nous connaissons. Évidemment, nous calculons la signification des expressions nouvelles à partir des significations de leurs sous-expressions, que nous connaissons déjà. La signification d'une expression complexe est, en ce sens, fonction des significations de ses constituants : la connaissance des significations des constituants suffit à déterminer, sur la base de la structure syntaxique de l'expression, la signification de l'expression complexe.

Cependant, le sens d'un énoncé est la pensée qu'il exprime. Or, la valeur sémantique du tout n'est pas toujours fonction des valeurs sémantiques des parties. Les limites de la compositionnalité sont atteintes lorsque les mêmes mots se trouvent avoir des sens différents dans des contextes différents.

Dans ce cadre de réflexion, Frege définit les notions de *concepts* et de *prédicats* en constatant, à l'instar des représentations mathématiques, que le langage naturel est capable lui aussi d'exprimer des fonctions. Par exemple, on peut considérer que l'expression *la capitale de [ ]* désigne une fonction, qui fait correspondre à chaque pays sa capitale. Le langage naturel est en particulier capable d'exprimer des concepts, c'est à dire des fonctions dont la valeur est une valeur de vérité. Par exemple, l'expression *[ ] est la capitale de la France* dénote une fonction qui prend la valeur Vrai pour l'argument *Paris* et Faux pour tout autre argument. De la même manière, *[ ] est un homme* dénote une fonction qui prend la valeur Vrai si l'argument est un homme et Faux dans tous les autres cas.

Les expressions telles que *[ ] est un homme* sont appelées des prédicats. Les prédicats sont donc des expressions linguistiques qui dénotent un type particulier de fonctions : des fonctions dont les valeurs sont des valeurs de vérité, autrement dit des concepts.

Sur cette base, Frege analyse *en intention* les énoncés simples, tels *Socrate est un homme*[3]. Dans cet exemple, Socrate est l'argument du prédicat *[ ] est un homme*. Ce prédicat dénote le concept homme, c'est à dire la fonction qui prend la valeur Vrai pour tout argument homme et Faux sinon.

Le philosophe et mathématicien anglais Bertrand Russell (1872-1970) oppose à cette approche une analyse du langage qui se dispense du concept de sens : la seule propriété sémantique d'une expression linguistique importante pour la valeur de vérité des énoncés est sa dénotation (Russell, 1905).

La conception *en extension*, purement dénotative, du langage de Russell contribue à la longue éclipse de la notion de sens. Influençant fortement la formation du *Tractatus logico-philosophicus* de Wittgenstein (1889-1951), elle fonde ainsi cette approche analytique qui se développe parallèlement aux travaux des linguistes structuralistes de l'époque tels Saussure (1857-1913) puis Jakobson (1896-1982).

3. Wittgenstein et Russell nommeront ces énoncés des *énoncés atomiques*.

La linguistique générative du linguiste Noam Chomsky (1928-) rapproche philosophie du langage et linguistique dès les années 50 (Chomsky, 1957). Philosophes et linguistes fournissent alors les outils linguistiques réflexifs et formels. Les informaticiens les implémentent et les exploitent dans des applications concrètes de Traitement Automatique de la Langue Naturelle (TALN) telles la traduction automatique ou les systèmes de dialogue homme-machine.

Cette approche strictement linguistique est limitée à trois niveaux. Tout d'abord, les applications fondées exclusivement sur ces approches formelles nécessitent l'utilisation de ressources décrivant la tâche visée. La constitution de ces ressources est une contrainte extrêmement lourde, tant du point de vue humain que temporel. En effet, elle ne peut être réalisée que par un travail méticuleux de linguistes et requiert souvent plusieurs années.

Ensuite, les approches formelles ne sont pas pensées pour permettre la gestion de l'ambigüité, pourtant inhérente à la langue naturelle et spécialement à la parole spontanée.

Enfin, les systèmes développés dans ce cadre théorique souffrent d'un manque de robustesse : toute situation rencontrée par le système mais n'ayant pas été prévue par ses concepteurs met logiquement l'application en défaut.

Ces approches, pertinentes pour décrire la langue naturelle et en expliquer les mécanismes, sont donc perfectibles dans le cadre applicatif. Consciente de ces limites, une partie de la communauté du TALN s'oriente vers des approches empiriques, essentiellement basées sur des processus stochastiques. Le principal atout de ces approches est de limiter l'intervention d'experts humains lors du développement de systèmes de compréhension. En effet, il n'est plus nécessaire de décrire la tâche par un ensemble de règles ou de grammaires spécifiques : il suffit de définir les entités linguistiques choisies pour l'apprentissage des modèles stochastiques. Ces modèles étant à même d'apprendre les relations entre objets linguistiques et sémantiques, les caractéristiques de la représentation sémantique choisie dans le cadre de l'application influent sur les performances des systèmes.

L'approche linguistique et l'approche stochastique ont donc servi de bases théoriques au développement de la majorité des systèmes de compréhension que nous connaissons. Ces systèmes s'appuient sur des représentations sémantiques variées.

Le chapitre 1 ci-après s'intéresse aux systèmes développés selon l'approche linguistique. Le chapitre 2 présente ceux issus de l'approche stochastique. Le chapitre 3 étudie les représentations sémantiques dédiées à la compréhension. Enfin, le chapitre 4 présente le corpus de dialogue MEDIA sur lequel les expériences ont été menées.

Chapitre 1

Approche linguistique

Sommaire

Résumé

Ce chapitre s'intéresse aux principaux fondements de l'approche linguistique du problème de la compréhension. L'application du principe de compositionnalité est exposé en 1.2. La partie 1.3 présente les grammaires formelles de Chomsky. Les évolutions et les premières applications de ces grammaires sont détaillées dans la section 1.4. Enfin, la partie 1.5 présente les grammaires stochastiques et quelques systèmes les utilisant.

1.1 Introduction

Les systèmes développés selon l'approche linguistique se basent sur l'application du principe de compositionnalité de Frege à partir d'une analyse syntaxico-sémantique de la proposition. Après de brefs rappels théoriques, quelques-uns de ces systèmes sont présentés dans ce chapitre. Les premiers d'entre eux ont été réalisés sous la contrainte technique d'une puissance de calcul restreinte et sont pionniers dans le domaine de l'interaction homme-machine.

Le principe de compositionnalité de Frege est explicité en 1.2. Les grammaires formelles de Chomsky, créées dans ce contexte réflexif et appliquées à la construction des arbres syntaxiques, sont présentées en 1.3. La partie 1.4 expose des évolutions de ces grammaires orientées vers la prise en compte de connaissances sémantiques. L'introduction de paramètres stochastiques dans les approches à base de grammaires est exposée dans la section 1.5.

1.2 Principe de compositionnalité

Dans la plupart des systèmes issus de l'approche linguistique, tous les sens possibles de chaque mot sont considérés. Ces informations sont ensuite composées sous la contrainte d'obtenir un sens cohérent pour chaque proposition. Une approche de ce type, décrite dans (Allen, 1988), consiste à analyser une phrase écrite pour obtenir l'arbre syntaxique qui lui est associé. Un ensemble de règles fait ensuite correspondre les blocs de l'arbre à des fragments de représentations sémantiques définis au sein d'une ontologie structurée.

Cette approche, issue des travaux de Frege, est justifiée par l'hypothèse que chaque constituant syntaxique important d'une phrase correspond à un constituant conceptuel, la réciproque étant fausse. La figure 1.1 présente l'exemple de l'arbre sémantique associé à la proposition "Je cherche un hôtel Sofitel pour le soir du 25 octobre"

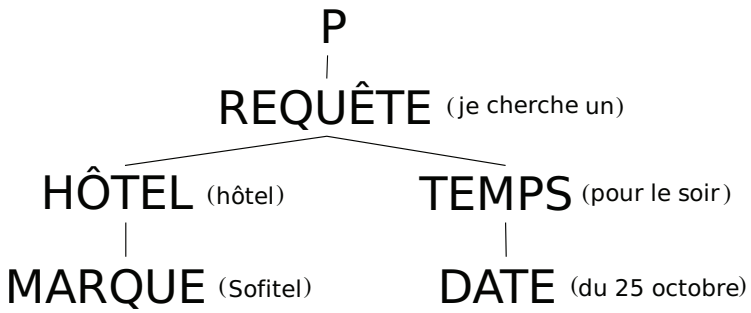


FIGURE 1.1 – *Arbre sémantique associé à la proposition "Je cherche un hôtel Sofitel pour le soir du 25 octobre".*

Ce formalisme est basé sur un ensemble de catégories. Chaque catégorie peut être détaillée par une fonction et un argument. Dans l'exemple ci-dessus, la catégorie DATE

est associée à la fonction *du* et l'argument *25 octobre*. L'expression P peut être obtenue à partir d'une structure syntaxique telle que :

G : P[VP [PR je, V cherche] NP [ART un, N hôtel, N Sofitel] NP [PREP pour, ART le, N soir] NP [PREP du, ADJ vingt-cinq, N octobre]]]

Une partie de l'arbre syntaxique de cette structure est présentée dans la figure 1.2

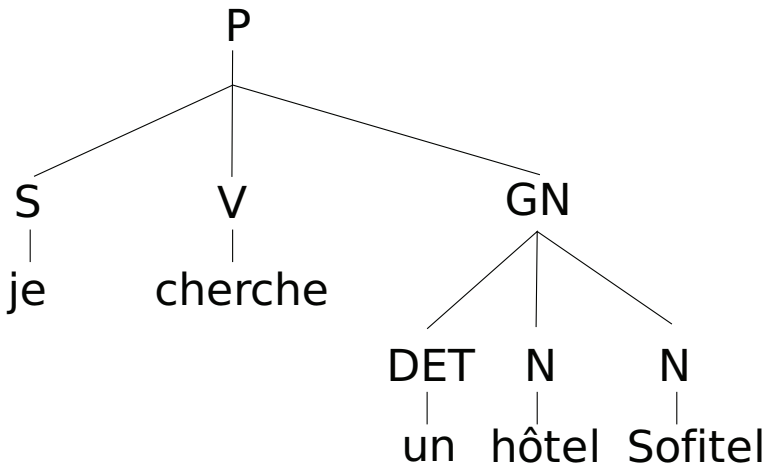


FIGURE 1.2 – *Arbre syntaxique associé à la proposition "Je cherche un hôtel Sofitel".*

Selon les domaines d'application, des représentations sémantiques peuvent être associées à des nœuds non terminaux de l'arbre syntaxique et l'interprétation de la phrase peut être réalisée en utilisant les étiquettes sémantiques de ces associations.

1.3 Grammaires formelles

Ces modélisations sont issues des grammaires formelles de Chomsky. Ces grammaires s'inspirent du langage formel et tentent d'intégrer les caractéristiques du langage humain à l'aide de règles d'association des mots. Composées d'un nombre fini de règles de production (règles de réécriture), elles permettent de générer et d'analyser un langage donné.

Par définition, une grammaire formelle G est un quadruplet (V_N, V_T, R, P) avec :
V_T : vocabulaire terminal (ensemble des symboles terminaux)
V_N : vocabulaire non terminal (ensemble des symboles non terminaux)
V : vocabulaire ($V = V_T \cup V_N$)
P : axiome, élément de V_N
R : ensemble de règles de réécriture de la forme : $\alpha \rightarrow \beta$ telles que $(\alpha, \beta) \in V^* \times V^*$ et $\alpha \neq \epsilon$

On appelle *langage engendré par G* l'ensemble de toutes les suites de symboles qui dérivent de l'axiome P de G : $L(G) = \{x, x \in V^* etP \Rightarrow^* x\}$ (i.e. x peut être obtenu à partir de P grâce à une succession de réécritures).

Un langage est dit *décidable* si pour toute phrase, on peut savoir au bout d'un temps fini si elle appartient ou non au langage.

Les grammaires formelles sont classées hiérarchiquement par Chomsky et Schützenberger (Chomsky et Schützenberger, 1963) par ordre d'expressivité décroissante.

◦ **Les grammaires de type 0** se définissent par des règles du type :

$$\alpha \rightarrow \beta \quad avec \quad \alpha, \beta \in V^*$$

Ces grammaires ne sont pas décidables.

◦ **Les grammaires de type 1**, dites grammaires contextuelles, se définissent par des règles du type :

$$\alpha A \beta \rightarrow \alpha \gamma \beta \quad avec \quad A \in V_N \quad \alpha, \beta, \gamma \in V^*, \gamma \neq \epsilon$$

Toute règle comporte un symbole non terminal entre deux mots que l'on retrouve après la dérivation. Le non terminal est transformé de façon non nulle. Les mots qui encadrent le non terminal représentent son contexte qui va influer sur sa dérivation. Les grammaires de type 1 sont décidables. Pour déterminer si une phrase de longueur n appartient au langage, il suffit de réaliser toutes les dérivations comportant n symboles ou moins, ce qui nécessite un temps fini. Cependant, la génération est de complexité exponentielle en n (le temps d'analyse est proportionnel à l'exponentielle du nombre de mots à analyser).

◦ **Les grammaires de type 2**, dites grammaires algébriques ou hors-contexte, se définissent par des règles de la forme :

$$A \rightarrow \gamma \quad avec \quad A \in V_N \quad \gamma \in V^*$$

Bien que limitées par leur incapacité à traiter les dépendances longues distances, l'usage des grammaires hors-contexte est souvent privilégié en TALN, essentiellement en raison du bon compromis entre leur capacité descriptive et leur complexité (polynomiale en $O(n^3)$, où n est le nombre de mots à analyser, d'après l'algorithme CYK).

◦ **Les grammaires de type 3** ou grammaires régulières sont linéaires gauches ou droites. Les règles qui les définissent sont de la forme :

grammaire linéaire gauche	*grammaire linéaire droite*
$A \rightarrow Ba$	$A \rightarrow aB$
$A \rightarrow a$	$A \rightarrow a$
$avec\ A, B \in V_N, a \in V_T$	$avec\ A, B \in V_N, a \in V_T$

Les grammaires régulières sont utilisés en TALN pour la représentation compacte des lexiques, la construction de correcteurs orthographiques, la composition de grammaires

locales (traitement des nombres). Dans le cadre du dialogue et pour des applications à des domaines restreints, elles sont souvent choisies en raison de leur complexité linéaire.

◦ **Les grammaires de type 4**, dites grammaires à choix finis, se définissent par des règles de la forme :

$$A \rightarrow a \quad avec \quad A \in V_N \quad a \in V_T$$

Ces grammaires ne permettent que l'énumération des phrases de leur langage sur V_T.

1.4 Évolutions

Conscient de l'incapacité des grammaires hors-contexte à modéliser toutes les subtilités du langage naturel (Chomsky, 1964), Woods propose l'utilisation des grammaires à base de réseaux de transitions augmentés (GRTA) dans les procédures de décomposition syntaxique (Woods, 1970). Dans la perspective de mieux modéliser richesse et complexité du langage naturel, ces grammaires contiennent des connaissances sémantiques sensibles au contexte et leurs stratégies de décomposition syntaxique incluent des processus d'inférence logique.

Ces grammaires sont une extension des grammaires à base de réseaux de transitions (GRT). Les GRT sont faiblement équivalentes aux grammaires hors-contexte dont elles ne diffèrent en équivalence forte que par leur aptitude à caractériser les arborescences redondantes du type S[S et S et ... et S]. Elles intègrent, via des réseaux de transitions, les caractéristiques que les grammaires de transitions ajoutent aux grammaires hors-contexte.

Composées d'états reliés par des arcs (graphes orientés), ces grammaires ont l'expressivité des grammaires hors-contexte à laquelle s'ajoute la capacité de déplacer des fragments de structure, de les recopier, de les supprimer : Ces actions sont généralement dépendantes du contexte dans lequel les fragments apparaissent. La chaîne d'entrée est analysée de gauche à droite durant la décomposition, mot par mot. Le mot entrant et l'état courant détermine l'arc emprunté par le processus. Des GRT sont utilisées pour la compréhension de la parole spontanée par (Young et al., 1989).

Dans les GRTA, des tests conditionnels peuvent être associés à certains arcs et un ensemble de structures de construction peuvent être effectuées si l'arc est emprunté (composition d'arbres, génération d'interprétations sémantiques). En effet, le réseau de transitions augmenté fournit une description structurelle partielle de la phrase à chaque état. Ces descriptions sont stockées dans des registres mis à jour au fil de l'analyse. Le contenu des registres est composé des valeurs des caractéristiques linguistiques et peut aussi être utilisé pour construire les arbres d'analyse. Une approche de ce type est décrite dans (Woods et al., 1976) et est proposée dans le projet ARPA de 1971, détaillé dans (Klatt, 1977). Il inclut des approches essentiellement basées sur l'Intelligence Artificielle (IA) pour combiner analyse syntaxique et représentation sémantique en logique formelle. Les systèmes de ce projet génèrent des hypothèses de séquences de mots grâce

à un système de reconnaissance automatique de la parole puis produisent une interprétation avec les mêmes approches que celles utilisées pour le traitement de l'écrit.

Les concepts et relations d'un réseau sémantique peuvent aussi être implémentés dans le formalisme des *cadres sémantiques* (Fillmore, 1985). Le concept linguistique original des cadres sémantiques (ou cadres de cas, *case-frame*, ou encore grammaires de cas, *case-based grammar*), comme proposé par (Fillmore, 1968), est basé sur la définition d'un ensemble de cas universels qui permet de mettre en avant la relation entre un verbe et ses composants nominaux. En se référant à la terminologie de (Bruce, 1975), un *cas* est une relation entre un prédicat (en général le verbe, mais pas exclusivement) et un des ses arguments. Un marqueur de cas est un indicateur de structure de surface (par exemple une préposition) pour le cas concerné. Un cadre de cas d'un prédicat est constitué de l'ensemble des cas propres à ce prédicat. Une grammaire de cas est un jeu complet de cas pour un langage entier.

L'approche par grammaire de cas est appropriée pour les systèmes de compréhension de la parole où le besoin d'un support sémantique lors de l'analyse est fondamental. L'analyseur sémantique réalise une analyse par cas pour déterminer le sens d'une requête, et construit la représentation en cadres correspondante. L'historique du dialogue est utilisé pour compléter le cadre sémantique en cas de besoin. Les cas définissant la grammaire complète ainsi que les mots-clés associés (*trigger keywords*) sont dans une large mesure dépendants de la tâche et du domaine du système de dialogue. Les grammaires de cas ont été appliquées avec succès dans de nombreux systèmes (par exemple (Hayes et al., 1986; Ward, 1991)) et largement popularisées à partir des années 90 par leur utilisation dans les systèmes du LIMSI (Matrouf et al., 1990; Bennacef et al., 1994, 1996; Lamel et al., 1999). Un exemple d'un cadre utilisé pour la tâche ATIS en français (Bennacef et al., 1994) est donné dans la figure 1.3.

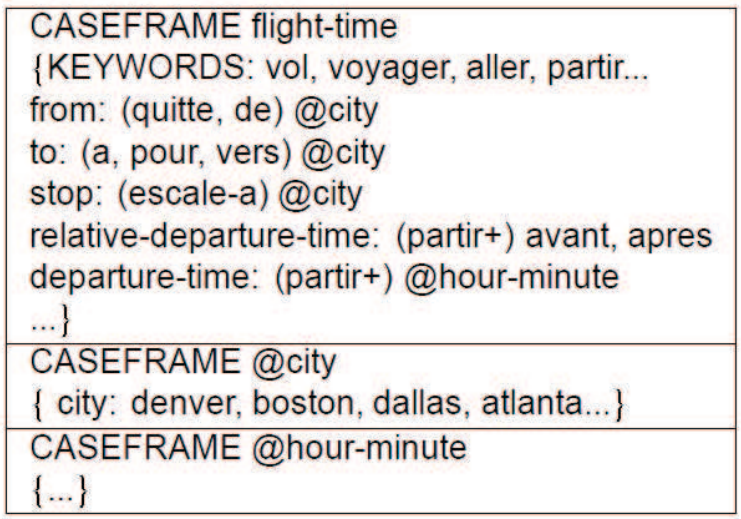


FIGURE 1.3 – *Exemple de cadre sémantique pour la tâche ATIS*

Le formalisme des cadres sémantiques servant de base à l'approche présentée dans

nos travaux, nous y reviendrons plus longuement dans la section 3.3.

Des exemples récents d'application du formalisme des grammaires à la problématique de la compréhension dans les systèmes de dialogues peuvent être trouvés dans (Villaneau et al., 2004; Denis et al., 2006). Dans (Denis et al., 2006), des grammaires d'arbres disjoints sont utilisées pour modéliser la tâche de dialogue. L'approche, basée sur l'analyse syntactique profonde (*deep-parsing*) et la logique de description, se décompose en 2 étapes :

- un analyseur LTAG (Crabbe et al., 2003) produit une analyse syntactique de la phrase. Seules les derivations partielles sont recherchées et les plus longues sont conservées ;
- un constructeur sémantique produit un graphe conceptuel à partir des analyses syntactiques précédentes. Le graphe conceptuel est ensuite réévalué en confrontation avec une ontologie interne, de sorte à éliminer les relations inconsistantes.

1.5 Grammaires stochastiques

Pour prendre en compte l'ambigüité d'analyse liée aux spécificités structurelles des messages oraux et surtout l'imprécision des transcriptions issues de la reconnaissance de la parole, les grammaires sémantico-syntaxiques présentées précédemment évoluent vers des grammaires stochastiques en utilisant un corpus d'apprentissage.

Des analyseurs CHART peuvent être utilisés pour stocker les forêts de sous-arbres représentant les résultats intermédiaires de l'analyse syntaxique lorsque les erreurs de reconnaissance ont empêché l'analyse complète d'aboutir. Une grammaire hors contexte probabiliste peut alors estimer la probabilité d'une analyse partielle. Cette estimation s'appuie sur un algorithme polynomial (en $O(n^3)$), ce qui rend possible l'utilisation de ces grammaires dans les systèmes de compréhension opérationnel (Corazza et al., 1994).

L'analyseur syntaxico-sémantique TINA développé à l'institut de technologie du Massachussetts (MIT) utilise ainsi une grammaire hors contexte probabiliste de type GRTA (Seneff, 1989). Cette grammaire est automatiquement convertie en un graphe dont les sommets sont les catégories syntaxiques ou sémantiques et les arcs portent les probabilités des règles, estimées sur un corpus d'apprentissage. Les constructions les plus fréquemment rencontrées sont donc privilégiées au cours de l'analyse. En cas d'échec de l'analyse complète, les analyses partielles sont réalisées à partir de chaque mot du message. D'autres approches à base de grammaires et d'analyse robuste ont été proposées, tel le Structured Language Model décrit dans (Chelba, 1997) et adapté à l'analyse sémantique par (Bod, 2000).

L'approche par analyse de surface de (Gildea et Jurafsky, 2002) permet de détecter les relations sémantiques contenues dans un message. Ces relations, nommées rôles sémantiques, sont formalisées par une représentation sémantique de haut niveau, indépendante de la tâche considérée.

Le système proposé par (Gildea et Jurafsky, 2002) est basé sur des classifieurs entrainés sur les phrases annotées manuellement en rôles sémantiques (selon les standards du projet FrameNet décrit en 3.4). Les phrases d'entraînement sont soumises à un analyseur syntaxique. Leurs sont alors associées leurs caractéristiques lexicales et syntaxiques ainsi que les probabilités a priori des différentes combinaisons des rôles sémantiques qu'elles contiennent. Les phrases testées sont analysées et annotées avec les caractéristiques extraites durant cette analyse puis soumises aux classifieurs, fournissant les étiquettes sémantiques de leurs constituants.

Ces méthodes sont reprises par (Pradhan et al., 2004) qui proposent un algorithme d'apprentissage pour l'analyse sémantique de surface basé sur des classifieurs à noyaux de type machines à vecteurs de support (séparateurs à vaste marge - SVM).

Dans le contexte de l'étiquetage sémantique, les travaux présentés dans (Moschitti, 2006; Moschitti et al., 2008) s'appuient sur l'utilisation de fonctions à noyaux adaptées aux traitement des arbres. L'analyseur utilisé génère un arbre syntaxique dans lequel les feuilles sont tout d'abord annotées. L'information sémantique est ensuite propagée vers la racine selon différentes stratégies pour produire un arbre syntaxico-sémantique initial. Tous ses sous-arbres sont alors extraits par des fonctions à noyaux spécifiques (*tree kernels*). L'usage de noyaux différents entraîne l'extraction de différents types de sous-arbres. Des classifieurs SVM, appris sur ces ensembles de sous-arbres, permettent de décider de l'arbre syntaxico-sémantique final à privilégier, de reclasser les hypothèses d'arbres et également d'évaluer les heuristiques de propagation de l'information sémantique.

1.6 Conclusion

Dans le cadre de la compréhension du dialogue oral, les approches strictement linguistiques fondées uniquement sur des grammaires sont donc souvent mises en défaut par la structure même du message véhiculé. En effet, les messages oraux ne sont pas formulés selon les mêmes normes que les messages écrits. Ces messages sont souvent agrammaticaux, contiennent des répétitions, des phrases inachevées. Une part importante des informations contenues dans les échanges oraux est perdue lors de leur transcription. Il en est ainsi par exemple des informations prosodiques indiquant souvent le mode de l'échange (interrogatif, affirmatif...) ou des informations implicites telles que les silences ou les hésitations.

De plus, les performances des systèmes de reconnaissance de la parole sont telles que de nombreuses erreurs émaillent encore les transcriptions automatiques des messages. Les méthodes d'analyse robuste présentées précédemment améliorent la qualité et la couverture des interprétations sémantiques obtenues par les approches linguistiques.

L'approche stochastique de la compréhension du dialogue oral est actuellement l'alternative principale aux méthodes linguistiques. Basée sur l'apprentissage, elle permet

de concevoir des systèmes mieux adaptés aux spécificités de l'oral et notamment aux erreurs de transcription. Cette approche est présentée dans le chapitre 2 suivant.

Chapitre 2

Approche stochastique

Sommaire

Résumé

Ce chapitre présente les approches stochastiques du problème de la compréhension. Le modèle théorique fondamental est détaillé en 2.2. Les applications classiques s'appuyant sur ce modèle sont exposées dans la section 2.3. La section 2.4 s'intéresse aux premiers systèmes de compréhension à base de réseaux bayésiens dynamiques qui ont inspiré les travaux présentés dans ce document.

2.1 Introduction

Les approches stochastiques de la compréhension du dialogue oral sont la principale alternative aux méthodes linguistiques à base de grammaires. Les méthodes stochastiques permettent de concevoir des systèmes adaptés aux spécificités de l'oral, facilement évolutifs, et robustes aux erreurs de transcription.

Basées sur le choix d'un modèle stochastique, adapté à la tâche visée, et l'apprentissage automatique de ses paramètres, ces méthodes réduisent les besoins en expertise humaine lors du développement d'applications tout en obtenant des résultats au moins comparables à ceux des méthodes à base de règles.

Le modèle théorique qui sous-tend les approches stochastiques de la compréhension est détaillé en 2.2. La section 2.3 s'intéresse à quelques systèmes de référence. L'approche à base de réseaux bayésiens dynamiques, qui a initié les travaux présentés dans ce document, est exposée dans la section 2.4.

2.2 Modèle théorique

L'approche stochastique de la compréhension est principalement basée sur le paradigme du *canal bruité* déjà utilisé pour formaliser le problème de reconnaissance de la parole (Jelinek, 1976). Ce paradigme est appliqué au problème de la compréhension sous deux hypothèses.

La première considère que le sens d'un message peut être exprimé par une séquence d'unités de sens en correspondance séquentielle avec les observations acoustiques (Pieraccini et al., 1993). Les unités sémantiques sont rassemblées dans un dictionnaire de *concepts*.

La seconde hypothèse consiste à considérer la représentation acoustique d'un message comme issue une séquence de concepts dégradée par un canal bruité de caractéristiques inconnues.

Sous ces deux hypothèses, le problème de la compréhension d'un message orale se ramène donc à un problème de décodage : il s'agit de déterminer la séquence conceptuelle $\hat{C}$ dont la probabilité *a posteriori* est maximale pour une séquence acoustique A observée.

Formellement, on a donc :

$$\hat{C} = \underset{C}{\operatorname{argmax}}\, P(C|A) = \underset{C}{\operatorname{argmax}} \sum_{W} P(W, C|A) \tag{2.1}$$

où $W = w_1, ..., w_L$ est la séquence de mots composants le message.

Le théorème de Bayes permet de renverser le conditionnement pour obtenir :

$$\hat{C} = \underset{C}{\operatorname{argmax}} \sum_{W} P(A|W, C)P(W, C) \tag{2.2}$$

Sous l'hypothèse d'indépendance entre la séquence acoustique observée A et la séquence de concepts C, l'équation précédente devient :

$$\hat{C} = \underset{C,W}{\operatorname{argmax}} P(A|W)P(W,C) \tag{2.3}$$

où A ne dépend plus que de la séquence de mots W.

La probabilité $P(A|W)$ étant évaluée dans le cadre de la reconnaissance de la parole, l'enjeu de la compréhension est donc la résolution de l'équation :

$$\hat{C} = \underset{C}{\operatorname{argmax}} P(W,C) = \underset{C}{\operatorname{argmax}} P(W|C)P(C) \tag{2.4}$$

La probabilité $P(W|C)$ d'une séquence de mots sachant une séquence conceptuelle représente le modèle de réalisation lexicale. Elle est généralement estimée par des probabilités n-grammes de mots conditionnées par le concept associé au mot courant et l'on a :

$$P(W|C) \backsimeq \prod_{i=1}^{L} P(w_i|w_{i-1},...,w_{i-n},c_i)$$

La probabilité $P(C)$, probabilité *a priori* d'une séquence de concepts, représente le modèle sémantique. Son estimation est également réalisée par des probabilités m-grammes de concepts selon :

$$P(C) \backsimeq \prod_{i=1}^{L} P(c_i|c_{i-1},...,c_{i-m})$$

Cette modélisation classique est une chaîne de Markov d'ordre n où seules les n dernières observations sont utilisées pour la prédiction du mot ou du concept suivant (i.e. un bigramme est une chaîne de Markov d'ordre 2).

On remarquera donc d'emblée que la formulation probabiliste du problème de la compréhension de la parole rend complexe une interprétation structurée des requêtes utilisateurs. La première hypothèse permettant l'expression sous forme d'un canal bruité impose la dépendance séquentielle des informations conceptuelles extraites. Or, il est clair que la sémantique d'une phrase présente très souvent des dépendances à long terme entre ses constituants. Les approches développées vont donc aborder ce problème de différentes manières : soit en l'occultant complètement par le biais d'une représentation purement "à plat" (qui peut se révéler suffisante pour un grand nombre d'applications), soit en composant différents niveaux de décodage permettant l'emboîtement des unités décodées (et devenant ainsi comparables à des grammaires non-contextuelles sémantiques probabilisées), soit finalement par le biais d'approches composites basées sur une structuration progressive de l'hypothèse.

2.3 Quelques applications

2.3.1 Le système CHRONUS

Un de premiers système basé sur le modèle stochastique présenté en 2.2 est le système CHRONUS (*Conceptual Hidden Representation Of Natural Unconstrained Speech*) (Pieraccini et al., 1991). Le décodeur conceptuel sur lequel repose CHRONUS utilise un modèle stochastique de type modèle de Markov Caché (*Hidden Markov Model*, HMM) dont les états représentent les concepts. Les séquences de mots associées à un concept donné sont également modélisées par un processus markovien représenté par un modèle de langage n-grammes conditionné par le concept. Les états du HMM conceptuel sont caractérisés par ces modèles de langage qui utilisent des techniques de repli (Riccardi et al., 1995) permettant d'attribuer des probabilités non nulles aux n-uplets non rencontrés dans les données observées.

Les paramètres du modèle sont appris à partir d'un corpus dans lequel les concepts sont associés à des portions de phrase. CHRONUS est évalué sur la tâche de renseignements dédiée aux voyages aériens ATIS (*Air Travel Information System*) décrite en détails dans (Dahl et al., 1994).

2.3.2 Le système CHANEL

Dans le système CHANEL (Kuhn et De Mori, 1995; De Mori, 1998), les règles d'interprétation sémantique sont apprises au moyen d'une forêt d'arbres de décision spécifiques appelés Arbres de Classification Sémantique (ACS). Contrairement à CHRONUS qui associe une séquence de mots à un concept, chaque ACS examine le message complet pour construire une partie de sa représentation sémantique. Chaque noeud d'un ACS est associé à une question binaire portant sur la présence ou non de séquences de mots prédéfinies. Ces questions sont générées et apprises automatiquement à partir de corpus dont les annotations préalables consistent uniquement en la liste des concepts présents dans chaque phrase. La sélection des questions s'appuie sur la maximisation de l'information d'un noeud à l'autre.

CHANEL est donc un système hybride dont l'originalité majeure est l'utilisation d'arbres de décision pour l'étiquetage sémantique.

2.3.3 L'approche HUM

Le modèle HUM (*Hidden Understanding Models*), proposé par (Miller et al., 1994), est inspiré des modèles stochastiques utilisés en reconnaissance de la parole et notamment des modèles de Markov cachés comme CHRONUS. Son objectif est de retrouver la structure sémantique la plus probable d'un message donné grâce à la résolution de l'équation (2.4).

La représentation sémantique est structurée sous forme d'arbres dont les nœuds non-terminaux sont les concepts abstraits et leurs composants, et les feuilles sont les mots du message. Un exemple d'arbre associé à une phrase de la tâche ATIS extrait de (Miller et al., 1994) est reproduit dans la figure 2.1.

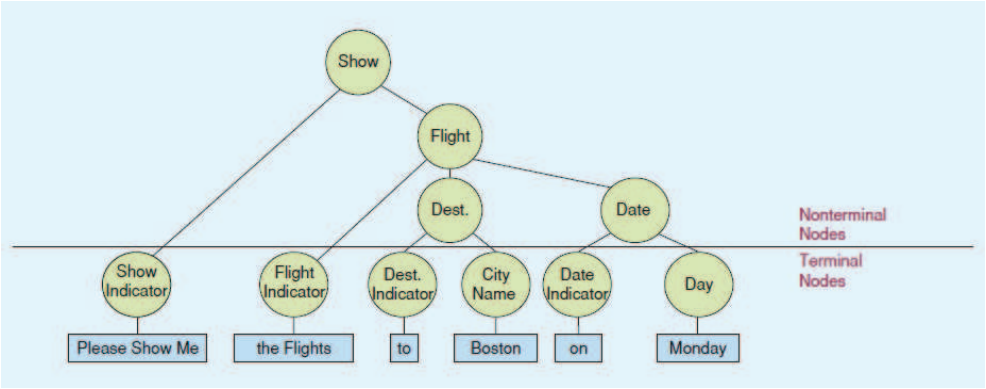


FIGURE 2.1 – *Exemple d'arbre associé à une phrase de la tâche ATIS*

Le modèle HUM est un modèle génératif basé sur deux composants stochastiques. Le premier composant, basé sur un modèle de langage sémantique, décide du sens à exprimer - i.e. *quoi dire* - tandis que le deuxième composant, basé sur un modèle de réalisation lexicale, sélectionne les séquences de mots adaptées à l'expression du sens - i.e. *comment le dire*.

Dans le modèle de langage sémantique, à chaque concept abstrait correspond un réseau de transition probabiliste contenant un état pour chacun de ses composants, un état d'entrée et un état de sortie. Un réseau est complet, autorisant ainsi tous les chemins sémantiques. Les probabilités de transition associées aux arcs du réseau sont obtenues en calculant $P(\acute{E}tat_n|\acute{E}tat_{n-1}, Concept)$. Les transitions entre états sont donc conditionnées par le contexte du concept courant qui permet de privilégier certains chemins par rapport à d'autres selon les paramètres collectés lors de l'apprentissage.

Les feuilles de l'arbre sémantique sont associées au modèle de réalisation lexicale qui repose sur les probabilités entre les mots, dans un contexte donné. Ces probabilités s'écrivent donc $P(Mot_n|Mot_{n-1}, Concept)$.

Le problème de la compréhension se ramène donc à la détermination du chemin le plus probable $\hat{T}$ dans la combinaison des réseaux sémantiques et lexicaux qui composent le message. Cette détermination repose sur la maximisation de la probabilité :

$$P(T) = \prod_{t \in T} \left[\begin{array}{ll} P(\acute{E}tat_n|\acute{E}tat_{n-1}, Concept) & \text{si} \quad t \in \{\text{Modèle de langage sémantique}\} \\ P(Mot_n|Mot_{n-1}, Concept) & \text{si} \quad t \in \{\text{Modèle de réalisation lexicale}\} \end{array} \right.$$

dont l'algorithme de calcul exacte est exponentiel par rapport à la longueur du message. Le calcul est donc réalisé par l'algorithme de Viterbi (programmation dynamique) aidé par la suppression des chemins de plus faibles probabilités (recherche en faisceau ou *bean search*).

2.3.4 Les systèmes HMM du LIMSI

Un premier système de compréhension stochastique est proposé par (Minker et al., 1996) et évalué par comparaison avec l'analyseur sémantique du LIMSI basé sur une grammaire de cas. Ce système utilise un modèle de Markov caché du premier ordre entrainé sur les données de la tâche ATIS. La représentation sémantique est voisine de celle utilisée par la grammaire de cas et permet l'extraction des valeurs associées aux concepts. Les performances de ce système restent modestes, incitant à l'introduction ultérieure d'informations contextuelles.

Plus récemment, (Maynard et Lefèvre, 2002) présente un système de compréhension stochastique développé sur des données collectées grâce au système de dialogue ARISE du LIMSI (Lamel et al., 2000). Ce système de dialogue est dédié à la réservation téléphonique de billets de train et propose également des renseignements sur les horaires, les tarifs et les prestations. Le système stochastique proposé par (Maynard et Lefèvre, 2002) utilise une représentation sémantique à plat, spécifique à la tâche. Le dictionnaire sémantique défini comporte 64 concepts auxquels valeurs normalisées et modalité (affirmative, négative) peuvent être associées. Ainsi, un énoncé est représenté par une liste de triplets [mode, concept, valeur normalisée].

Le modèle stochastique développé est conforme au modèle génératif présenté au paragraphe 2.2, utilisant des bigrammes conceptuels ($m = 1$ dans le modèle du paragraphe 2.2) et conditionnant le mot courant par le concept courant ($n = 0$ dans le modèle du paragraphe 2.2).

La comparaison des performances de ce système à celles obtenues par l'analyseur à base de grammaire de cas du LIMSI met en évidence sa robustesse aux erreurs de reconnaissance. Il est également montré que le système reste performant lorsque l'on réduit la taille du corpus d'apprentissage.

2.3.5 L'approche par FSM

Suivant une proposition initiale de (Pereira et Wright, 1997), une stratégie pour la compréhension de la parole basée sur l'utilisation de machines à états finis (*Finite State Machine*, FSM) est présentée en détails dans (Raymond et al., 2006). Son fonctionnement est résumé dans la figure 2.2, par la description du système MEDIA du LIA. L'interprétation débute par une transduction pour laquelle les modèles de langages stochastiques sont implémentés sous forme de FSM qui émettent des constituants sémantiques. Ces constituants correspondent aux concepts définis par l'ontologie de la tâche. A chaque concept est attaché une chaîne de mots qui lui sert de support et à partir de laquelle la valeur peut être obtenue (e.g. la date, les noms propres ou les informations numériques).

Un FSM est construit pour chaque concept élémentaire. Ces FSM sont des transducteurs qui prennent des mots en entrée et proposent en sortie les concepts supportés par les locutions reconnues. Ils peuvent être définis manuellement pour les concepts

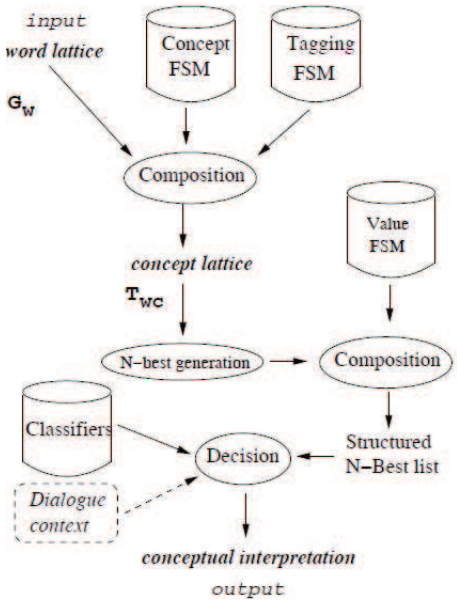


FIGURE 2.2 – *Exemple d'un système à base de FSM, le système* MEDIA *du* LIA

indépendants de la tâche (e.g. les dates ou les nombres) ou appris à partir des données fournies par un corpus d'apprentissage annoté. Tous ces transducteurs sont regroupés dans un unique transducteur, représentant leur union. Afin d'obtenir la meilleure séquence de concepts correspondant à la séquence de mots, un étiqueteur HMM, lui-même implémenté sous forme d'un FSM, est entraîné sur le corpus d'apprentissage. Enfin, une dernière étape de transduction est appliquée à chaque sous-séquence associée à un concept afin d'obtenir la valeur normalisée. Tous ces traitements peuvent être regroupés en appliquant les opérations adéquates sur les FSM intermédiaires. La disponibilité de l'AT&T FSM toolkit (Mohri et al., 2002) permettant l'implémentation de toutes les opérations usuelles sur les FSM est un grand atout de cette approche.

2.3.6 L'approche HVS

Le modèle HVS (*Hidden Vector State*), proposé par (He et Young, 2003, 2006), est dédié à l'analyse sémantique hiérarchique. Il est composé d'un modèle de Markov caché dans lequel chaque état représente un automate à pile de taille finie. La figure 2.3, issue de (He et Young, 2006), montre la séquence d'états du HSV correspondant à l'arbre d'analyse d'un message.

Les transitions entre états sont rassemblées dans des piles d'opérations d'entrée ou de sortie distinctes, limitées de façon à produire un espace de recherche calculable. Ce modèle est assez complexe pour capturer des structures hiérarchiques mais peut cependant être entraîné automatiquement à partir de données annotées sommairement.

Chaque état est représenté à l'instant t par un vecteur conceptuel c_t de dimension

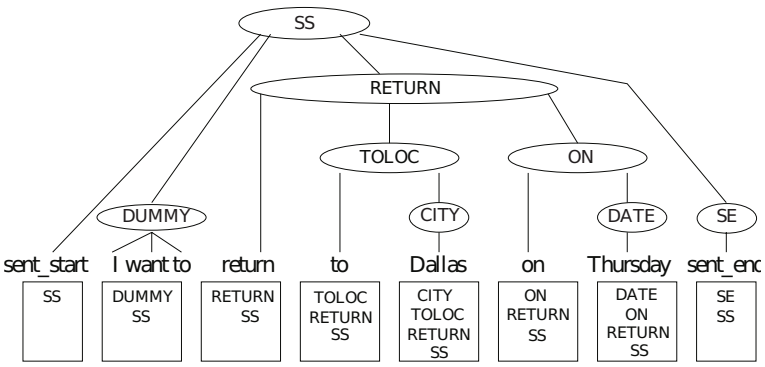


FIGURE 2.3 – *Arbre d'analyse d'un message et vecteurs d'états correspondants pour HVS*

D_t :

$$c_t = [c_t[1], c_t[2], \cdots, c_t[D_t]]$$

où $c_t[1]$ est le concept situé au sommet de la pile et $c_t[D_t]]$ est le concept racine (libellé "SS" dans la figure 2.3.

Pour une séquence de mots W, une séquence de vecteurs conceptuels C et une séquence d'opérations de sortie de la pile données, la probabilité jointe peut être décomposée sous la forme :

$$P(W, C, N) = \prod_{t=1}^{T} P(n_t|c_{t-1}) P(c_t[1]||[c_t[2 \cdots D_t]) P(w_t|c_t)$$

où n_t est le vecteur décrivant les opérations de modification de la pile, à valeurs dans l'intervalle $||[0, \cdots, D_t]||$ et $c_t[1] = c_{w_t}$ est le nouveau concept au sommet de la pile associé au mot w_t à l'instant t.

Le système nécessite donc l'apprentissage de trois tables de probabilités conditionnelles distinctes :

- $P(n|c)$, loi des opérations de sortie des concepts de la pile,
- $P(c_t[1]||[c_t[2 \cdots D_t])$, loi d'entrée d'un concept au sommet de la pile,
- $P(w|c)$, loi de génération des mots

Chacune de ces tables est estimée par un entraînement utilisant un algorithme de maximisation de la vraisemblance des paramètres des modèles (*Expectation-Maximization*, EM). De plus, pour garantir la calculabilité, les états non consistants avec le mot courant et ses concepts associés sont supprimés lors de l'apprentissage. Les tables obtenues sont ensuite utilisées pour produire les arbres d'analyse grâce à un décodage de Viterbi.

Une version étendue du modèle HVS a été proposée récemment qui présente l'avantage de permettre les branchements gauches et droits lors du décodage (alors que la version initiale ne permet que les branchements gauches) (Jurcicek et al., 2008). Une représentation par modèle graphique (ces modèles sont décrits plus en détails dans le chapitre 7) est donnée dans la figure 2.4. On notera la complexité du modèle qui engendre des difficultés pour l'apprentissage de ses paramètres. De même, l'extension du

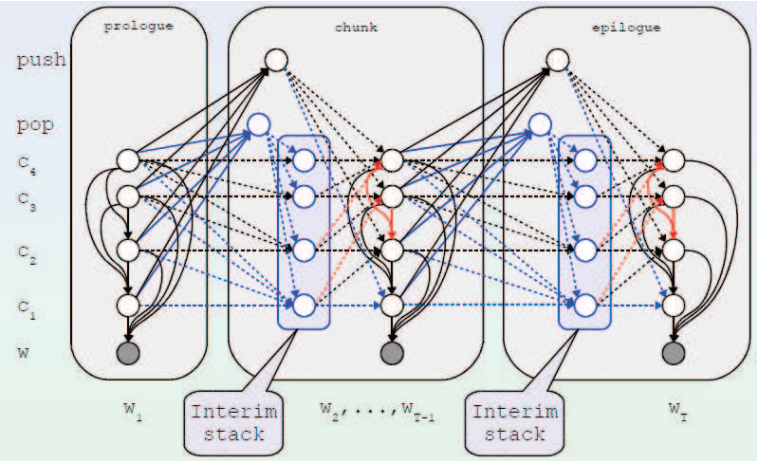


FIGURE 2.4 – *Représentation par modèle graphique du modèle HVS étendu avec insertion probabiliste (HVS-PP)*

modèle ne lui retire pas toutes ses limites ; ainsi le nombre de concepts pouvant être insérés simultanément (permettant le branchement droit) est codé dans la structure du modèle (limité à 3 dans le cas du modèle de la figure 2.4).

2.4 L'approche à base de réseaux bayésiens dynamiques

Un système de compréhension stochastique modélisé par des réseaux bayésiens dynamiques (*Dynamic Bayesian Networks*, DBN) est évoqué par (Bonneau-Maynard et Lefèvre, 2005) puis développé par (Lefèvre, 2006, 2007).

La représentation sémantique utilisée dans ce système est semblable à celle présentée dans (Maynard et Lefèvre, 2002) et détaillée en 2.3.4. Un énoncé est représenté par une liste de triplets [mode, concept, valeur normalisée].

Le modèle de compréhension intègre trois niveaux (mots W, concepts C et valeurs normalisées V), son objectif étant la recherche des séquences de concepts et de valeurs de probabilité *a posteriori* maximale selon :

$$\hat{C}, \hat{V} = \underset{C,V}{\operatorname{argmax}} P(C, V|W) = \underset{C,V}{\operatorname{argmax}} P(W|C, V)P(V|C)P(C) \tag{2.5}$$

La figure 2.5 présente le réseau bayésien dynamique utilisé par ce modèle pour deux mots successifs.

Le conditionnement par les valeurs, dont la liste est ouverte, augmente très sensiblement la complexité du modèle. Les auteurs résolvent ce problème en proposant un décodage en deux temps modélisé par un DBN à 2+1 niveaux selon :

$$\hat{C} = \underset{C}{\operatorname{argmax}} \sum_{V} P(W|C, V)P(V|C)P(C) \tag{2.6}$$

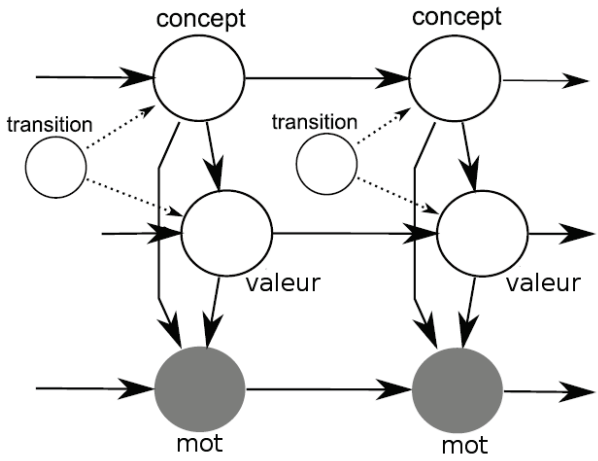


FIGURE 2.5 – *Modèle DBN à 3 niveaux*

$$\hat{V} = \underset{V}{\operatorname{argmax}}\, P(\hat{C}, V|W) = \underset{V}{\operatorname{argmax}}\, P(W|\hat{C}, V)P(V|\hat{C})P(\hat{C}) \tag{2.7}$$

Les concepts étant décodés par un premier DBN à partir des mots observés. Les valeurs sont obtenues par un second décodage utilisant les séquences de concepts produites lors de la première phase comme des observations. Les réseaux bayésiens dynamiques utilisés sont représentés figure 2.6

Les résultats encourageants fournis par ce système sur la tâche MEDIA 4 ont motivé les travaux présentés dans ce document, explorant la capacité de ces modèles à intégrer un système de compréhension de haut niveau.

2.5 Conclusion

Principales alternatives aux méthodes à base de règles, les méthodes stochastiques sont particulièrement adaptées à la tâche de compréhension de la parole. Dédiées à la modélisation de l'incertitude, ce sont par nature des méthodes robustes et peu dépendantes de l'application visée. Le coût de leur développement est essentiellement lié à la mise en forme de modèles théoriques par un ou plusieurs experts. Il est en cela très inférieur à celui des méthodes à base de règles. Les méthodes stochastiques étant basées sur l'apprentissage, leur emploi est cependant conditionné à la disponibilité de corpus d'entraînement de taille suffisante et cohérent avec l'espace sémantique visé.

Dans le contexte de la compréhension de la parole, méthodes à base de règles et méthodes stochastiques sont également dépendantes de la représentation sémantique choisie pour modéliser le sens des messages. Les travaux contemporains abordant cette problématique sont présentés dans le chapitre 3.

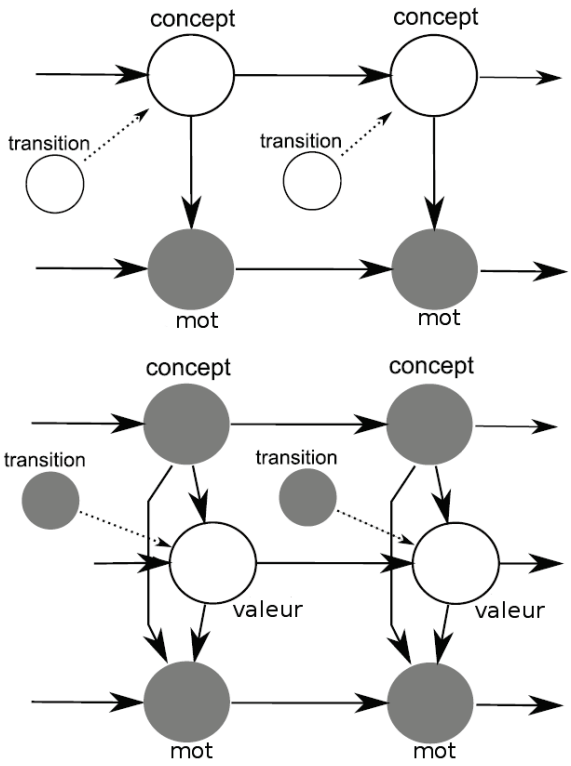


FIGURE 2.6 – *Modèle DBN à 2+1 niveaux*

Chapitre 3

Représentation sémantique

Sommaire

Résumé

Ce chapitre présente les quelques formalismes existants pour la représentations sémantiques, développés dans le cadre de la modélisation de la compréhension.

3.1 Introduction

Les théories et formalismes sémantiques présentés dans ce chapitre s'inscrivent dans un cadre applicatif contemporain. Le sens d'un message est envisagé d'un point de vue formel et non fondamental. Il s'agit en effet de représenter les informations présentes dans ce message sous une forme cohérente et apte à renseigner un système sur les attentes de son interlocuteur.

Le cadre théorique des représentations sémantiques présentées est donc celui de la sémantique procédurale (Woods, 1981) : le sens d'un symbole est portée par une procédure abstraite liant l'expression symbolique au monde réel par l'intermédiaire d'opérations calculatoires et inférentielles réalisées par un interpréteur. Le sens d'un message est alors approché d'un point de vue procédural : le monde réel est représenté par l'état du système, le module de compréhension est l'interpréteur reliant le message reçu aux fonctionnalités du système.

Ce module doit donc s'appuyer sur une représentation structurée pour extraire le sens formel d'un message. Cette représentation est un langage possédant syntaxe et sémantique, basé sur le principe de compositionnalité présenté en I. Les connaissances sémantiques d'une application forment une *base de connaissances* qu'il est possible de représenter à l'aide d'un ensemble de formules logiques, généralement du premier ordre. Ces formules contiennent des variables potentiellement typées et instanciées par des constantes liées au domaine de l'application. Dans ce contexte, les objets sémantiques sont définis par l'instanciation de toutes les variables d'une formule ou par la composition d'objets existants. La génération de ces objets sémantiques est donc le fait d'un processus inférentiel porté par le module de compréhension.

La section 3.2 présente les notions clés de la représentation des relations sémantiques. La théorie et le formalisme des cadres sémantiques, fondés sur cette représentation, sont explicités en 3.3. Le projet FrameNet, dont les principes ont été utilisés dans ce travail, est détaillé dans la section 3.4.

3.2 Réseaux sémantiques

La représentation des relations sémantiques par des liens entre classes et objets sémantiques est discutée dans (Woods, 1975). Les formules de la base de connaissances d'une application décrivent des concepts et leurs relations qui peuvent être représentés par un *réseau sémantique*. Ce réseau est composé de nœuds matérialisant les concepts et d'arcs correspondant aux relations inter-conceptuelles (Brachman, 1979). Cette structure permet de modéliser à la fois connaissances factuelles et relationnelles, telles les relations de composition étudiées par (Jackendoff, 1990).

Le plus célèbre langage développé pour représenter des structures de type réseaux sémantiques est le langage KL-ONE (Brachman et Schmolze, 1985) dont l'élément central est le concept. Une base de connaissances KL-ONE est un réseau sémantique dans

lequel les concepts génériques sont fortement hiérarchisés. Les concepts sont les composants à partir desquels l'interprétation du message est réalisée. Ils sont définis par un ensemble d'attributs descriptifs et relationnels, les *rôles sémantiques*.

3.3 Cadres sémantiques

Les concepts et relations d'un réseau sémantique peuvent être implémentés en utilisant le formalisme des *cadres sémantiques* présenté dans (Fillmore, 1985) dans la logique des *cadres de cas* (Fillmore, 1968). Un *cadre* définit tout système relationnel de concepts au sein duquel la compréhension d'un concept fait appel à la compréhension du système complet.

Dans ce contexte, Fillmore définit les cadres sémantiques comme des structures cognitives empiriques associées au processus de compréhension (Fillmore, 1982, 1985). Les cadres sémantiques sont rassemblés dans une grammaire de cadres. Une telle grammaire génére des cadres décrivant des concepts généraux et leurs instances spécifiques. Un cadre sémantique est une structure de données représentant un concept en associant à son nom un ensemble d'éléments décrivant ses rôles situationnels (attributs) ou relationnels (rôles sémantiques).

Les mots ou groupes de mots associés aux cadres sémantiques représentent une catégorie d'expériences (situationnelles, événementielles...) liées au monde réel. Ils évoquent les cadres auxquels ils appartiennent lorsqu'ils sont présents dans un message. L'interpréteur peut alors invoquer ces cadres pour attribuer une interprétation au message (Petruck, 1996).

Un exemple d'instance de cadre sémantique est donné dans le tableau 3.1.

```
{idt0001929
instance_de        identite
  nom                Levi-Strauss
  prenom             Claude
  sexe               masculin
  date_naissance     28.11.1908
  lieu_naissance     Bruxelles
}
```

TABLE 3.1 – *Exemple d'instance du cadre sémantique* `identite`.

Le cadre sémantique présenté dans cet exemple associe au concept `identité` les éléments `nom`, `prenom`, `sexe`, `date_naissance` et `lieu_naissance`. Certains de ces éléments sont à leur tour des instances d'autres cadres sémantiques (`date` ou `lieu` dans notre exemple). La sémantique procédurale de (Woods, 1981) peut définir le mode de génération d'instances de cadres sémantiques : des procédures conditionnelles sont alors associées aux éléments des cadres. Ces procédures peuvent générer, supprimer ou modifier des cadres par inférences sur les cadres existants.

Les cadres sémantiques représentant des catégories d'expériences, leur définition - concepts et éléments associés - est fondamentalement dépendante de la tâche à traiter. Un certain nombre de projets tentent cependant de rassembler des ressources génériques. Le *répertoire de cas* de Fillmore (Fillmore, 1968), *TreeBank* (Marcus et al., 1994), *PropBank* (Kingsbury et Palmer, 2003) et surtout le projet *FrameNet* en sont quelques exemples.

Le formalisme sémantique utilisé dans ce travail étant inspiré de FrameNet, ce projet est détaillé dans la section suivante 3.4.

3.4 FrameNet

Le projet FrameNet (`http://framenet.icsi.berkeley.edu/`) de l'Université de Berkeley fournit une base de données de *frames* [1], cadres sémantiques pour la langue anglaise (Baker et al., 1998; Fillmore et al., 2003). Dans l'esprit des cadres sémantiques de Fillmore (Fillmore, 1982), les frames sont des représentations schématiques de situations. L'objectif du projet est la définition de frames alliant généricité et spécificité pour permettre leur utilisation dans des applications variées.

A chaque frame est associé de manière unique des rôles appelés *frame elements* (FE). Certains FE sont indispensable à l'instanciation de la frame, d'autres sont optionnels. La base de données construite dans le cadre du projet met en relation les frames, leurs FE et les unités lexicales (mots ou groupe de mots) qui les évoquent. Actuellement, celle-ci contient 963 frames reliées hiérarchiquement et plus de 10.000 unités lexicales (LUs). Une ressource de 135.000 propositions annotées à l'aide de ces frames et de FE est également disponible dans le cadre du projet.

L'exemple de la frame **Cogitation** est donnée ci-dessous :

```
COGITATION

Definition:
A person, the Cognizer, thinks about a Topic over a period
of time. What is thought about may be a course of action
that the person might take, or something more general.

    ex: The men were silently MULLING OVER the proposition
        of committing an assassination

FEs:
Core:

Cognizer [Cog]  With a target verb, the Cognizer is usually
```

1. Par habitude, nous préférons parler de frame sémantique. Mais les deux termes, frame et cadre, sont strictement équivalents pour nous.

```
Semantic Type   expressed as an External Argument, with the
Sentient        Topic appearing as an Object NP,  a gerundive
                verbal Complement, or a PP.
                ex: ...

Topic [Top]     With a target verb,the Topic is usually expressed
                as an Object NP, a gerundive verbal complement,
                or a PP.
                ex: ...

Non-Core:

Degree [Degr]   The FE Degree indicates the degree to which the
Semantic Type   cognizing occurs.
Degree

Depictive [Dep] Depictive phrase describing the actor of an action


Manner [Manr]   The FE Manner indicates the way in which the
Semantic Type   cognizing is being done.
Manner

Means [Mns]     An intentional action performed by the Cognizer
Semantic Type   that makes them able to cogitate.
State_of_affairs

Medium [Medium] The physical or abstract setting in which the
                Cognizer considers the Topic.


Purpose [pur]  The state-of-affairs that the Cognizer is trying to
               bring about by thinking.

Result [Result] Result of an event

Time [tim]      The time at which the Cognizer considers the Topic.


Inherits From:
Is Inherited By: Assessing, Emotion_active, Memorization
Subframe of:
Has Subframes:
Precedes:
Is Preceded by:
```

```
Uses: Mental_activity
Is Used By: Remembering_experience, Research
Perspective on:
Is perspectivized in:
Is Causative of:
See Also:

Lexical Units

brood.v, consider.v, consideration.n, contemplate.v, contemplation.n,
deliberate.v, deliberation.n, dwell.v, give, thought.v, meditate.v,
meditation.n, mull_over.v, muse.v, ponder.v, reflect.v, reflection.n,
ruminate.v, think.v, thought.n, wonder.v
```

Cette frame modélise une situation où une personne (Cognizer) pense à un sujet (Topic) pendant une période de temps. Le sujet de réflexion peut être relatif au déroulement d'une action impliquant la personne ou plus général. Deux FE principaux sont associés à la frame `COGITATION` : `Cognizer`, la personne qui réfléchit et `Topic`, le sujet de réflexion. D'autres FE secondaires lui sont également rattachés (`Depictive`...), détaillant la situation selon divers points de vue.

Un extrait du contexte relationnel de cette frame au sein de la base FrameNet est illustré par la figure 3.1.

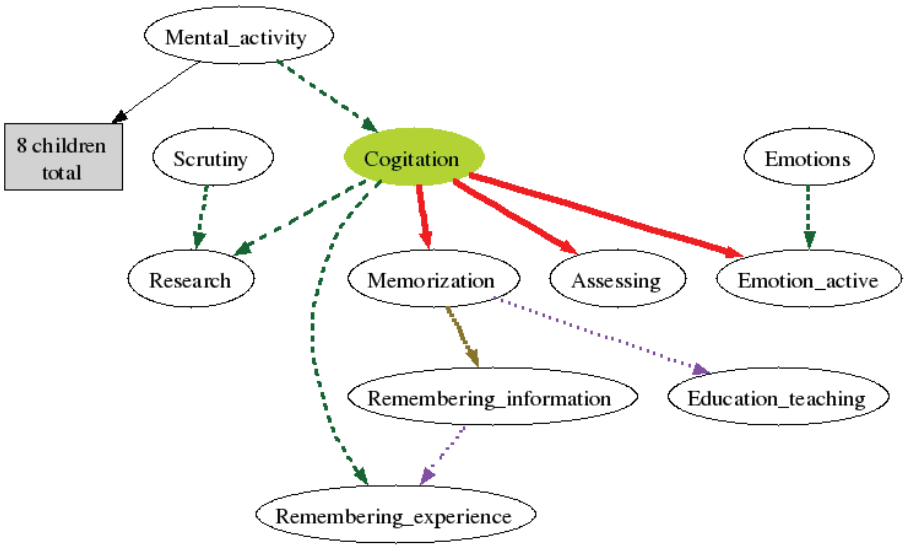


FIGURE 3.1 – *Extrait des relations liant la frame `COGITATION` à d'autres frames de FrameNet*

La liste de LUs associée à la frame `COGITATION` montre qu'une frame peut être évoquée par des LUs de différentes natures (ici, noms et verbes). Il en est de même pour les FE. Cette souplesse est un atout pour l'exploitation de ce formalisme dans les systèmes de compréhension automatique.

Par opposition aux représentations sémantiques “à plat”, la représentation en frames du sens d’un message est une représentation structurée. En effet, les représentations planes associent un concept de base à chaque segment du message mais ne composent pas les concepts ainsi obtenus. L’usage des frames définie dans FrameNet produit en revanche une représentation hiérarchiquement structurée et donc très adaptée à la tâche de composition sémantique.

3.5 Conclusion

Les capacités d’un système de compréhension sont dépendantes de la représentation sémantique choisie. Initiés par la sémantique procédurale de Woods (Woods, 1981), les réseaux sémantiques figurent parmi les premiers modèles de représentation des relations inter-conceptuelles.

En introduisant la notion et le formalisme des cadres sémantiques, Fillmore (Fillmore, 1985) propose des objets sémantiques situationnels. Les cadres sémantiques peuvent être combinés au sein de structures relationnelles pour représenter une situation réelle.

Sous-tendu par ce formalisme, le projet FrameNet s’attache à définir des objets sémantiques plus génériques pour favoriser leur emploi dans des contextes applicatifs variés. Les frames sémantiques, leurs frame-éléments et les unités lexicales qui les évoquent sont rassemblés dans une base de données qui met en évidence les relations hiérarchiques entre les objets.

La définition par unités lexicales et la structure hiérarchique des frames en font des objets sémantiques particulièrement bien adaptés à la représentation du sens d’un message dans le contexte du dialogue. En effet, ces caractéristiques les rendent aptes à évoluer sans remise en cause fondamentale et à supporter la composition sémantique avec finesse.

Chapitre 4

Matériau expérimental : le corpus MEDIA

Sommaire

Résumé

Ce chapitre propose une présentation du corpus MEDIA *qui a servi de matériau d'expérimentation et d'évaluation à ce travail. Composé de dialogues en français issus de la simulation d'un serveur téléphonique d'informations touristiques et de réservation d'hôtel,* MEDIA *a été manuellement transcrit et annoté à l'aide de structure sémantiques de type attribut-valeur. La section 4.2 détaille le mode d'obtention et les caractéristiques des dialogues composant le corpus. Les différentes transcriptions et annotations du corpus sont ensuite présentées en 4.3. Enfin, la section 4.4 rapporte les résultats des mesures de qualité effectuées sur le corpus.*

4.1 Introduction

Le corpus ayant servi de materiau d'expérimentation et d'évaluation à ce travail est un corpus de dialogues en français, produit dans le cadre du projet MEDIA (Maynard et al., 2004). L'objectif de ce projet était de tester une méthodologie d'évaluation de la compréhension hors et en contexte des systèmes de dialogue basée sur le paradigme PEACE (*Paradigme d'Evaluation Automatique de la Compréhension hors et en contexte dialogique*) (Devillers et al., 2002), fondé sur la constitution de batteries de tests reproductibles issues de dialogues réels.

La section 4.2 détaille le mode d'obtention et les caractéristiques des dialogues composant le corpus MEDIA. Les différentes transcriptions et annotations du corpus sont ensuite présentées en 4.3. Enfin, la section 4.4 rapporte les résultats des mesures de qualité effectuées sur le corpus.

4.2 Collecte du corpus

Le corpus MEDIA est dédié à l'étude des applications de demande de renseignements accédant à des bases de données. Il est composé de dialogues en français issus de la simulation d'un serveur téléphonique d'informations touristiques et de réservation d'hôtels.

Ces dialogues ont été collectés en utilisant le protocole du *Magicien d'Oz* (*Wizard of Oz*, WoZ). Lors de l'échange, les utilisateurs croient converser avec une machine alors que le dialogue est en fait pris en charge par un opérateur humain qui simule les réponses d'un serveur d'information et de réservation. L'opérateur est assisté par l'outil WoZ dans la génération des réponses à fournir à l'utilisateur. Les informations relatives à la tâche sont issues de la consultation par l'opérateur du site de réservation d'hôtels de la chaîne ACCOR [1] et du site d'informations touristiques "Tourisme en France" [2].

Le protocole de collecte des dialogues est illustré par le schéma 4.1, emprunté au "Manuel d'utilisation de l'outil WoZ. Projet MEDIA".

Après chaque phrase de l'utilisateur, l'opérateur consulte l'outil WoZ qui lui propose la réponse à fournir en fonction du nouvel état du dialogue. Pour diversifier les réponses de l'opérateur, l'outil WoZ est paramétré au niveau des messages, des consignes et des scenarii. Un ensemble de messages est associé à l'application pour varier les formulations des réponses. A chaque appel, l'opérateur doit respecter une série de consignes (par exemple, faire semblant de ne pas avoir compris l'utilisateur pour simuler les erreurs que ferait un système réel). Ces consignes doivent être fournies à l'outil WoZ et dépendent du scenario choisi pour le dialogue à enregistrer.

1. http ://www.accorhotels.com
2. http ://www.tourisme.fr

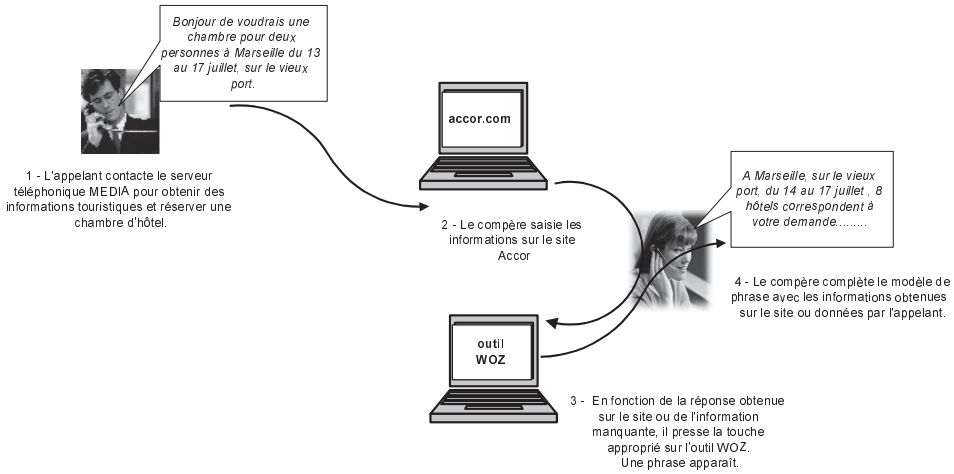


FIGURE *4.1* – *Protocole du Magicien d'Oz*

L'outil WoZ ne permet pas la gestion de l'ensemble du dialogue. L'opérateur doit gérer l'intégration des informations pratiques aux schémas de réponses proposés par le WoZ. Il doit également prendre en charge toutes les opérations de saisie, de correction, de négation d'items ainsi que l'énonciation orale de la réponse à l'utilisateur.

Les dialogues ont pour but la réservation d'une ou plusieurs chambres dans un ou plusieurs hôtels. Les réservations s'effectuent dans le cadre de l'organisation d'un week-end, de vacances ou d'un séjour professionnel. Ainsi, les dialogues peuvent débuter à partir de thèmes variés : choix d'une ville de séjour, recherche d'un itinéraire ou d'un événement touristique, satifaction d'une contrainte de prix ou de date (Bonneau-Maynard et al., 2005). Huit catégories de scenarii ont été définies avec différents niveaux de complexité, chaque locuteur ayant enregistré cinq scenarii différents. Un extrait de dialogue est présenté dans le tableau 4.1.

Le corpus MEDIA est composé de 1257 dialogues produits par 250 locuteurs, pour une durée totale d'environ 70 heures d'enregistrement audio. L'ensemble de ces dialogues contient un total de 15.000 requêtes réparties sur les portions d'entraînement, de développement et de test du corpus comme indiqué dans le tableau 4.2.

Une description détaillée des caractéristiques techniques du corpus est donnée dans le tableau 4.3.

La collecte du corpus et la gestion technique du projet MEDIA ont été prises en

WoZ	Bienvenue sur le serveur MEDIA système d'informations touristiques et de réservations d'hôtel. Quelle information désirez-vous ?
Locuteur	Je souhaite réserver un hôtel à Nancy
WoZ	A Nancy, quelles sont vos dates de séjour ?
Locuteur	Une nuit, le quinze juillet deux mille quatre
WoZ	Dans quelle partie de la ville souhaitez vous que votre hôtel se situe ?
Locuteur	Je n'ai pas d'exigence particulière.
WoZ	Deux hôtels correspondent à votre demande...

TABLE *4.1* – *Extrait de dialogue du corpus* MEDIA.

portion du corpus	nombre de requêtes
entraînement	11.010
développement	1.009
test	3.005

TABLE *4.2* – *Nombre de requêtes sur les différentes portions du corpus* MEDIA.

nombre de mots moyen par phrase utilisateur	8,3
nombre de mots moyen par phrase système	14,4
taille du vocabulaire utilisateur	2.715 mots
taille du vocabulaire système	1.932 mots
durée moyenne d'un dialogue	3 min et 30 s

TABLE *4.3* – *Caractéristiques du corpus* MEDIA.

charge par ELDA/ELRA [3]. La société VECSYS [4] a mis en place la plate-forme d'enregistrement du corpus (matériel et outil WoZ).

4.3 Transcription et annotation du corpus

L'annotation sémantique du corpus a nécessité la définition d'une représentation sémantique adaptée au domaine de la tâche MEDIA. Cette représentation est générique, assure une bonne couverture du domaine et est cependant suffisament simple pour permettre l'annotation d'un corpus de la taille de MEDIA. Elle est basée sur une structure de type attribut-valeur dans laquelle les relations conceptuelles sont représentées implicitement par le nom des attributs. Les attributs sont donc les concepts liés au domaine.

Le dictionnaire sémantique utilisé associe à un mot (ou un groupe de mots) une paire *concept-valeur* puis un spécifieur définissant des relations entre concepts et enfin un *mode* (affirmatif, négatif, interrogatif ou optionnel) attaché au concept. Avec 19 spé-

3. http ://www.elda.org
4. http ://www.vecsys.fr

cifieurs pouvant être associés aux 83 concepts de base, le schéma d'annotation MEDIA offre un mécanisme simple permettant de préserver certaines relations élémentaires entre les concepts au sein de la phrase.

Chaque tour de parole du locuteur est scindé en segments sémantiques correspondant à une unique paire.

Un exemple de message annoté du corpus MEDIA est donné dans le tableau (4.4). La première colonne contient les séquences de mots W^c supports de chaque concept, présenté dans la seconde colonne. La troisième colonne indique le mode et la quatrième colonne fournit les spécifieurs associés aux concepts. La dernière colonne présente les valeurs normalisées des concepts c associés aux séquences W^c.

W^c	*concept c*	*mode*	*spécifieur*	*valeur*
Je voudrais réserver	`commande`	+		réservation
une chambre	`chambre-quantité`	+	réservation	1
pour deux nuits	`séjour-nbNuit`	+	réservation	2
à Marseille	`localisation-ville`	+	hôtel	Marseille

TABLE 4.4 – *Exemple d'annotation sémantique du corpus* MEDIA.

Dans cet exemple, le spécifieur *réservation* est lié aux concepts `chambre-quantité` et `séjour-nbNuit`. On obtient ainsi une structure hiérarchique représentant une réservation associée au concept `commande` et développée grâce aux valeurs des concepts `chambre-quantité` et `séjour-nbNuit`. Le spécifieur *hôtel* adjoint au concept `localisation-ville` permet de relier le lieu évoqué dans le segment *"à Marseille"* à la partie précédente de l'énoncé.

La combinaison des spécifieurs et des concepts permet de recomposer un premier niveau de représentation hiérarchique de la requête du locuteur à partir de l'annotation *à plat*.

4.4 Qualité du corpus : l'accord inter-annotateur

Le corpus MEDIA est transcrit manuellement et enrichi par une annotation conceptuelle également manuelle réalisée par deux annotateurs de la société ELDA.

Pour déterminer la qualité des annotations, l'accord entre les différents annotateurs a été évalué. Cet accord inter annotateur (IAg pour Inter-annotator Agreement) est mesuré en utilisant la mesure de Kappa k telle que :

$$k = \frac{P(A) - P(E)}{1 - P(E)}$$

où $P(A)$ est le rapport du nombre de fois où les annotateurs sont d'accord sur le nombre total d'annotation et $P(E)$ la probabilité que l'annotation correcte ait été posée par hasard.

Le coefficient de Kappa et ses différents modes de calcul sont présentés dans (Siegel et N.J. Castellan, 1988). La mesure de Kappa est détaillée et discutée dans (Carletta, 1996). Il est admis dans la littérature que la fiabilité des annotations est bonne dès lors que l'IAg mesuré par le coefficient de Kappa est supérieur à 80%. Les IAg mesurés dans le cadre du projet atteignent presque 90% dans la phase finale du projet (Bonneau-Maynard et al., 2005), ce qui permet de valider la qualité des annotations dans le corpus. Le tableau 4.5 présente les résultats des IAg mesurés au cours de la phase finale d'annotation du corpus.

Evaluation	1	2	3	4
Nb de dialogues	10	10	10	10
Nb de tours de parole locuteur	165	137	106	163
Nb de segments sémantiques	372	455	342	459
IAg (%)	89,5	83,1	83,9	87,8

TABLE 4.5 – *IAg finales obtenues sur l'annotation du corpus* MEDIA.

4.5 Conclusion

Le corpus de dialogues MEDIA est le matériau d'expérimentation de ce travail. Les dialogues MEDIA ont pour but la réservation d'hôtels et l'obtention d'informations touristiques. Ils sont collectés en utilisant le protocole du Magicien d'Oz dans lequel un opérateur humain assisté d'un outil d'aide à la décision simule les réponses d'un serveur téléphonique.

Le corpus contient 1257 dialogues pour environ 15.000 requêtes utilisateur. Transcrit manuellement, chaque dialogue est également annoté sémantiquement à l'aide de concepts de base. Les requêtes utilisateur sont scindées en segments sémantiques (d'un ou plusieurs mots) auxquels est attribuée une paire concept-valeur.

Le dictionnaire de concepts MEDIA comporte 83 entités. L'annotation est enrichie par l'indication de relations entre les concepts de base du message et le mode de la proposition. Les annotations sémantiques associées aux requêtes ont été conçues pour pouvoir être utilisées par le module de compréhension d'un système de dialogue.

La qualité d'annotation du corpus MEDIA, évaluée tout au long de la phase de travail manuel des experts, est attestée par un IAg final supérieur à 85%. Les données MEDIA représentent donc un support d'expérimentation conséquent et fiable.

Deuxième partie

CONTRIBUTIONS

Introduction

Le cœur de cette étude est le développement de modèles stochastiques adaptés à la composition sémantique pour la compréhension automatique dans le contexte du dialogue oral. La première partie du travail a consisté en la production de données pour l'apprentissage des paramètres des modèles.

La production de ces données a nécessité le choix d'une représentation sémantique adaptée puis la création d'une base de connaissances d'objets sémantiques (frames et FE) dédiée au domaine du corpus MEDIA. Un sous-ensemble de dialogues du corpus a été manuellement annoté pour devenir l'ensemble des dialogues de référence. Ces travaux sont présentés dans le chapitre 5.

La base ontologique associée au domaine MEDIA étant définie, un système à base de règles et d'inférences logiques a permis de produire automatiquement les annotations sémantiques en frames et FE sur les dialogues du corpus dédiés à l'apprentissage des modèles stochastiques. Le système développé à cette fin est décrit dans le chapitre 6.

La seconde partie 6.5 présente les modèles stochastiques à base de réseaux bayésiens dynamiques (DBN) conçus pour la génération de fragments de structures sémantiques. Les bases théoriques définissant les DBN sont rappelées dans le chapitre 7. Les différents modèles proposés sont décrits dans le chapitre 8. Les expériences menées et les résultats obtenus sont détaillés au chapitre 9.

Enfin la troisième partie 9.4 présente les stratégies de recomposition de structures sémantiques appliquées aux fragments générés par les systèmes DBN. Ces stratégies utilisent la notion d'arbre. L'une est heuristique tandis que l'autre s'appuie sur une méthode de classification à base de séparateurs à vaste marge. Les définitions et propriétés nécessaires à la maîtrise de ces notions sont rappelées au chapitre 10. Les deux stratégies sont présentées dans le chapitre 11. Les expériences menées et les résultats obtenus sont détaillés au chapitre 12.

Production des données d'apprentissage

Chapitre 5

Représentation sémantique

Sommaire

Résumé

Ce chapitre décrit concrétement la représentation à base de frames sémantiques choisie, ainsi que l'ontologie mise au point pour le système de compréhension stochastique développé dans ce travail.

5.1 Introduction

Comme décrit dans le chapitre 4, l'annotation du corpus MEDIA fournit des étiquettes comparables aux constituants proposés par un analyseur sémantique de surface. Cependant, pour obtenir une représentation hiérarchique complète de la composition sémantique d'une proposition, l'utilisation de structures plus riches et plus complexes est nécessaire.

Les propriétés des frames sémantiques, exposées en 3.4, ont conduit à privilégier leur emploi dans ce travail. Cette approche est détaillée dans la section 5.2. La base de connaissances constituée pour couvrir le domaine du corpus MEDIA est présentée en 5.3. La section 5.4 explicite le processus d'annotation manuelle en frames d'un ensemble de dialogues de test. Une version étendue de la base de connaissance est mentionnée dans la section 5.5.

5.2 Frames Sémantiques

Le choix d'une annotation en frames et FE (frame elements) dans ce travail est motivé par leur capacité à représenter des dialogues de négociation et à s'adapter aux actions complexes du gestionnaire de dialogue. Une frame décrit une situation concrète ou abstraite impliquant ses FE. Les mots ou groupes de mots déclenchant l'instanciation d'une frame ou d'un FE sont ses *unités lexicales* (LU). Ces LU associent un mot (ou une séquence de mots) à un sens.

Le projet FrameNet de l'université de Berkeley, décrit en 3.4, fournit une base de données de frames pour la langue anglaise (Fillmore et al., 2003). Actuellement, celle-ci contient environ 963 frames reliées hiérarchiquement et plus de 10.000 LU. Une ressource de 135.000 propositions annotées à l'aide de ces frames et de leurs FE est également disponible dans le cadre du projet.

Une base de données comparable est disponible pour la langue française (Pado et Pitel, 2007). Les travaux de Pado et Pitel s'appuient sur la *projection d'annotations* automatique présentée par (Padó et Lapata, 2005, 2006) dans le contexte de langues au parallélisme sémantique suffisant. Obtenue par projection *cross-linguistique* anglais-français, l'utilisation du FrameNet français de (Pado et Pitel, 2007) a été écartée au profit d'une définition manuelle des frames pour plusieurs motifs :

- l'usage d'une base de connaissances de taille plus réduite mais couvrant parfaitement le domaine du corpus MEDIA limite la confusion lors de l'annotation ;
- la nature particulière du support textuel (transcriptions de parole) induit des différences importantes dans la définition des LU et CU (*unités conceptuelles* de base) ;
- les frames de FrameNet sont potentiellement trop génériques pour satisfaire les besoins d'un système de dialogue automatique.

Nous avons donc défini manuellement un ensemble de frames et de FE pour décrire nos connaissances en terme de composition sémantique adaptée au corpus MEDIA. Cette base de connaissances contient 21 frames et 86 FE. Elle est donc de taille modeste

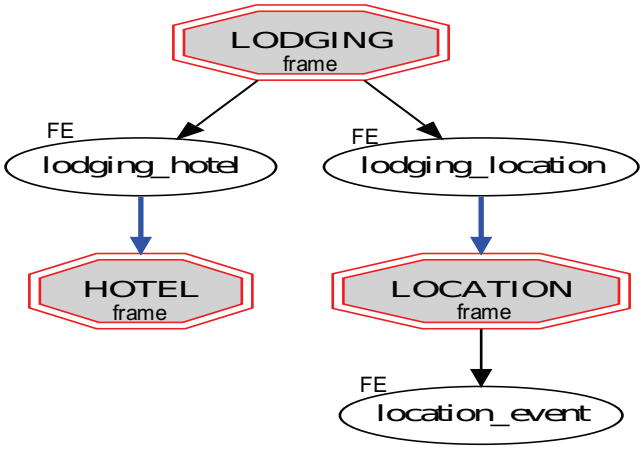


FIGURE 5.1 – *frames, FE et relations associés à la séquence de mots "séjourner dans un hôtel proche du Festival de Cannes".*

en comparaison des versions françaises de FrameNet (Pado et Pitel, 2007) ou anglaises. Le tableau 5.1 reprend les informations sur le dimensionnement des trois ressources (base de connaissances MEDIA, FrameNet français et FrameNet). A titre de comparaison, la frame REQUEST, commune aux trois ressources, est présentée avec ses FE selon les différentes versions.

	Base MEDIA	FrameNet français	FrameNet
Frames	21	138	963
FE	86	1371	6800
Frame	REQUEST	REQUEST	REQUEST
FE	*agent* *recipient* *theme*	*speaker* *medium* *addresse* *manner* *message* *means* *topic*	*speaker* *medium* *addresse* *manner* *message* *means* *topic* *beneficiary* *time*

TABLE 5.1 – *Comparaison des 3 bases de connaissances : base* MEDIA *- FrameNet français - FrameNet. Exemple de la frame* REQUEST *et de ses éléments dans chacune des bases.*

La principale différence structurelle entre la base de connaissances MEDIA et les base de type FrameNet française ou anglaise réside dans sa construction hiérarchique. En effet, un FE de cette base peut prendre pour valeur une frame, permettant une représentation par arbres sémantiques des propositions du locuteur. Cette caractéristique n'est pas proposée pour les objets sémantiques de type strictement FrameNet pour lesquels les liens hiérarchiques sont établis entre frames et non *via* les FE. Un exemple d'arbre sémantique est présenté dans la figure 5.1.

Bien que la définition de frames dédiées au domaine du corpus MEDIA ait été privilégiée, une attention particulière a été apportée afin de maintenir la définition des

frames la plus indépendante possible de l'application. Ainsi certaines frames décrivant des connaissances générales comme les relations spatiales sont conformes à leur version FrameNet, quand d'autres plus spécifiques à l'application ont été créées ou largement adaptées. Toutefois la plupart des frames définies restent en correspondance avec des frames présentes dans le projet FrameNet. La figure 5.2 montre un extrait de ces correspondances.

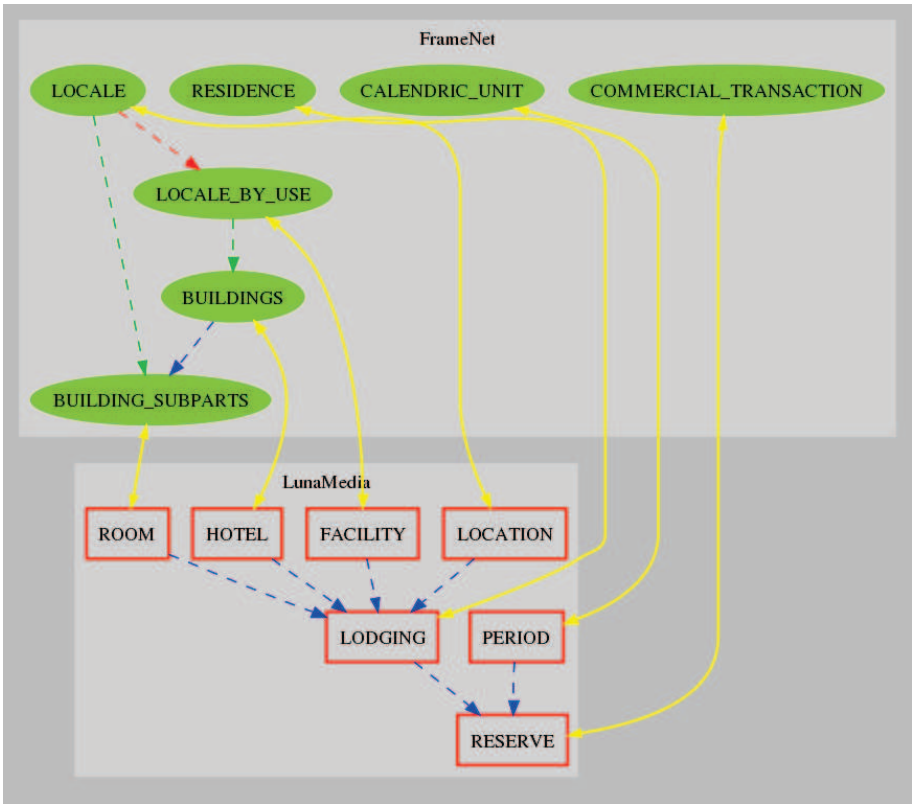


FIGURE 5.2 – *Exemples de correspondances entre les frames de la base de connaissances associée au corpus* MEDIA *et les frames FrameNet*

5.3 Base de connaissances

La construction des frames et FE a été réalisée dans un esprit de généricité mais en privilégiant l'aptitude des objets à la composition. Ainsi, dans le cas des frames abstraites, telle `REQUEST` présentée en 5.2, les FE sont de deux types. Ceux relatifs aux acteurs - locuteur, destinataire du message - sont de type `PERSON` et sont donc reliés à des instance de frames `PERSON`. Le FE relatif au thème évoqué par la frame est de type général `THING`, ce qui l'autorise à être relié à toute frame du domaine. Cet exemple est illustré par la figure 5.3

La base de connaissances complète est présentée graphiquement en annexe A.

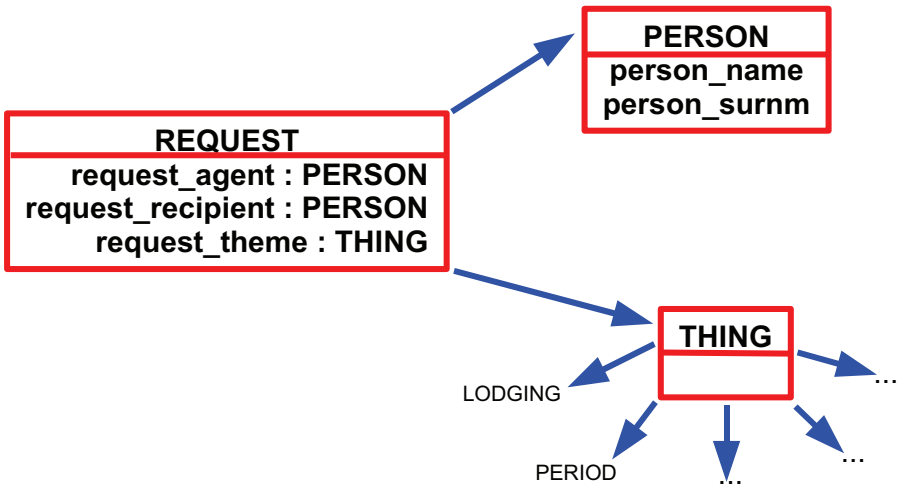


FIGURE 5.3 – *La frame abstraite* `REQUEST` *dans son contexte relationel*

Les frames et FE de la base de connaissances MEDIA sont définis par des modèles composés de LU et de concepts de base (*unités conceptuelles*, CU). Ces CU sont issus du dictionnaire sémantique de concepts MEDIA dès lors qu'ils peuvent être associés à une frame ou à un FE. Dans le cas contraire, des CU adaptés ont été définis. L'extraction automatique des composants (LU, CU) présents dans le corpus permet de s'assurer de la couverture complète du domaine par les modèles. Ces composants sont rassemblés dans un fichier XML de modèles. La base de connaissances MEDIA contient actuellement 106 CU et plus de 1000 LU.

Le tableau 5.2 présente un extrait du modèle définissant de la frame `LOCATION` et deux de ses FE, `location_town` et `location_region`.

```
<frame frname="LOCATION">
    <concept value="localisation" />
    <lexical_units value="lieu, endroit" />
    <framelement fename="location_town">
      <concept value="localisation_ville" />
      <generic_lexical_units value="ville,cité" />
      <specific_lexical_units value="paris,marseille,lyon..." />
    </framelement>
    <framelement fename="location_region">
      <concept value="localisation_region" />
      <generic_lexical_units value="région" />
      <specific_lexical_units value="auvergne,lorraine,guyane..." />
    </framelement> ...
</frame>
```

TABLE 5.2 – *Extrait de la définition de la frame* MEDIA `LOCATION` *et de ses FE* `location_town` *et* `location_region`.

Le processus de construction de la base de connaissances de frames dédiée à MEDIA ayant pour objectif d'assurer une couverture complète du domaine, les annotations manuelles ont été réalisées au fil de cette construction. Cette démarche, inspirée du projet FrameNet, a permis la création de nouvelles frames en fonction des besoins de représentation sémantique rencontrés.

La phase d'annotation manuelle est décrite dans le paragraphe 5.4 ci-après.

5.4 Annotations manuelles

L'annotation de 255 messages [1], issus de dialogues MEDIA aléatoirement sélectionnés hors de l'ensemble des dialogues de test, a été réalisée manuellement, en paralléle avec la finalisation de la base de connaissances de frames. Les besoins sémantiques pointés par la tâche d'annotation ont ainsi été comblés par la création de frames, de FE et de liens entre frames et FE dans la base de connaissances.

Un tour de parole est analysé et annoté à l'aide des objets sémantiques de la base de connaissances. L'annotateur évalue ensuite la couverture de l'annotation sémantique produite. Lorsque la représentation sémantique du message est complète, le tour de parole annoté est intégré à l'ensemble des tours de référence. Dans ce cas, le temps moyen d'annotation d'un tour de parole est de 5 minutes environ. Dans le cas contraire, les frames et FE manquants sont définis, leurs modèles complétés (LU, CU) et intégrés à la base de connaissances. Les relations sémantiques entre les nouveaux objets et ceux pré-existants sont également définies. La base de connaissances étant mise à jour, le tour de parole partiellement annoté est réintégré à l'ensemble des tours à annoter. Cette approche est illustrée par le schéma 5.4.

Après traitement complet de l'ensemble des tours de parole à annoter manuellement, la base de connaissances a été figée.

Un outil de visualisation des frames, dédié à la manipulation de dialogues, a apporté une aide précieuse à l'annotation manuelle, la vérification et la correction des annotations. Cet outil fournit pour chaque tour de parole une vue d'ensemble des frames et FE instanciés ainsi que des relations entre ces objets. Il permet un accès direct à un tour de parole sélectionné et diffuse le fichier audio correspondant. La figure 5.5 donne un exemple de visualisation des frames et FE associées à un tour de parole du locuteur.

5.5 Version LUNA

Dans le cadre du projet européen LUNA supportant ce travail, une version étendue de la base de connaissances décrite en 5.3 a été développée en 2009. Cette version comporte 54 frames et 186 FE. Un ensemble de test de plus de 3000 messages annotés et

1. Les 225 messages ont été choisis dans l'ensemble d'entraînement car cet ensemble devait initialement être intégralement annoté en frames. Cet objectif a été abandonné par la suite.

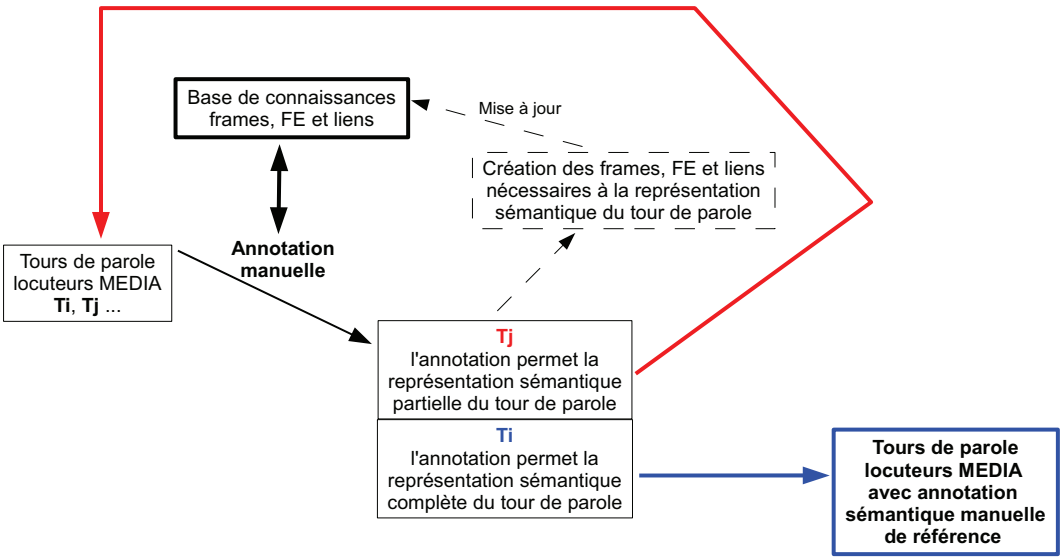


FIGURE 5.4 – *Annotation manuelle et finalisation de la base de connaissances de frames* MEDIA

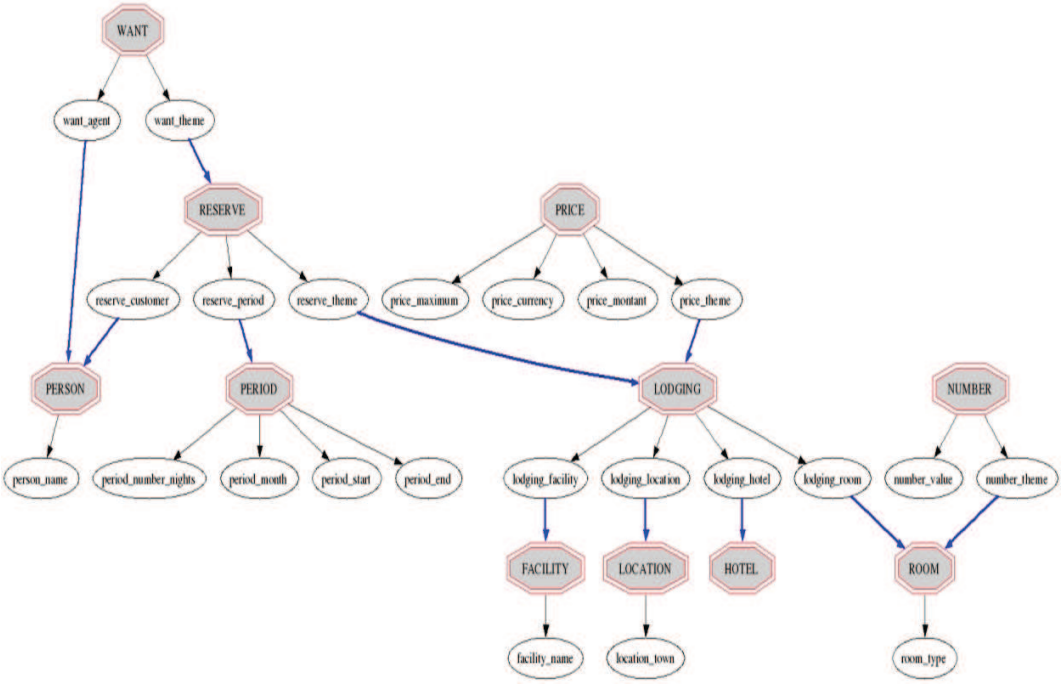


FIGURE 5.5 – *Visualisation des frames et FE du tour de parole locuteur "alors j'aurais souhaité réserver euh deux chambres individuelles euh dans un hôtel à Orange pour vingt nuits du douze juillet au trente et un juillet euh je souhaiterais que le prix soit inférieur à cent euros ou alors un peu plus s'il y a une piscine euh je souhaiterais que dans un cadre très calme et avec une piscine donc si possible"*

manuellement vérifiés est disponible. La partie du corpus MEDIA dédiée à l'entraînement, annotée automatiquement en frames et FE, est également accessible.

Faute de temps, cette nouvelle version de l'ontologie n'a pas pu être prise en compte dans ce document. Toutefois cette ressource sera exploitée rapidement. Elle nous permettra une première évaluation de la capacité de nos modèles à s'adapter à la croissance de la base de connaissances utilisée.

5.6 Conclusion

La représentation sémantique choisie dans le cadre de ce travail est issue du formalisme FrameNet (Fillmore et al., 2003). Ce choix est motivé par l'aptitude des objets sémantiques à la composition et à la représentation de dialogues de négociation. L'utilisation des ressources françaises existantes (Pado et Pitel, 2007), obtenues par projection automatique d'une partie du FrameNet américain, a été écartée au profit de la définition manuelle d'une base de connaissances dédiée au domaine MEDIA.

Cette base de connaissances est conçue pour s'adapter au domaine tout en conservant une généricité maximale. Sa définition prend en compte les particularités syntaxiques et sémantiques des transcriptions de messages oraux, permettant ainsi d'obtenir une annotation précise du corpus.

Une phase d'annotation manuelle de 225 messages locuteurs a été réalisée en interaction avec le processus de construction de la base de connaissances sémantique pour optimiser la couverture de la base et la qualité des annotations manuelles ultérieures.

Dans la perspective de produire les données d'apprentissage nécessaires aux modèles stochastiques explorés dans ce travail, un système à base de règles a été développé. Ce système, décrit au chapitre 6 exploite les connaissances issues de la tâche de définition du modèle de représentation sémantique et d'annotation manuelle présentée dans ce chapitre.

Chapitre 6

Annotation déterministe : un système à base de règles en deux étapes

Sommaire

Résumé

Ce chapitre présente le système d'annotation déterministe utilisé pour produire de manière semi-automatique les annotations de référence sur les données du corpus MEDIA. *Il comporte 2 étapes distinctes : une instanciation des frames et frame-éléments par dètection de motifs, puis une inférence à base de règles logiques des composants manquants et des relations entre les frames.*

6.1 Introduction

L'application de modèles stochastiques à la composition sémantique impose l'utilisation de données d'apprentissage pour construire les tables de probabilités conditionnelles supportant ces modèles.

L'annotation manuelle en frames de l'intégralité du corpus n'était pas envisageable, tant pour des raisons de coûts que de délai de disponibilité. Un processus d'annotation en deux étapes à base de règles a donc été développé pour produire les annotations en frames des données d'apprentissage.

La première étape du processus, décrite en 6.2, utilise les modèles définissant les frames pour déclencher l'instanciation de frames et de leurs FE selon que LU et CU sont rencontrés dans les données à annoter. La seconde étape, décrite en 6.3, compose les frames et FE proposés durant l'étape précédente grâce à l'application d'une série de règles logiques. Ce processus est progressivement enrichi pour améliorer ses performances.

6.2 Reconnaissance de modèles

Les modèles définissant les frames et leurs FE, présentés en 5.3, sont composés d'unités conceptuelles (CU) et lexicales (LU). La présence de ces CU et/ou LU dans les données à annoter déclenche l'instanciation des frames et FE auxquels ils sont associés.

Aux paires concept-valeur annotées dans le corpus MEDIA peuvent être attachés un mode (affirmatif, négatif, interrogatif ou optionnel) et un spécifieur (définissant les relations entre concepts). Ces informations ne sont pas reprises dans la définition des frames et FE pour préserver leur généricité. Seuls les unités lexicales composant le message et les concepts de base annotés dans le corpus MEDIA servent de support à la définition des objets sémantiques frames et FE.

L'algorithme de *pattern-matching* développé pour décider l'instanciation des objets sémantiques intègre plusieurs options :

- la prise en compte des LU peut être liée ou non à la présence des CU associés à l'objet sémantique à instancier ;
- les FE instanciés peuvent ou non être automatiquement reliés aux frames mères candidates ;
- un segment lexical associé à un CU peut être autorisé ou non à déclencher l'instanciation de plusieurs objets sémantiques.

Les frames et FE produits lors de cette phase d'instanciation peuvent être vus comme des fragments isolés de représentation sémantique du message du locuteur. Les seuls liens relationnels établis entre les frames et FE sont les liens d'appartenance d'un FE à une frame. La majorité des frames et FE instanciés lors de cette étape est composée d'objets sémantiques concrets, déclenchés par la présence de LU ou de CU identifiés

dans leurs modèles. Certaines frames abstraites de haut niveau, essentiellement sous-tendues par la présence d'autres frames et/ou FE, sont rarement instanciées lors de cette étape.

Le message *"réserver un hôtel"*, fréquemment rencontré dans le corpus MEDIA, est ainsi annoté lors de cette première phase à l'aide des deux seules frames `RESERVE` et `HOTEL`, non reliées entre elles (figure 6.1).

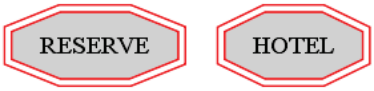


FIGURE 6.1 – *Annotation initiale par reconnaissance de modèles du message "réserver un hôtel"*

La frame `LODGING`, définissant la notion globale d'hébergement, va permettre de lier ces frames pour obtenir une représentation sémantique consistante du message sous la forme donnée figure 6.2.

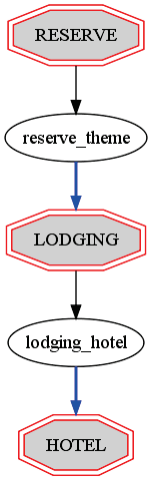


FIGURE 6.2 – *Annotation complète du message "réserver un hôtel"*

Cette représentation par arbre sémantique est obtenue lors de la deuxième étape du processus, grâce à l'application de règles d'inférences. Cette étape est décrite dans la section 6.3.

6.3 Règles d'inférences

La seconde étape soumet les frames et FE instanciés au cours de l'étape précédente à l'application d'une série de règles logiques. Frames et FE déterminent les valeurs de vérités des prédicats de ces règles. Selon ces valeurs de vérité, des frames et des FE peuvent être créés, supprimés, modifiés ou reliés.

La création de frames et FE concerne essentiellement les frames abstraites comme illustré par l'exemple de la frame `LODGING` présenté en 6.2. La suppression et la modification de frames et FE sont motivées par la présence d'objets redondants, instanciés par la présence de redondances dans de nombreux messages oraux du type *"je veux un hotel... euh... un hotel proche de la mer."*.

Enfin, la dernière fonction de cette étape d'annotation est la création de liens entre les frames et les FE. La création de ces liens est justifiée par la hiérarchie présente au sein des frames, traduite par l'aptitude de certains FE à prendre des frames pour valeurs. Les fragments sémantiques obtenus à l'issue de l'étape de reconnaissance des modèles sont composés. Les liens instanciés par les règles logiques permettent d'obtenir un arbre sémantique représentant le sens du message du locuteur.

Le langage de programmation logique `Prolog` (Colmerauer et Roussel, 1996), basé sur le calculs des prédicats du premier ordre, a été utilisé pour réaliser toutes les inférences logiques. Un programme `Prolog` se compose d'une base de faits et de règles logiques décrivant les relations entre des faits potentiels. Cette base rassemble les *connaissances* du programme. Les faits sont représentés par des prédicats affirmatifs et les règles logiques s'expriment simplement sous la forme `conclusion SI condition`.

L'exécution d'un programme `Prolog` consiste à soumettre une requête à l'interpréteur. `Prolog` cherche à prouver que la requête est vraie en analysant chaque règle et prouvant qu'elle est vérifiée.

L'implémentation des règles de composition sémantique utilisées dans cette étape d'annotation est réalisée sous `SWI-Prolog` (Wielemaker, 2003).

Une des règles `Prolog` s'écrit sous la forme :

```
do_link(LODH,H) :-
  is_fe(lodging_hotel,LODH),
  is_concept_of(hotel,LODH),
  is_fr(HOTEL,H).
```

où le symbole `:-` signifie "`SI`" [1].

Dans cet exemple, la règle crée un lien entre le FE `lodging_hotel` et la frame `HOTEL` sous la condition que le FE `lodging_hotel` ait été déclenché par la présence du concept `hotel` associé au message. Le FE `lodging_hotel` prend la frame `HOTEL` pour valeur, construisant ainsi une branche de l'arbre sémantique représentant le message.

Environ 100 règles sont actuellement appliquées. Agissant sur les frames et les FE, elle peuvent prendre en compte la présence de mots et de concepts. L'ordre dans lequel les frames et les FE ont été instanciés avant d'être soumis au programme Prolog n'influence pas les inférences réalisées.

L'inférence logique est appliquée itérativement, chaque sortie fournissant les faits soumis à l'étape de résolution suivante. L'itération peut être, au choix, poursuivie jusqu'à

1. Expression du modus-ponens.

ce qu'aucune modification ne soit plus inferée ou un nombre pré-défini de fois.

6.4 Évaluation

Pour évaluer les performances des système de composition sémantique développés dans ce travail, la préparation d'un ensemble de données de test a été nécessaire. Les 3005 tours de parole utilisateur du lot de test MEDIA, manuellement transcrits et annotés en concepts de base, ont été automatiquement annotés en frames et FE par le système à base de règles puis corrigés manuellement par un linguiste expert.

Etant données l'ampleur de la tâche et la disponibilité d'un unique linguiste expert, il n'a pas été possible d'obtenir d'IAg sur les annotations en frames et FE. Les 3005 tours de parole utilisateur manuellement transcrits, annotés en concepts de base et en frames et FE composent l'ensemble de test nommé "REF". Les annotations produites par le système d'annotation en deux étapes à base de règles sur les 3005 tours de parole du lot de test ont été évaluées par comparaison à cet ensemble "REF".

Les performances de ce système ont été mesurées en termes de précision, rappel et F-mesure. La précision est le nombre de frames, FE ou liens corrects proposés par le système rapporté au nombre total de frame, FE ou liens proposés par le système. Le rappel est le nombre de frames, FE ou liens corrects proposés par le système divisé par le nombre total de frames, FE ou liens contenus dans l'annotation de référence. La F-mesure est la moyenne harmonique standard de la précision et du rappel.

Le tableau 6.1 présente les résultats obtenus par le système d'annotation à base de règles sur les 3005 tours de parole de référence.

Les différents niveaux d'évaluation sont :

- **Frames** : les hypothèses de frames sont considérées comme correctes dès lors que les frames correspondantes sont présentes dans la référence (sans prise en compte des FE qui les composent).
- **FE** : les hypothèses de FE sont considérées comme correctes dès lors que les FE correspondants sont présents dans la référence.
- **FE{Frames}** : seules les hypothèses de FE appartenant à des hypothèses de frames correctes sont examinées. L'ensemble de référence est restreint aux FE appartenant aux frames correspondantes dans la référence.
- **Liens** : les hypothèses de liens sont considérées comme correctes dès lors que les liens correspondants sont présents dans la référence.
- **Liens{Frames}** : seules les hypothèses de liens reliant des hypothèses de frames et FE correctes sont examinées. L'ensemble de référence est restreint aux liens reliant les frames et FE correspondants dans la référence.

Les résultats obtenus par le système d'annotation à base de règles, avec des F-mesures toutes supérieures à 90%, confirme sa fiabilité et sa capacité à produire sur l'ensemble du corpus MEDIA des données d'apprentissage consistantes.

Le nombre total de frames, FE et liens présents sur les tours de parole de l'ensemble REF ansi que sur l'ensemble de test MEDIA et les dialogues d'entraînement (train) annotés grâce au système à base de règles sont indiqués dans le tableau 6.2.

		Frames	FE	FE{Frames}	Liens	Liens{Frames}
Système	p	0.98	0.97	1.00	0.95	1.00
à base de règles	r	0.99	0.94	0.95	0.86	0.88
	F-m	**0.98**	**0.96**	**0.97**	**0.90**	**0.94**
	$\bar{p}$	0.99	0.99	1.00	0.99	1.00
	$\bar{r}$	0.99	0.97	0.97	0.94	0.95
	$\overline{F\text{-}m}$	**0.99**	**0.97**	**0.98**	**0.95**	**0.96**

TABLE 6.1 – *Précision ($\bar{p}$), Rappel ($\bar{r}$) et F-mesure ($\bar{F}$-m), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) obtenus par le système d'annotation à base de règles sur les 3005 tours de parole de l'ensemble de test* MEDIA.

Ensemble	Annotation	Frames	FE	Liens
TRAIN MEDIA	à base de règles	33923	35101	15828
	nb moyen par tour de parole	2,83	2,93	1,32
TEST MEDIA	à base de règles	8315	8680	3845
	nb moyen par tour de parole	2,77	2,89	1,28
TEST MEDIA	manuelle	8241	9020	4251
	nb moyen par tour de parole	2,74	3,00	1,41

TABLE 6.2 – *Nombre de frames, FE et liens présents dans les ensembles d'apprentissage et de test* MEDIA*, annotés grâce au système à base de règles et après correction manuelle.*

On remarque que les arbres associés aux tours de parole sont d'ordre peu élevé avec une moyenne de moins de 3 frames et 3 FE par tour. Leur taille est également restreinte avec moins de 1,5 lien par tour de parole. La comparaison entre les résultats obtenus sur le test MEDIA annoté manuellement et ceux obtenus sur la version annotée par le système à base de règles indique une tendance du système déterministe à insérer des frames et à omettre des FE et des liens.

6.5 Conclusion

La mise en place de modèles stochastiques dans un système applicatif nécessite l'emploi des tables de probabilités conditionnelles qui leur sont associées. Les valeurs numériques rassemblées dans ces tables de probabilités conditionnelles doivent donc être apprises sur des ensembles de données comportant les informations que l'on souhaite étudier.

Le corpus MEDIA n'étant pas annoté en frames et FE, un système à base de règles en deux étapes a été développé pour permettre l'annotation des données d'apprentissage.

Ce système crée tout d'abord frames et FE par reconnaissance de modèles puis associe ces objets sémantiques lors d'une étape d'inférence logique.

Évalué sur les données de test du corpus MEDIA, les données annotées automatiquement s'avèrent suffisamment fiables pour être utilisées comme données d'apprentissage par les systèmes stochastiques.

Génération des fragments sémantiques

Chapitre 7

Réseaux bayésiens dynamiques : formalismes, caractéristiques, exemples

Sommaire

Résumé

Ce chapitre propose une présentation théorique des réseaux bayésiens dynamiques. Après une brève description des modèles espace-état, quelques rappels théoriques sur les modèles graphiques sont introduits. Ils permettent de définir les réseaux bayésiens dynamiques et de mettre en évidence leur aptitude à modéliser un processus temporel comme la compréhension de la parole.

7.1 Introduction

Les modèles stochastiques que nous avons conçus pour la génération de fragments de structures sémantiques sont à base de réseaux bayésiens dynamiques (DBN). Les bases théoriques définissant les DBN sont rappelées dans ce chapitre.

Comme la plupart des situations liées à l'activité humaine, le processus de compréhension ne peut être étudié sans considérer sa dimension temporelle. La compréhension d'un message impose l'utilisation d'*observations* multiples au cours de sa formulation. L'analyse de ces observations séquentielles a pour but de capturer le sens du segment de message observé en utilisant les connaissances issues des précédents segments. Le sens et les variables sémantiques utilisées pour le représenter n'étant pas observés, la modélisation du processus par des modèles espace-état est adéquate.

Les modèles espace-état intègrent la distinction entre les variables observées (par exemple les mots...) et les variables cachées (par exemple les frames sémantiques associées au fragment de message). Ils permettent de représenter les relations de dépendances et de causalités dans un cadre temporel (Dean et Kanazawa, 1989).

Les variables du modèle étant fixées, un modèle espace-état est complètement défini par la donnée de la distribution de probabilités :

- de l'état initial du système étudié,
- des transitions entre états,
- des observations sur les états cachés.

Les modèles graphiques sont des outils de représentation efficaces des modèles espace-état. Alliant théorie des probabilités et théorie des graphes, les modèles graphiques permettent de représenter de façon factorisée des distributions jointes de probabilités sur un ensemble de variables aléatoires. Une étude exhaustive des capacités de modélisation, d'inférence et d'apprentissage de ces modèles est proposée dans (Jordan, 1998).

Un des atouts des modèles graphiques est leur modularité : un modèle complexe peut être construit en associant des modèles simples. Les graphes sont des objets de représentation intuitifs des variables, de leurs dépendances et des dépendances entre états. La consistance du modèle est assurée par la théorie des probabilités.

Dans le cadre de ce travail, les modèles graphiques étudiés sont des modèles dirigés acycliques, nommés *Réseaux bayésiens* (Pearl, 1986, 1998). Ces réseaux utilisent conjointement connaissances d'experts et observations dans le cadre inférentiel bayésien. Le graphe dirigé acyclique modélisant la structure du réseau est déterminé par l'expert qui fixe les relations de dépendance entre les variables du modèle. La distribution de probabilités jointe sur l'ensemble des variables du modèle est ensuite définie en utilisant les propriété de décomposition propres aux modèles graphiques rappelées dans la section 7.2 suivante.

7.2 Modèles graphiques orientés

Les notions rappelées dans ce paragraphe sont développées dans de nombreux travaux de référence tels (Pearl, 1998; Charniak, 1991).

Soit $X_1, ..., X_n$, variables aléatoires discrètes définies par leur loi jointe P. Ces variables sont représentées par les nœuds $v_i \in V$ du graphe orienté $G(V, E)$ associé au réseau bayésien. La figure 7.1 présente un exemple de graphe de réseau bayésien.

Les arcs $e_i \in E$ du graphe associé au réseau bayésien représentent les dépendances entre les variables. On dit que $u \in V$ est un parent de $v \in V$ si $(u, v) \in E$. L'ensemble des nœuds parents d'un nœud v est noté $pa(v)$. Les fils de v sont les nœuds dont v est un parent. Les descendants de v sont les nœuds fils de fils et leurs descendants.

La structure graphique d'un réseau bayésien satisfait le critère de *d-séparation* : toute variable est indépendante de tout sous-ensemble de ses non-descendants, conditionnellement à ses parents.

En toute généralité, la formule des probabilités composées permet d'écrire :

$$P(X_1, ..., X_n) = P(X_n|X_{n-1}, X_{n-2}, ..., X_2, X_1)...P(X_2|X_1)P(X_1) \quad (7.1)$$

$$= P(X_1)\prod_{i=n}^{2} P(X_i|X_{i-1}, ..., X_1) \quad (7.2)$$

$pa(X_i)$ étant l'ensemble des variables qui conditionnent X_i, la d-séparation permet d'écrire :

$$P(X_i|X_{i-1}, ..., X_1) = P(X_i|pa(X_i) \quad (7.3)$$

D'après (7.2) et (7.3) la loi jointe s'écrit alors sous la forme :

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i|pa(X_i) \quad (7.4)$$

qui définit complètement le réseau bayésien.

La probabilité d'une variable X_i n'est plus calculée conditionnellement à toutes les réalisations de ses prédécesseurs $X_{i-1}, ..., X_1$ mais seulement à celles de ses parents, éléments de $pa(X_i)$.

Si le graphe du réseau est complet, soit au pire cas, la factorisation (7.4) est équivalente à la factorisation générale (7.2) à l'ordre des variables près. Dans tous les autres cas, d'autant plus favorables que la densité du graphe est faible, le calcul de la loi jointe est très simplifiée par l'utilisation de la factorisation (7.4).

Considérant l'exemple de réseau bayésien illustré par la figure 7.1, on a $pa(X_1) = pa(X_2) = \emptyset$, $pa(X_3) = \{X_1, X_2\}$, $pa(X_4) = pa(X_5) = \{X_3\}$ et $pa(X_6) = \{X_4, X_5\}$.

La loi jointe définissant le réseau permet donc d'écrire :

$$P(X_1, X_2, X_3, X_4, X_5, X_6) = P(X_1)P(X_2)P(X_3|X_1, X_2)P(X_4|X_3)P(X_5|X_3)P(X_6|X_4, X_5)$$

grâce à la factorisation (7.4).

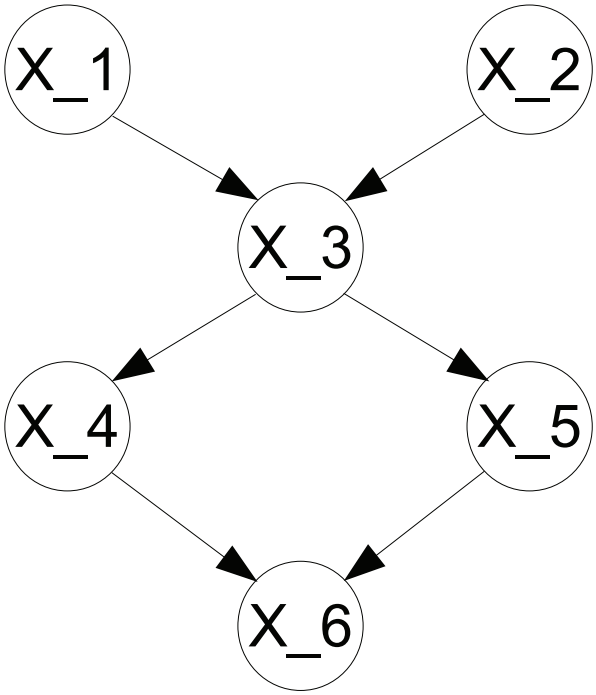


FIGURE 7.1 – *Graphe d'un réseau bayésien*

Les réseaux bayésiens décrits dans ce paragraphe ne peuvent appréhender que des ensembles finis de variables aléatoires. Ils ne sont donc pas aptes à modéliser le processus de la compréhension orale dans sa dimension temporelle et séquentielle. De nouveaux types de réseaux ont été proposés pour représenter les processus temporels en conservant les atouts des réseaux bayésiens. Il s'agit des *réseaux bayésiens dynamiques* étudiés dans le paragraphe suivant.

7.3 Réseaux bayésiens dynamiques

Les réseaux bayésiens dynamiques (*Dynamic Bayesian Network*, DBN) étendent la notion de réseaux bayésiens à la modélisation de distributions de probabilités sur des ensembles dénombrables de vecteurs aléatoires. Les phénomènes étant modélisés dans un *espace d'états*, ces vecteurs aléatoires représentent les variables observées, cachées et inférées du modèle. Une étude exhaustive des DBN est présentée dans (Mihajlovic et Petkovic, 2001) et (Murphy, 2002).

Seuls les processus stochastiques à temps discret sont étudiés dans ce travail. Le pas

de temps t est incrémenté à chaque nouvelle observation, le modèle proposant ainsi la représentation d'une succession d'événements discrets.

L'aspect *dynamique* d'un DBN est lié à sa faculté de modéliser l'évolution temporelle d'un phénomène. La structure du réseau bayésien choisi pour représenter le phénomène n'évolue pas avec le temps. En revanche, les distributions de probabilités associées aux états successifs de ce réseau sont dépendantes de la succession temporelle des observations. Ces dépendances sont supportées par les variables aléatoires de *transition* entre deux états successifs. Les DBN permettent la modélisation des dépendances entre les variables du réseau bayésien au temps t ainsi que celles liant deux étapes temporelles successives $t-1$ et t. Un DBN modélise donc un système dynamique.

Le schéma 7.2 décrit un exemple de modèle DBN sur trois étapes temporelles successives. Les arcs pleins reliant les nœuds matérialisent les dépendances entre les variables du réseau à chaque étape de temps tandis que les arcs pointillés indiquent les dépendances entre deux étapes de temps successives.

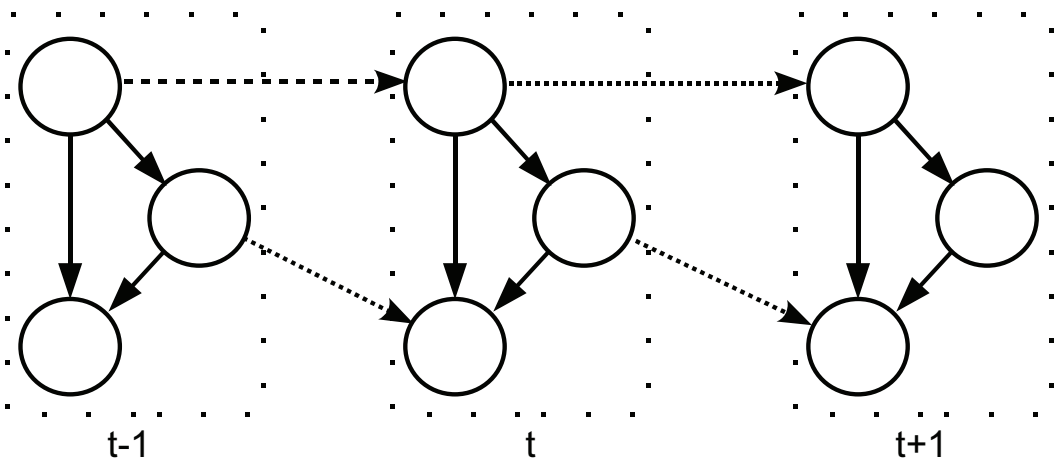


FIGURE 7.2 – *Schéma représentant l'évolution d'un processus sur trois étapes temporelles successives $t-1$, t et $t+1$.*

V_t étant le vecteur aléatoire représentant l'état du système au temps t, un DBN est défini par l'état initial du système et le réseau bayésien temporel décrivant les relations entre deux états successifs.

Formellement, un DBN est donc complètement défini par le couple $(B_1, B_{\rightarrow})$ (notation partiellement empruntée à (Murphy, 2002)) tel que :

- B_1 est le réseau bayésien définissant la distribution de probabilités initiale $P(V_1)$;
- $B_{\rightarrow}$ est le réseau bayésien associé à deux pas de temps qui définit la distribution de probabilités $P(V_t|V_{t-1})$.

Représentée par un graphe orienté acyclique, $P(V_t|V_{t-1})$ s'écrit sous la forme :

$$P(V_t|V_{t-1}) = \prod_{i=1}^{N} P(V_t^i|pa(V_t^i))$$

où V_t^i est la $i$ème composante aléatoire de V_t (associée au nœud i du graphe) et $pa(V_t^i)$ représente les composantes aléatoires dont dépend V_t^i (i.e. les nœuds parents du nœud i dans le graphe de $B_\rightarrow$).

Chaque nœud i est associé à une distribution de probabilités conditionnelle au temps t, $P(V_t^i|pa(V_t^i)$ pour tout $t > 1$.

Dans ce travail, les parents $pa(V_t^i)$ du nœud i au temps t peuvent être des nœuds du même état (temps t) ou des nœuds de l'état précédent (temps $t-1$). Tout système représenté par un DBN est donc markovien du premier ordre : l'état du système au temps t dépend uniquement de son état au temps $t-1$. Cette propriété traduit l'indépendance du futur conditionnellement au passé. Ce choix est motivé par le souhait de limiter la complexité des modèles utilisés. En effet, du point de vue théorique, rien n'interdit à un DBN de modéliser des dépendances entre des étapes de temps non successives.

Selon le principe de causalité, les arcs reliant les états successifs sont orientés dans l'ordre temporel. Les modèles considérés dans ce travail sont homogènes dans le temps, i.e. les distributions de probabilités conditionnelles sont invariantes au cours du temps.

La distribution de probabilités jointe associée à un DBN, définie par le *déroulement* du réseau bayésien $B_\rightarrow$ sur T séquences temporelles, s'écrit donc :

$$P(V_{1,\ldots,T}) = \prod_{t=1}^{T}\prod_{i=1}^{N} P(V_t^i|pa(V_t^i))$$

L'exemple le plus simple de DBN est le HMM pour lequel un état comporte une seule variable cachée et une seule variable observée à chaque étape temporelle. Un HMM est représenté graphiquement sur trois périodes temporelles par la figure 7.3.

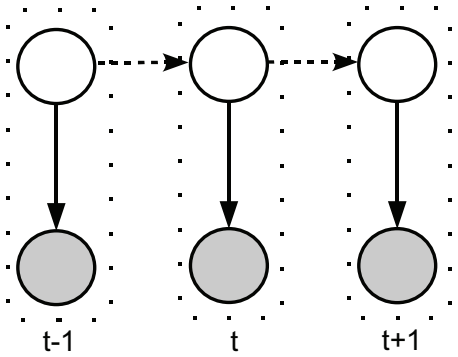


FIGURE 7.3 – *Exemple de modèle HMM représenté graphiquement sur trois périodes temporelles*

Par convention, les noeuds symbolisant des variables observées sont grisés tandis que les noeuds clairs désignent les variables cachées.

7.4 Conclusion

Ce chapitre a présenté les fondements théoriques des modèles graphiques orientés et en particulier l'aspect dynamique des DBN. Leurs caractéristiques mettent en évidence l'aptitude de ces modèles à modéliser des systèmes dynamiques complexes, tant dans leur dimension temporelle que situationnelle. Ils offrent un cadre théorique et pratique complet pour représenter une large gamme de probabilités conditionnelles mettant en jeu des variables stochastiques discrètes ou continues et réaliser des opérations complexes comme l'inférence sur celles-ci.

Ainsi, la capacité des DBN à modéliser la tâche de compréhension de messages oraux a motivé leur utilisation dans ce travail. Le chapitre suivant 8 détaille les raisons de ce choix. Il présente les modèles conçus pour générer les fragments sémantiques sur lesquels repose la compréhension et les paramètres associés à ces modèles.

Chapitre 8

Des réseaux bayésiens dynamiques pour la génération de fragments sémantiques

Sommaire

Résumé

Ce chapitre présente l'approche é base de réseaux bayésiens dynamiques utilisée dans ce travail pour la génération de fragments de structures sémantiques. L'organisation structurelle des modéles utilisés est détaillée. Les paramétres stochastiques choisis sont précisés et les méthodes dédiées é l'apprentissage é partir des données de ces paramétres sont explicitées.

8.1 Introduction

Les réseaux bayésiens dynamiques ont été présentés dans le chapitre précédent. Ce sont des modéles d'une grande flexibilité permettant de représenter des systémes stochastiques complexes. Leur adaptabilité autorise des modélisations variées qu'il est peu coûteux de faire évoluer. Les DBN, utilisés dans de nombreuses tâches de modélisation de données séquentielles, obtiennent des résultats au niveau de l'état de l'art (voir (Lefèvre, 2007) pour un exemple d'application des DBN au probléme de l'interprétation littérale de la parole).

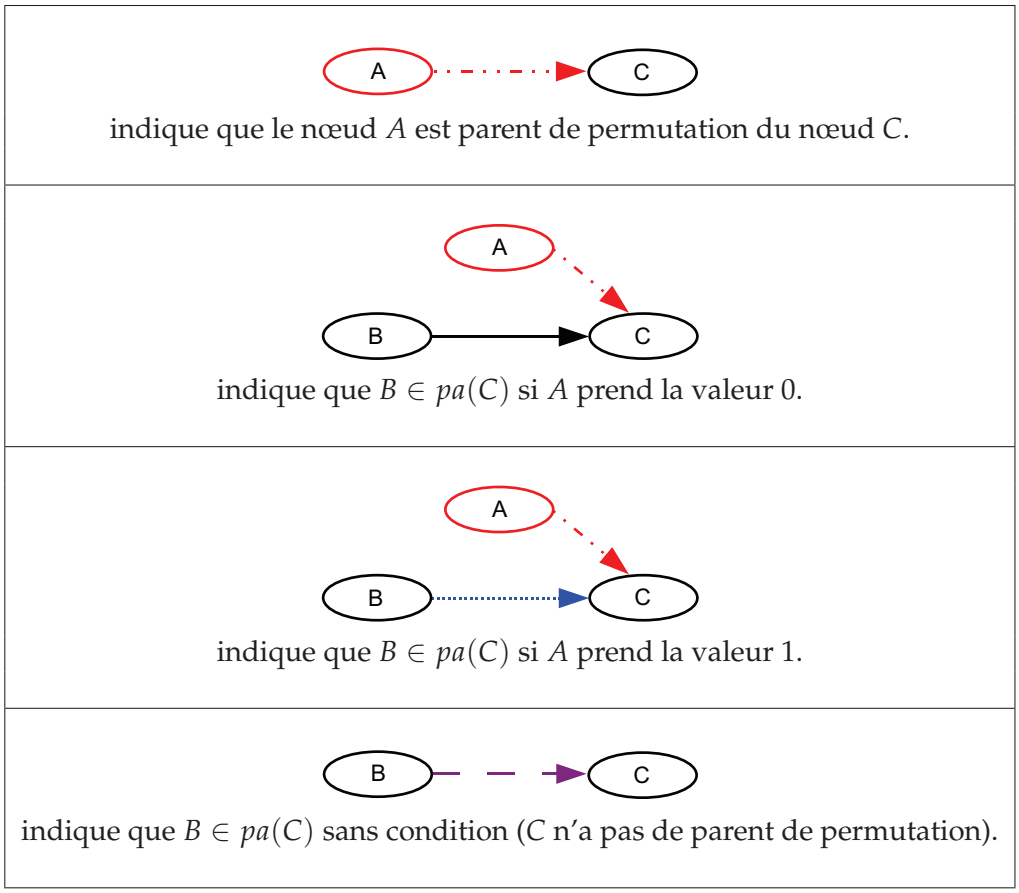
Le contexte dans lequel ce travail a été réalisé est sous-tendu par les théories de la causalité sous incertitude et des réseaux de croyance (Dean et Kanazawa, 1988, 1989). La modélisation du processus de compréhension de la parole pour laquelle nous avons opté repose sur les capacités des DBN é générer les fragments sémantiques associés aux observations des mots fournies par le module de reconnaissance de la parole et aux concepts de base fournies par le module de compréhension littérale.

Le corpus MEDIA (voir CHAP.4) étant manuellement transcrit et annoté en concepts de base, les paramétres des modéles évalués peuvent être appris directement sur les observations. Les modéles mettent en œuvre les variables aléatoires représentant les mots, les concepts de base et les objets sémantiques (frames et FE) ainsi que les transitions entre ces entités. Les objets sémantiques sont portés par des variables cachées tandis que les mots et concepts de base sont considérés comme des observations. Les relations de dépendance entre variables sont établies dans l'esprit d'une influence directe des objets sémantiques sur les mots et concepts de base observés.

Les paragraphes suivants présentent les structures et les paramétres des modéles génératifs é base de DBN utilisés pour la composition des frames dans notre systéme de compréhension. Dans les graphes proposés, les nœuds symbolisent les variables et les arcs matérialisent les dépendances conditionnelles. Par souci de lisibilité, certains nœuds et certains arcs ne sont pas représentés dans les graphes simplifiés des figures 8.1, 8.3 et 8.5. Les versions complétes des deux premiers modéles sont données dans les figures 8.2 et 8.4.

Dans les graphes simplifiés, seuls deux événements temporels (soit deux mots) sont indiqués. En pratique, ce schéma est répété autant que nécessaire au long de la séquence de mots étudiée, ce qui est indiqué par la flêche circulaire des graphes généraux. Les nœuds sont grisés lorsque les variables sont observées tandis qu'ils restent clairs pour les variables cachées. Les arcs pleins illustrent les dépendances conditionnelles entre les variables. Les arcs pointillés qui indiquent que les dépendances conditionnelles changent en fonction de la valeur d'un *parent de permutation* représenté par un nœud également en pointillés. L'utilisation de parents de permutation dans les réseaux bayésiens (nommés alors *multi-réseaux*) permet de représenter efficacement le changement de conditionnement en fonction de la valeur d'un nœud (Geiger et Heckerman, 1996; Bilmes, 2000). Les valeurs des parents de permutation sont binaires. Elles indiquent dans nos modéles l'état particulier d'un nœud en fonction du contexte.

La légende des graphes de représentation des modéles investigués est donnée ci-aprés :



Légende des graphes de représentation des modéles DBN étudiés

Toutes les variables sont observées pendant l'entraînement du modéle. Ainsi, les tables de probabilités conditionnelles associées aux arcs sont directement obtenues é partir des observations. Aucune itération EM n'est donc nécessaire.

Le calcul des probabilités de ces tables est réalisé grâce é des modéles de langage factorisés (*Factored Language Models*, FLM) utilisant des techniques de repli paralléle généralisé (*Generalized Parallel Backoff*, GPB) (Bilmes et Kirchhoff, 2003; Kirchhoff et al., 2008). Les FLM sont une extension des modéles de langage classiques dans laquelle les prédictions sont basées sur un ensemble de caractéristiques et non plus seulement sur les précédentes occurrences de la variable. Le repli paralléle généralisé permet d'étendre les procédures de repli standard au cas oé des éléments de différents types sont considérées, sans contrainte temporelle imposée : contrairement aux modéles de langage classiques, dans un FLM les éléments intervenant au moment de la prédiction peuvent être pris en compte.

La section suivante 8.2 présente un modéle compact dans lequel les fragments de frames et FE sont représentés par une seule variable composite. Le modéle factorisé qui considére des frames et FE représentés par deux variables distinctes mais simultanément décodées est décrit en 8.3. Enfin, un modéle é deux niveaux dans lequel les frames sont décodées en premier lieu, puis utilisées comme des valeurs observées lors du décodage des FE, est détaillé dans la section 8.4.

8.2 Modéle compact

La figure 8.1 décrit le modéle génératif é base de DBN dans lequel les fragments de frames et FE sont représentés par une seule variable composite **frM**. Le graphe complet du DBN est donné dans la figure 8.2.

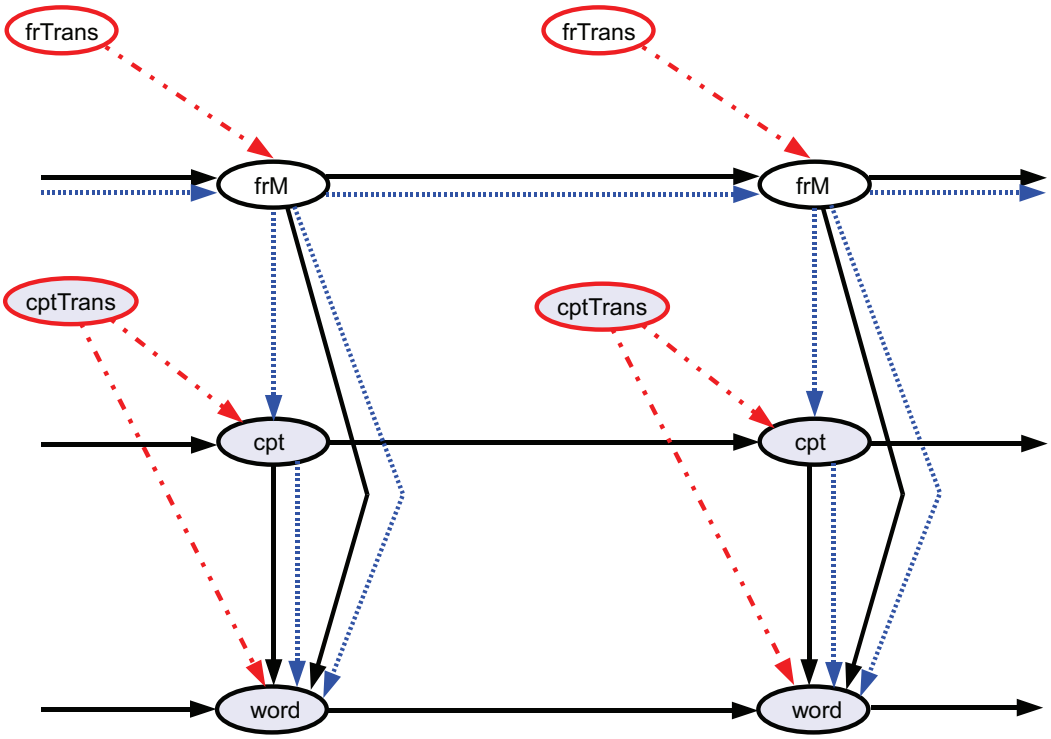


FIGURE 8.1 – *frame et FE considérés comme une seule variable non-observée.*

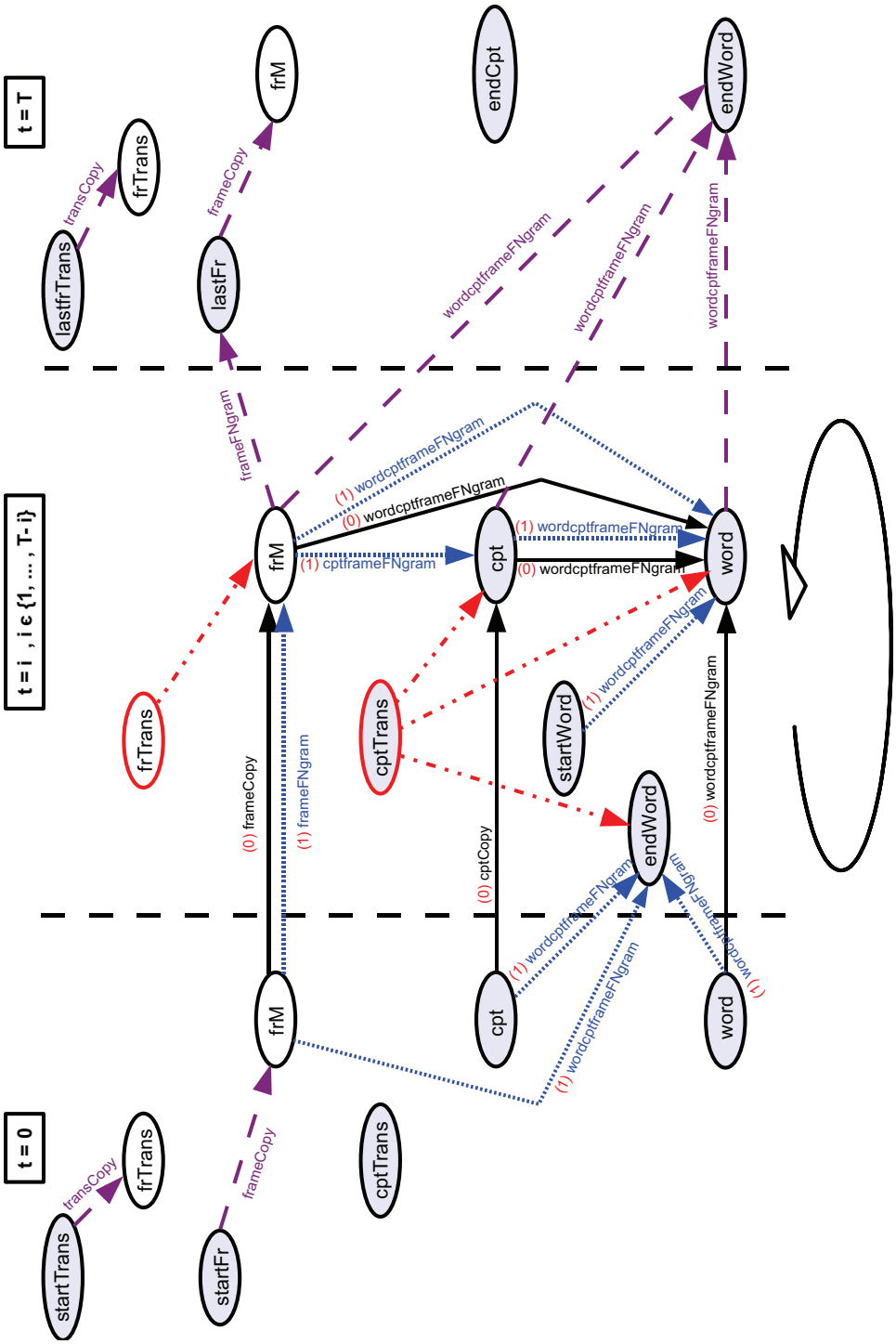


FIGURE 8.2 – *Modéle compact frame/FE*

Pour ce modéle, les variables observées sont :
- les mots : nœuds **word**,
- les concepts de base : nœuds **cpt**,
- les transition entre concepts : nœuds **cptTrans**,
- les variables “outils” :
  - nœuds **startTrans**, **startFr**, **lastfrTrans**, **lastFr** et **endCpt** qui définissent le DBN é l'origine du message (temps $t = 0$) et é la fin du message (temps $t = T$),
  - nœuds **startWord** et **endWord** qui indiquent début et fin d'une séquence de mots associée é un concept.

Les variables cachées sont :
- les fragments de frames-FE : nœuds **frM**,
- les transitions entre ces fragments : nœuds **frTrans**.

La variable transitionnelle cachée représentée par le nœud clair **frTrans** est parent de permutation du nœud **frM**. La permutation porte dans ce cas sur la distribution de probabilités utilisée pour prédire la valeur du nœud **frM** au temps t sachant l'état du modéle é t_1. Ainsi, si **frTrans** a une valeur nulle, le fragment frame/FE associée au nœud **frM** est identique é celui de son prédécesseur. La table de probabilités conditionnelles utilisée pour déterminer la valeur du fragment frame/FE é l'étape t est la matrice identité de taille nxn oé n est le cardinal de l'ensemble des fragments rencontrés dans les données d'apprentissage. Cette table est nommée *frameCopy* et mentionnée dans le graphe 8.2.

Lorsque **frTrans** est égal é un, la nouvelle valeur du fragment frame/FE est déterminée en fonction de la probabilité $P(f|f_{-1})$ du fragment f connaissant le fragment précédent f_{-1}. La table de probabilités conditionnelles utilisée est nommée *frameFNgram* dans le graphe 8.2.

La variable transitionnelle observée représentée par le nœud grisé **cptTrans** est parent de permutation des nœuds **cpt** et **word**. Cette variable prend la valeur 0 si le concept ne change pas de l'étape $t - 1$ é t, et la valeur 1 dans le cas contraire.

En l'absence de changement de concept observé, le concept associé au nœud **cpt** courant est identique é celui de son prédécesseur selon la table de probabilités *cptCopy*, identité de taille mxm oé m est le cardinal de l'ensemble des concepts rencontrés dans les données d'apprentissage.

Le mot associé au nœud **word**, observé, détermine les valeurs de probabilités associées é son conditionnement par le mot précédent, le concept courant et le fragment de frame/FE courant selon la table de probabilités conditionnelles *wordcptframeFNgram* mentionnée en 8.2

Lors du changement de concept observé, **cptTrans** prend la valeur 1 et le concept associé au nœud **cpt** courant détermine les valeurs de probabilités associées au conditionnement du concept par le fragment frame/FE. La table de probabilités conditionnelles utilisée dans ce cas, nommée *cptframeFNgram*, est mentionnée dans le graphe 8.2.

Le mot associé au nœud **word**, observé, est conditionné selon *wordcptframeFNgram* par le concept et le fragment de frame/FE courants et par le nœud **startWord** qui in-

dique l'entrée dans une nouvelle séquence de mots associée au concept courant.

Les tables de probabilités conditionnelles correspondant aux arcs du graphe 8.1 sont produites par les implémentations des FLM données ci-aprés. Les variables FFE, C et W représentent respectivement un fragment frame/FE, un concept et un mot tandis que h fixe la longueur de l'historique ($h = -1$ pour un bigramme).

– distribution conditionnelle sur les séquences de fragments de F/FE ;

$P(FFE) \simeq \prod P(ffe|ffe_h) : frameFNgram$

– distribution conditionnelle sur les séquences de concepts conditionnées par les fragments frame/FE. Le GPB est effectué dans l'ordre $\{c_h, ffe\}$;

$P(C|FFE) \simeq \prod P(c|c_h, ffe) : cptframeFNgram$

– distribution conditionnelle sur les séquences de mots conditionnées par les concepts et les fragments frame/FE. Le GPB est effectué dans l'ordre $\{w_h, c, ffe\}$.

$P(W|C, FFE) \simeq \prod P(w|w_h, c, ffe) : wordcptframeFNgram$

Le choix de représenter les fragments de frames et FE par une seule variable est essentiellement motivé par la réduction de la complexité du décodage. La cardinalité de l'ensemble des valeurs possibles de frame/FE est limitée au nombre de fragments de frame/FE observés dans les données d'apprentissage. Cependant, cette approche conduit é l'utilisation de liens déterministes entre frames et FE.

8.3 Modéle factorisé

La figure 8.3 décrit le modéle génératif é base de DBN dans lequel les fragments de frames et FE sont représentés par deux variables distinctes **frM** et **feM**, simultanément décodées. Le graphe complet du DBN est donné dans la figure 8.4.

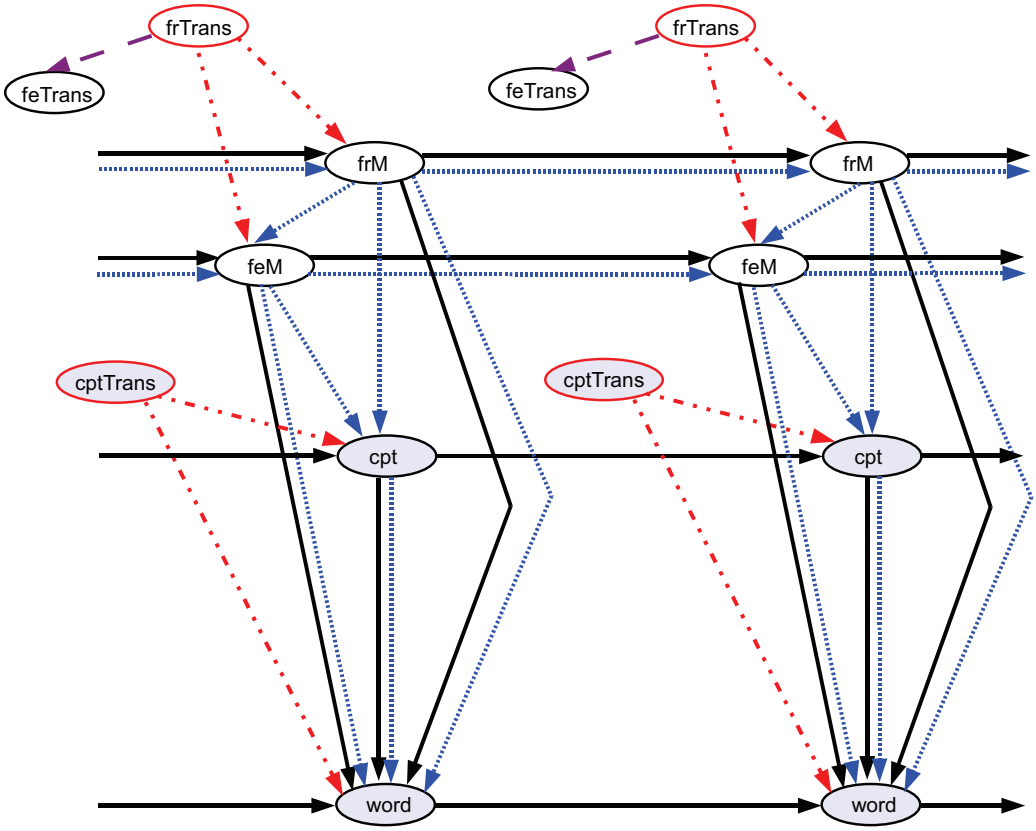


FIGURE 8.3 – *frame et FE considérés comme deux variables non-observées.*

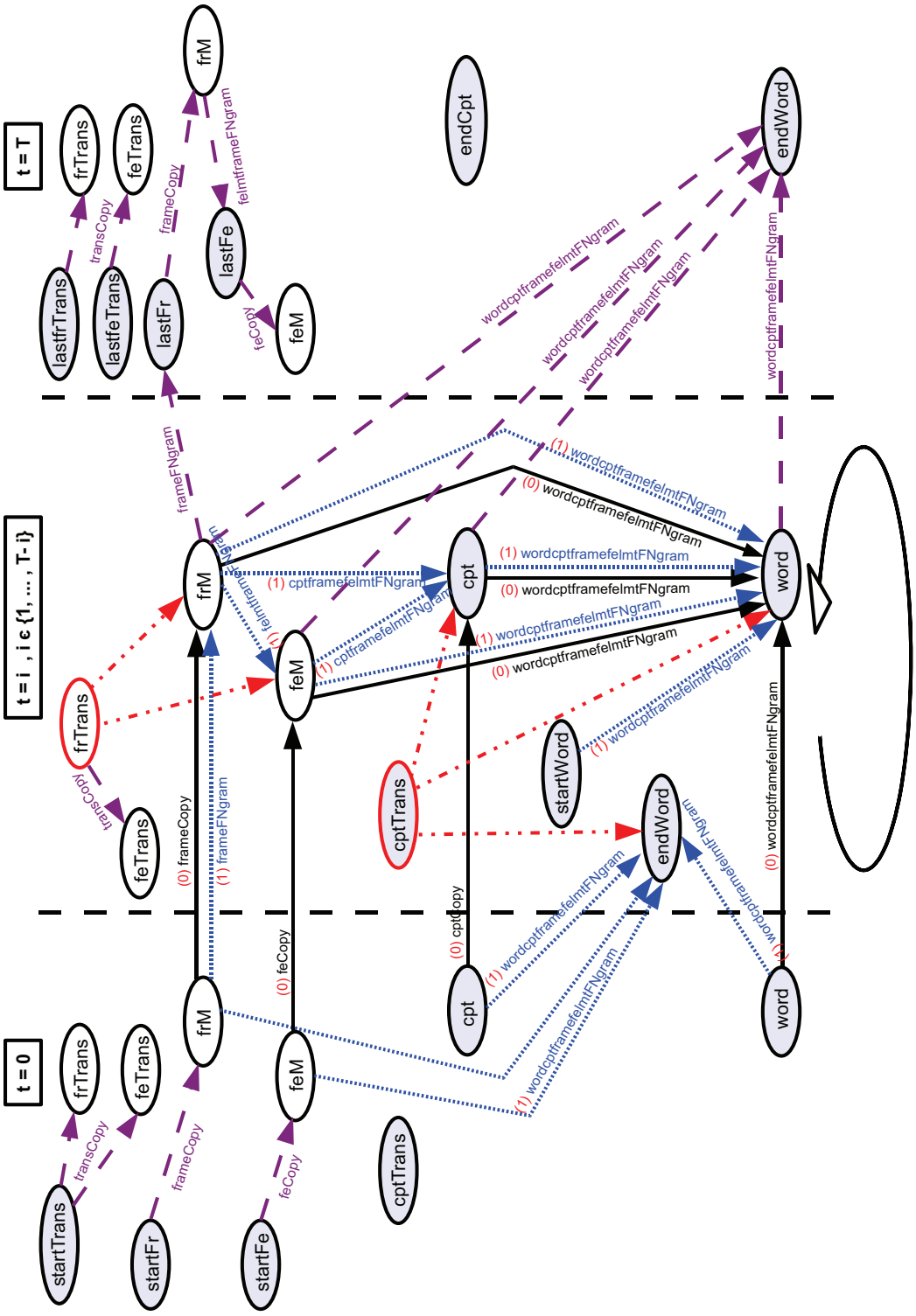


FIGURE 8.4 – *Modéle factorisé*

Pour ce modéle, les variables observées sont :
- les mots : nœuds **word**,
- les concepts de base : nœuds **cpt**,
- les transition entre concepts : nœuds **cptTrans**,
- les variables "outils" :
  - nœuds **startTrans**, **startFr**, **startFe**, **lastfrTrans**, **lastfeTrans**, **lastFr**, **lastFe** et **endCpt** qui définissent le DBN é l'origine du message (temps $t = 0$) et é la fin du message (temps $t = T$),
  - nœuds **startWord** et **endWord** qui indiquent début et fin d'une séquence de mots associée é un concept.

Les variables cachées sont :
- les fragments de frames : nœuds **frM**,
- les fragments de FE : nœuds **feM**,
- les transitions entre ces fragments : nœuds **frTrans** et **feTrans**.

Variables transitionnelles et parents de permutation se comportent comme décrit en 8.2.

Les tables de probabilités conditionnelles correspondant aux arcs du graphe 8.3 sont produites par les implémentations des FLM données ci-aprés. Les variables F, FE, C et W représentent respectivement un fragment frame, un fragment FE, un concept et un mot tandis que h fixe la longueur de l'historique ($h = -1$ pour un bigramme).

- distribution conditionnelle sur les séquences de fragments de frames ;

$$P(F) \simeq \prod P(f|f_h) : \textit{frameFNgram}$$

- distribution conditionnelle sur les séquences de fragments de FE conditionnées par les fragments de frames. Le GPB est effectué dans l'ordre $\{fe_h, f\}$;

$$P(FE|F) \simeq \prod P(fe|fe_h, f) : \textit{felmtframeFNgram}$$

- distribution conditionnelle sur les séquences de concepts conditionnées par les fragments de frames et les fragments de FE. Le GPB est effectué dans l'ordre $\{c_h, fe, f\}$;

$$P(C|FE, F) \simeq \prod P(c|c_h, fe, f) : \textit{cptframefelmtFNgram}$$

- distribution conditionnelle sur les séquences de mots conditionnées par les concepts, les fragments de frames et les fragments de FE. Le GPB est effectué dans l'ordre $\{w_h, c, fe, f\}$.

$$P(W|C, FE, F) \simeq \prod P(w|w_h, c, fe, f) : \textit{wordcptframefelmtFNgram}$$

Le modéle factorisé permet de considérer les ambiguïtés des liens entre frames et FE en leur attribuant des probabilités et testant chaque combinaison au cours du décodage. Les combinaisons non rencontrées dans les données d'apprentissage sont évaluées grâce é des techniques de repli. Dans nos modéles le GPB utilise la technique de discount absolu (ou Kneser-Ney) (Kneser et Ney, 1995). évidemment, cette approche est coûteuse et la complexité du modéle est telle qu'il est nécessaire d'utiliser un algorithme sous-optimal de recherche en largeur (de type recherche en faisceau, *beam search*) pendant le décodage.

8.4 Modéle é deux niveaux

Le modéle é deux niveaux dans lequel les frames sont décodées en premier lieu, puis utilisées comme des valeurs observées lors du décodage des FE est illustré par la figure 8.5. Le graphe développé du premier niveau de ce modéle est semblable é celui de la figure 8.2, oé la variable **frM** représente les fragments de frames. Le graphe développé du second niveau est semblable é celui de la figure 8.4 oé les variables **frM** et **frTrans** sont observées.

Pour le premier niveau :

Les variables observées sont :
– les mots : nœuds **word**,
– les concepts de base : nœuds **cpt**,
– les transition entre concepts : nœuds **cptTrans**,
Les variables cachées sont :
– les fragments de frames : nœuds **frM**,
– les transitions entre ces fragments : nœuds **frTrans**.

Pour le second niveau :

Les variables observées sont :
– les mots : nœuds **word**,
– les concepts de base : nœuds **cpt**,
– les transition entre concepts : nœuds **cptTrans**,
– les fragments de frames : nœuds **frM**,
– les transitions entre ces fragments : nœuds **frTrans**
Les variables cachées sont :
– les fragments de FE : nœuds **feM**

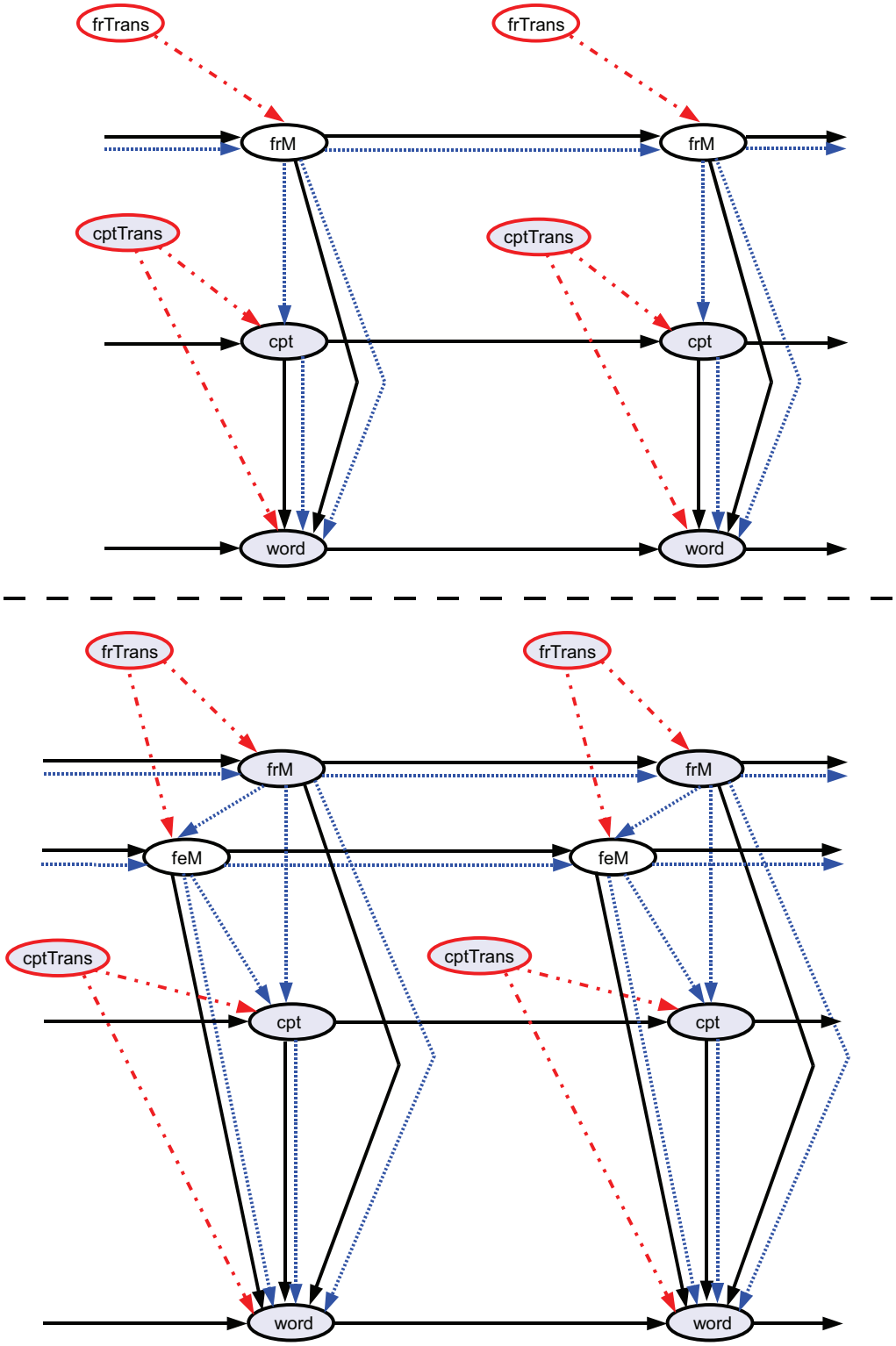


FIGURE 8.5 – *Décodage é deux niveaux des frames et FE*

Variables transitionnelles et parents de permutation se comportent comme décrit en 8.2.

Les tables de probabilités conditionnelles correspondant aux arcs du graphe 8.5 sont produites par les implémentations des FLM données ci-aprés. Les variables F, FE, C et W représentent respectivement un fragment frame, un fragment FE, un concept et un mot tandis que h fixe la longueur de l'historique ($h = -1$ pour un bigramme).

⋆ *Premier niveau :*

– distribution conditionnelle sur les séquences de fragments de frames ;

$P(F) \simeq \prod P(f|f_h)$: *frameFNgram*

– distribution conditionnelle sur les séquences de concepts conditionnées par les fragments de frames. Le GPB est effectué dans l'ordre $\{c_h, f\}$;

$P(C|F) \simeq \prod P(c|c_h, f)$: *cptframeFNgram*

– distribution conditionnelle sur les séquences de mots conditionnées par les concepts et les fragments de frames. Le GPB est effectué dans l'ordre $\{w_h, c, f\}$.

$P(W|C, F) \simeq \prod P(w|w_h, c, f)$: *wordcptframeFNgram*

⋆ *Second niveau :*

– distribution conditionnelle sur les séquences de fragments de frames observés ;

$P(\hat{F}) \simeq \prod P(\hat{f}|\hat{f}_h)$: *frameFNgram*

– distribution conditionnelle sur les séquences de fragments de FE conditionnées par les fragments de frames observés. Le GPB est effectué dans l'ordre $\{fe_h, \hat{f}\}$;

$P(FE|\hat{F}) \simeq \prod P(fe|fe_h, \hat{f})$: *felmtframeFNgram*

– distribution conditionnelle sur les séquences de concepts conditionnées par les fragments de frames observés et les fragments de FE. Le GPB est effectué dans l'ordre $\{c_h, \hat{f}, fe\}$;

$P(C|\hat{F}, FE) \simeq \prod P(c|c_h, \hat{f}, fe)$, GPB dans l'ordre $\{c_h, \hat{f}, fe\}$: *cptframefelmtFNgram*

– distribution conditionnelle sur les séquences de mots conditionnées par les concepts, les fragments de frames observés et les fragments de FE. Le GPB est effectué dans l'ordre $\{w_h, c, \hat{f}, fe\}$.

$P(W|C, \hat{F}, FE) \simeq \prod P(w|w_h, c, \hat{f}, fe)$: *wordcptframefelmtFNgram*

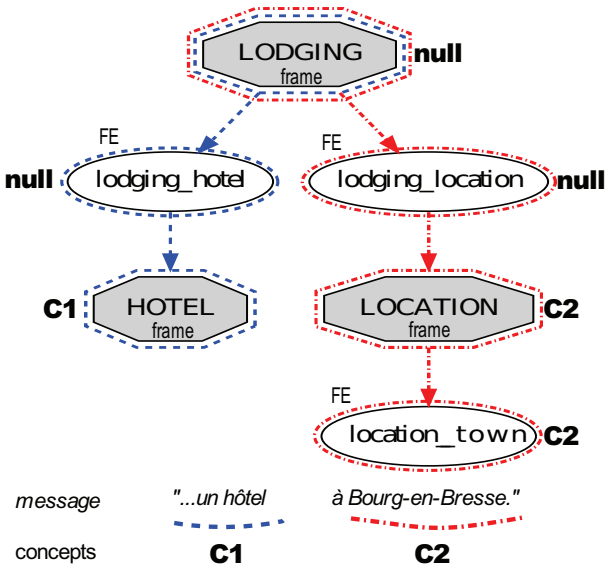


FIGURE 8.6 – *Branches projetées associées é la séquence "... un hôtel é Bourg-en-Bresse".*

oé le chapeau indique les variables ayant des valeurs fixées.

Bien qu'également sous-optimale, l'approche é deux niveaux de décodage propose un modéle de complexité moindre que celle du modéle factorisé tout en conservant l'aptitude é décoder des frames et FE reliés par des liens non déterministes.

8.5 Définition et dérivation des fragments sémantiques

La représentation des frames et FE étant hiérarchique, des situations de recouvrement peuvent se produire lors de la détermination des frames et des FE associés é un concept. Cela arrive principalement lorsque plusieurs frames ou FE ont été déclenchés par le méme concept mais également lorsque les processus d'inférence et de composition ont créé des structures de frames et FE imbriquées reliées au même concept. Pour résoudre ce probléme, un algorithme de projection d'arbre est appliqué. Il est décrit ci-dessous et son pseudo-code est donné dans l'algorithme 1.

La projection est réalisée sur l'annotation en frames et FE, structurée en arbre, de la phrase compléte. Elle permet de définir des sous-branches de l'arbre associées é un seul concept. Partant d'une feuille de l'arbre, une branche de frame/FE est obtenue en agrégeant les valeurs de nœuds péres (frame ou FE) aussi longtemps qu'ils sont associés au méme concept (ou é aucun). Les arétes des branches sont définies par les liens d'appartenance des FE aux frames et par les liens de typage des FE en frame (un FE prenant une frame pour valeur).

Par exemple, la séquence de mots *"... un hôtel é Bourg-en-Bresse"*, proposée dans la figure 8.6, entraéne la création des branches projetées
`HOTEL-lodging_hotel-LODGING` et
`location_town-LOCATION-lodging_location-LODGING`.

Les branches sont exploitées selon les différents modéles DBN. Dans le cas du modéle compact, les branches sont directement considérées comme des classes composées. Dans les modéles factorisés et é deux niveaux, frames et FE sont séparés pour produire deux ensembles de classes distinctes.

t] Algorithme de projection d'arbres

Entrée : $\{c_i\}$ séquence de concepts associés é la phrase, $\mathcal{T}$ arbre de frames et FE représentant le message
Sortie : $\mathcal{B}$ ensemble des branches

1: $\mathcal{B} \leftarrow \emptyset$
2: **pour tout** $c \in \{c_i\}$ **faire**
3: branche $b_c \leftarrow \emptyset$
Génération des branches principales
4: **pour tout** $l \in$ feuilles$(\mathcal{T}, c)$ **faire**
5: b_c.ajouter(extraire_branche(l))
6: **fin pour**
Contrôle des branches internes
7: **pour tout** $n \in$ noeuds$(\mathcal{T}, c)$ **faire**
8: **si** $n \notin b_c$ **alors**
9: b_c.ajouter(extraire_branche(n))
10: **fin si**
11: **fin pour**
12: $\mathcal{B} \leftarrow b_c$
13: **fin pour**
14: **retourner** $\mathcal{B}$

Fonction extraire_branche
Entrée : c_i concept, n noeud
Sortie : branche $b \leftarrow \emptyset$
15: **répéter**
16: $b. = n$
17: $n \leftarrow n$.pére()
18: **jusqu'à** !$(n$ ET $(n$.concept $\in \{c_i, null\})$
19: **retourner** b

end

8.6 Conclusion

Les trois modéles DBN dédiés é la génération de fragments sémantiques proposés dans ce travail se distinguent par leurs structures.

Le modéle compact considére des variables représentant des fragments sémantiques composés de frames et de FE. Son atout principal est de réduire la complexité générale du décodage. Le choix d'une variable unique pour la représentation conjointe des frames et FE est cependant une contrainte forte puisqu'elle induit l'utilisation de liens déterministes entre frames et FE.

A l'opposé, le modéle factorisé est celui qui offre la plus grande liberté de combinaison au sein des fragments sémantiques. En effet, chaque association frames - FE est évaluée au cours du décodage. Ce modéle est en contrepartie celui de complexité maximale.

Le modéle é deux niveaux présente l'avantage d'une approche non déterministe des liens frames - FE tout en ayant une complexité inférieure é celle du modéle factorisé.

L'apprentissage des distribution de probabilité utilisées par ces modéles s'appuie sur un algorithme de décomposition des arbres sémantiques en sous-branches conceptuelles.

Les modéles détaillés dans ce chapitre sont évalués sur le corpus MEDIA. Les conditions d'évaluation et les résultats obtenus sont présentés dans le chapitre suivant 9.

Chapitre 9

Expériences et résultats

Sommaire

Résumé

Ce chapitre rapporte les expériences menées pour évaluer les processus d'annotation stochastique à base de DBN utilisés sur le corpus MEDIA. *Les trois systèmes proposés dans le chapitre 8 sont appliqués à un ensemble de test, comprenant 3005 tours de parole utilisateur. Les résultats obtenus par chaque modèle sont détaillés selon la nature manuelle ou automatique des données de test considérées.*

9.1 Introduction

Pour évaluer les performances des système de composition des frames utilisant les DBN, un ensemble de données de test est préparé comme indiqué en 6.4. Les 3005 tours de parole utilisateur annotés en frames et FE par un expert forment l'ensemble de référence REF. Le système d'annotation en deux étapes à base de règles (décrit en CHAP.6) est utilisé pour produire une annotation en frames et FE sur le corpus MEDIA (transcription et annotation conceptuelle manuelles), les données de test étant exclues.

La qualité de cette annotation a été évaluée sur les données de test : l'obtention d'une F-mesure toujours supérieure à 0,9 pour l'identification des frames, FE et liens confirme la fiabilité du système et la consistance des données d'apprentissage.

Les expériences visant à évaluer les systèmes de compréhension stochastique à base de DBN proposés dans ce travail sont décrites dans la section 9.2. Les résultats obtenus sont donnés en 9.3.

9.2 Expériences

Les expériences sont menées sur l'ensemble de test dans trois conditions différentes, fonctions de la nature des données initiales :

- MAN : les tours de parole du locuteur sont manuellement transcrits et annotés en concepts ;
- SLU : les concepts de base sont décodés à partir des transcriptions manuelles des tours de parole locuteur, en utilisant le module SLU à base de DBN décrit dans (Lefèvre, 2006) ;
- ASR+SLU : les concepts sont décodés par le modèle de compréhension en utilisant la meilleure hypothèse (1-best) de séquence de mots générée par un système ASR, conforme à (Barrault et al., 2008).

Les données SLU et ASR+SLU comportent des erreurs de transcription et d'annotation conceptuelle liées à l'imperfection des systèmes qui les produisent. Les taux d'erreurs observés sur les 3005 tours de parole de test sont rappelés dans le tableau 9.1.

Type de données	SLU	ASR + SLU
Taux d'erreurs mots (%)	0,0	27
Taux d'erreurs concepts (%)	10,6	24,3

TABLE 9.1 – *Taux d'erreurs en mot et en concept observés sur les données SLU et ASR+SLU de l'ensemble de test* MEDIA.

9.3 Résultats

Toutes les expériences présentées ici ont été réalisées en utilisant GMTK (Bilmes et Zweig, 2002), outil logiciel de calcul et de manipulation des modèles graphiques et SRILM (Stolcke, 2002), outil logiciel pour les modèles de langage.

Les implémentations des trois modèles DBN proposés sont fournies dans l'Annexe C au format standard utilisé par GMTK.

Pour indiquer le dimensionnement des modèles DBN, le nombres de mots, concepts et fragments de frames-FE, frames et FE distincts utilisés pour leur entraînement est donné dans le tableau 9.2.

Modèle DBN	Mots	Concepts	Frag. frames-FE	Frag. frames	Frag. FE
compact	2201	78	636	x	x
factorisé	2201	78	x	234	339
2-niveaux	2201	78	x	234	339

TABLE 9.2 – *Cardinalités des variables de mots, concepts et des classes de fragments de frames-FE, frames et FE distincts utilisées dans les 3 types de modèles DBN (compact, factorisé et 2-niveaux).*

Le nombre total de frames, FE et liens présents sur les tours de parole de l'ensemble REF ansi que sur l'ensemble de test MEDIA est donné en 6.4 (tableau 6.2).

Les résultats des systèmes DBN sont donnés en termes de précision, rappel et F-mesure. La précision est le nombre de frames, FE ou liens corrects proposés par le système divisé par le nombre total de frame, FE ou liens proposés par le système. Le rappel est le nombre de frames, FE ou liens corrects proposés par le système divisé par le nombre total de frames, FE ou liens contenus dans l'annotation de référence. La F-mesure est la moyenne harmonique standard de la précision et du rappel.

Les fragments de frames et FE produits par les systèmes DBN sont évalués à trois niveaux :

- **Frames** : les hypothèses de frames sont considérées séparément et comparées aux frames présentes dans la référence ;
- **FE** : les hypothèses de FE sont considérées séparément et comparées aux FE présents dans la référence ;
- **Frames-FE** : les hypothèses de frames et de FE sont considérées conjointement.

Dans tous les cas, l'ordre d'apparition n'est pas pris en compte.

Pour chacun de ces niveaux, la précision p, le rappel r et la F-mesure $F\text{-}m$ sont calculés globalement sur l'ensemble des N tours de parole (ici $N = 3005$). On a donc :

$$p = \frac{\text{nb d'hypothèses correctes dans les N tours}}{\text{nb total d'hypothèses présentes dans les N tours}}$$

$$r = \frac{\text{nb d'hypothèses correctes présentes dans les N tours}}{\text{nb total d'objets sémantiques présents dans les N tours de référence}}$$

Type de données		**MAN**		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.89	*0.86*	**0.85**
	r	0.81	*0.77*	**0.76**
	$F\text{-}m$	**0.85**	***0.82***	**0.80**
	$\bar{p}$	0.94	*0.93*	**0.92**
	$\bar{r}$	0.89	*0.90*	**0.87**
	$\overline{F\text{-}m}$	**0.92**	***0.92***	**0.89**
F et FE (factorisé)	p	0.85	*0.78*	**0.77**
	r	0.83	*0.72*	**0.72**
	$F\text{-}m$	**0.83**	***0.74***	**0.73**
	$\bar{p}$	0.92	*0.89*	**0.88**
	$\bar{r}$	0.89	*0.88*	**0.86**
	$\overline{F\text{-}m}$	**0.91**	***0.88***	**0.87**
F puis FE (2-niveaux)	p	0.84	*0.76*	**0.75**
	r	0.82	*0.69*	**0.71**
	$F\text{-}m$	**0.83**	***0.73***	**0.73**
	$\bar{p}$	0.91	*0.88*	**0.85**
	$\bar{r}$	0.90	*0.86*	**0.84**
	$\overline{F\text{-}m}$	**0.91**	***0.87***	**0.85**

TABLE 9.3 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version MAN pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

$$F\text{-}m = \frac{p+r}{2}$$

Sont également évalués la précision moyenne $\bar{p}$, le rappel moyen $\bar{r}$ et la F-mesure moyenne $\bar{F\text{-}m}$ pour un tour de parole par les calculs suivants :

$$\bar{p} = \frac{\sum_{i=1}^{N} p_i}{N} \text{ où } p_i \text{ est la précision obtenue au tour } i$$

$$\bar{r} = \frac{\sum_{i=1}^{N} r_i}{N} \text{ où } r_i \text{ est le rappel obtenu au tour } i$$

$$\overline{F\text{-}m} = \frac{\sum_{i=1}^{N} F\text{-}m_i}{N} \text{ où } F\text{-}m_i \text{ est la F-mesure obtenue au tour } i$$

L'intervalle de confiance à 10% des valeurs estimées est d'amplitude 0.02.

Les systèmes apparaissent robustes à la dégradation des données d'entrées : une dégradation de plus de 20% sur les variables observées (mots et concepts) entraine une baisse des performances obtenues sur la génération des fragments sémantiques de moins de 10%.

	Type de données	SLU		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.88	*0.85*	**0.84**
	r	0.78	*0.69*	**0.71**
	$F\text{-}m$	**0.83**	***0.77***	**0.78**
	$\bar{p}$	0.93	*0.92*	**0.91**
	$\bar{r}$	0.87	*0.83*	**0.84**
	$\overline{F\text{-}m}$	**0.90**	***0.88***	**0.87**
F et FE (factorisé)	p	0.84	*0.78*	**0.77**
	r	0.81	*0.64*	**0.68**
	$F\text{-}m$	**0.83**	***0.71***	**0.74**
	$\bar{p}$	0.92	*0.89*	**0.86**
	$\bar{r}$	0.88	*0.81*	**0.82**
	$\overline{F\text{-}m}$	**0.89**	***0.85***	**0.84**
F puis FE (2-niveaux)	p	0.84	*0.75*	**0.75**
	r	0.80	*0.62*	**0.67**
	$F\text{-}m$	**0.82**	***0.69***	**0.71**
	$\bar{p}$	0.91	*0.88*	**0.85**
	$\bar{r}$	0.89	*0.80*	**0.82**
	$\overline{F\text{-}m}$	**0.90**	***0.84***	**0.83**

TABLE 9.4 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version SLU pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

On remarque également que sur les données SLU, le taux d'erreur sur les fragments est voisin du taux d'erreurs concepts observé. Le taux d'erreur concepts est majoré de 13,4% sur les données ASR+SLU (taux d'erreur mots de 27%) alors que les résultats sur les fragments ne sont dégradés que de 6%.

Les résultats des tableaux 9.3, 9.4 et 9.5 montrent que les performances du modèle compact sont supérieures à celles des deux autres modèles. Le domaine de connaissance MEDIA est défini de telle façon qu'un FE ne peut prendre qu'un nombre très limité de frames pour valeur. Ainsi, dans ce contexte, l'utilisation par le modèle compacte de liens déterministes entre frame et FE favorise la production de fragments sémantiques consistants et disposant de statistiques fiables. La simplicité du modèle compact est également un atout dans le cadre de l'intégration de ce modèle à un système de dialogue complet.

Les performances du modèle factorisé et du modèle à deux niveaux permettent de considérer que ces deux modèles sont également adaptés à la tâche de décodage de fragments sémantiques. Nous espérons pouvoir les évaluer rapidement sur la base de connaissances LUNA évoquée en 5.5. Son dimensionnement induit potentiellement un niveau d'incertitude plus élevé dans le choix des frames valeurs de FE. La liberté de combinaison des frames et FE dans les fragments offerte par le modèle factorisé et le

Type de données		ASR + SLU		
Modèles DBN		F	*FE*	**Frames-FE**
F/FE (compact)	p	0.83	*0.78*	**0.78**
	r	0.72	*0.65*	**0.67**
	$F\text{-}m$	**0.77**	***0.71***	**0.72**
	$\bar{p}$	0.87	*0.88*	**0.85**
	$\bar{r}$	0.80	*0.80*	**0.77**
	$\overline{F\text{-}m}$	**0.84**	***0.84***	**0.81**
F et FE (factorisé)	p	0.78	*0.73*	**0.72**
	r	0.75	*0.60*	**0.63**
	$F\text{-}m$	**0.76**	***0.67***	**0.69**
	$\bar{p}$	0.85	*0.84*	**0.80**
	$\bar{r}$	0.82	*0.78*	**0.76**
	$\overline{F\text{-}m}$	**0.83**	***0.82***	**0.78**
F puis FE (2-niveaux)	p	0.79	*0.71*	**0.70**
	r	0.74	*0.58*	**0.62**
	$F\text{-}m$	**0.77**	***0.65***	**0.66**
	$\bar{p}$	0.86	*0.84*	**0.80**
	$\bar{r}$	0.82	*0.77*	**0.75**
	$\overline{F\text{-}m}$	**0.84**	***0.81***	**0.77**

TABLE 9.5 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *en version ASR+SLU pour les trois systèmes de génération de fragments sémantiques à base de DBN.*

modèle à deux niveaux pourra être un atout dans ce contexte.

9.4 Conclusion

Les résultats obtenus par les systèmes évalués confirment que les modèles à base de DBN peuvent être utilisés pour générer des sous-structures sémantiques hiérarchiques consistantes. De plus, ces modèles ayant la capacité de produire des hypothèses avec leurs scores de confiance, ils peuvent être utilisés dans des contextes où les hypothèses sont multiples (réseaux de confusion, n-meilleures hypothèses) ou encore dans des protocoles d'évaluation en classant les hypothèses proposées par d'autres systèmes.

Les modèles factorisé et à deux niveaux sont aptes à produire des fragments sémantiques consistants sur les dialogues de test MEDIA. Leurs performances restent cependant inférieures à celles du modèle compact, certainement avantagé par la structure et le dimensionnement de notre base de connaissances.

Les fragments sémantiques sont générés par les DBN dans le cadre d'un processus séquentiel qui ne prend pas en compte les dépendances "longue-distance" aux obser-

vations. Ces fragments forment les constituants structurés de la représentation sémantique complète du message de l'utilisateur. Celle-ci est obtenue grâce à une étape de recomposition complémentaire présentée dans la dernière partie de ce document.

Composition des fragments sémantiques

Chapitre 10

Composition d'arbres : modèles et stratégies

Sommaire

Résumé

Ce chapitre propose une présentation de la notion d'arbre employée pour représenter les relations sémantiques dans notre contexte de travail. Il rappelle ensuite dans la section 10.3 les fondements théoriques des modèles de classification basés sur les séparateurs à vaste marge utilisés dans l'une des stratégies de composition des fragments sémantiques.

10.1 Introduction

Les fragments sémantiques générés par les DBN sont représentés par des arbres que nous composons selon deux stratégies, dont une à base de séparateurs à vaste marge (SVM). La définition des arbres et les bases théoriques des SVM sont rappelées dans ce chapitre.

Les systèmes génératifs à base de DBN présentés au chapitre précédent ont la capacité de produire des fragments d'arbres sémantiques que l'on doit ensuite recomposer.

La première partie 10.2 de ce chapitre rappelle tout d'abord quelques définitions et propriétés associées à la structure d'arbre qui est utilisée dans nos travaux pour supporter la représentation sémantique des messages utilisateur. Quelques approches classiques de composition des structures d'arbres sont ensuite présentées.

Une des stratégie de composition évaluée dans ce travail s'appuie sur un algorithme de décison à base de séparateurs à vaste marge (SVM). La seconde section de ce chapitre 10.3 s'attache à exposer les points théoriques qui sous-tendent cet algorithme de décision et définit les SVM dans leur contexte mathématique.

10.2 Notion d'arbre

La notion d'*arbre* est définie dans le cadre de la théorie des graphes. Les graphes permettent de modéliser toute situation mettant en jeu un nombre fini d'éléments en interaction. Les éléments considérés sont les *sommets* ou *nœuds* du graphe. Les interactions entre ces sommets sont matérialisées par les *arêtes* du graphe.

Un graphe G est donc bien défini par la donnée du couple (V, E) tel que V est l'ensemble des sommets de G et $E \subset V \times V$ est l'ensemble des arêtes de G. L'arête $e \in E$ ayant pour extrémités les sommets u et v de V est souvent notée $e = uv$. On se limite ici à définir les graphes simples (une seule arête relie deux sommets) et non orientés (les arêtes de G ne sont pas dirigées). On peut remarquera cependant qu'un graphe non orienté peut être considéré comme un graphe orienté tel que pour toute arête de u vers v, l'arête de v vers u appartient à E.

L'exemple du graphe G_5 est présenté 10.1 pour illustrer les définitions données ci-après :

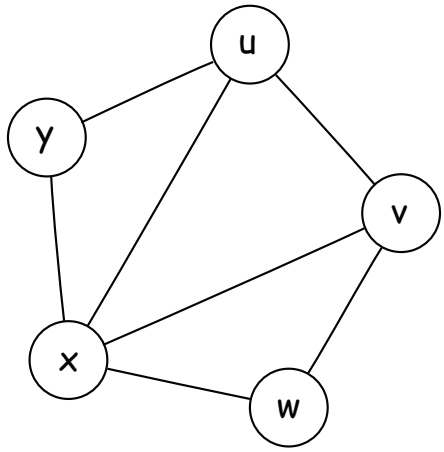


FIGURE *10.1* – *Le graphe G_5*

Un graphe a n sommets est dit d'*ordre* n tandis que sa *taille* est le nombre de ses arêtes. G_5 est d'ordre 5 et de taille 7.

Deux sommets reliés par une arête sont dits *adjacents* ou *voisins*. Le *degré* d'un sommet est le nombre de ses voisins. Les sommets u et v de G_5 sont voisins. v est de degré 3 et w est de degré 2.

Une *chaine* est une suite de sommets reliés par des arêtes et un *cycle* est une chaine dont les extrémités coïncident. On dit qu'un graphe est *connexe* si et seulement si il existe une chaine reliant toute paire de sommets. (u, v, w) est une chaine de G_5, (x, y, u) est un cycle et G_5 est connexe.

Les graphes utilisés dans ce travail sont non orientés, connexes et sans cycles. Ces graphes particuliers sont des *arbres* non orientés.

Le graphe A_5 présenté 10.2 est un arbre d'ordre 5.

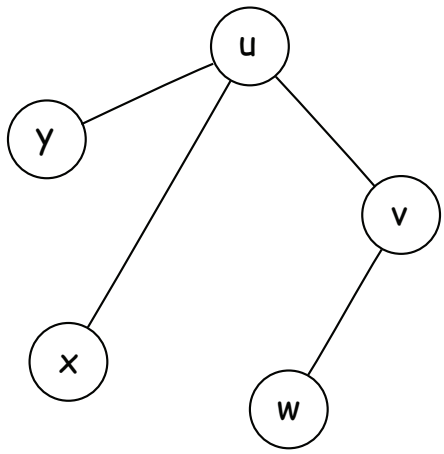


FIGURE *10.2* – *L'arbre A_5*

Les sommets de degré 1 sont les feuilles de l'arbre. A_5 possède 3 feuilles w, x et y.

La manipulation des arbres est rendue plus aisée par le choix d'une *racine*. Il s'agit d'un nœud de l'arbre qui sert de repère dans l'exploration des *branches*, chaines ayant pour extrémités la racine et une feuille. Ce choix est arbitraire dans le cas des arbres non orientés tandis que dans les arbres orientés, la racine est l'unique nœuds sans prédécesseur de l'arbre. Si l'on choisit u pour racine de l'arbre A_5, cet arbre possède alors trois branches (u, v, w), (u, x) et (u, y).

Un arbre est *étiqueté* si à chacun de ses sommets est attribuée une étiquette issue d'un ensemble fini de symboles. A_5 est étiqueté par l'ensemble $\{u, v, w, x, y\}$.

Pour être à même de comparer et de transformer des arbres étiquetés il est nécessaire de définir les opérations réalisables sur les nœuds de ces arbres. Ces opérations sont généralement de trois types : suppression, insertion et renommage. La donnée d'un ensemble d'arbre étiquetés et de ces trois opérations permet de définir plusieurs distances entre les arbres (alignement, édition) ainsi que de considérer les problèmes d'inclusion (Bille, 2005).

L'emploi d'arbres étiquetés pour représenter les connaissances syntaxyco-sémantiques associées à une proposition a été présenté dans le chapitre 1. L'usage de ces structures est également privilégié dans la manipulation des fichiers de données au format XML (*eXtensible Markup Language*). Ce langage permet de décrire des données sous forme arborescente à partir d'une structure préalablement définie. La grande variété de structures des arbres XML renouvelle l'intérêt pour la maîtrise des transformations d'arbres étiquetés. En effet, ces transformations conditionnent la communication entre les applications utilisant des données XML . Elles sont indispensables à l'utilisation des données du Web.

Les opérations de transformation sont centrales dans les taches de classification d'arbres et de découverte de motifs fréquents. Les travaux de (Candillier, 2006) adaptent différentes techniques de clustering à la classification de documents XML . Des approches algorithmiques voisines de celles que nous avons développées (voir chapitre 11) sont proposées par (Candillier et al., 2007) dans le contexte de la fouille de documents XML .

Les transformations peuvent être réalisées par des programmes dédiés à chaque application en utilisant par exemple le langage de transformation XSLT (*eXtensible Stylesheet Language Transformations*) ou des langages généralistes tels Perl ou Python. Cette approche est coûteuse et produit des solution spécifiques à chaque application, non évolutives et non génériques. Une alternative intéressante est proposée par (Jousse, 2007) en utilisant des techniques d'apprentissage supervisé : les opérations de transformation sont apprises à l'aide de modèles probabilistes à partir d'exemple d'arbres XML originaux et transformés.

Au cours des transformations, les décisions à prendre lors de la décomposition d'un arbre ou de la recomposition de branches peuvent donc se baser sur les opérations observées dans un corpus d'apprentissage. Dans ce travail, l'approche adoptée pour prendre en compte ces observations fait intervenir des techniques de classification automa-

tique supervisée. Des classifieurs à base de machines à vecteurs supports ou séparateurs à vaste marge (SVM) sont utilisés. Les notions de base permettant d'appréhender leur fonctionnement sont présentées dans la partie suivante 10.3.

10.3 Séparateurs à vaste marge

La classification a pour but de regrouper des objets de même nature en fonction de certaines de leurs caractéristiques. Chaque groupe d'objets forme une *classe*. Dans le contexte de la classification automatique, un *classifieur* désigne un algorithme permettant d'attribuer une classe à un objet à partir de l'observation de ses caractéristiques.

Les méthodes de classification non supervisée cherchent à partitionner l'ensemble des objets en groupes d'objets similaires sans qu'aucune partition *apriori* ne soit fournie. Ces méthodes, utilisées parfois en classification sémantique comme l'analyse latente sémantique (Bellegarda, 2007), ne permettent toutefois pas une extraction fine des composants. Elles sont adaptées à des applications de routage d'appels ou de classification de phrases.

Dans les méthodes de classification supervisée, l'ensemble des classes est fixé. L'application de ces méthodes à l'analyse sémantique est proposée par (Pradhan et al., 2004). Deux classes au moins sont définies et la répartition des données d'apprentissage au sein de ces classes est connue, ce qui justifie l'appellation "*supervisée*" de cette classification. Une donnée dont la classe d'appartenance est connue est souvent qualifiée de donnée *étiquetée*.

On dispose d'un ensemble X de n données étiquetées et d'un ensemble fini U de k classes. Chaque donnée $x_{i\in[\![1,n]\!]}$ est caractérisée par p caractéristiques et par sa classe $u_i \in U$. Le problème de classification consiste à prédire la classe de toute nouvelle données x en s'appuyant sur la connaissance des données de X. Procédant par induction puisqu'ils prédisent une connaissance plus générale à partir d'un ensemble de cas particuliers, les classifieurs produits dans ce contexte ont donc de bonnes capacités de généralisation.

Les données sont décrites vectoriellement dans un espace de Hilbert [1] de dimension p. Dans ce travail, la classification opérée est binaire : $U = \{-1, 1\}$. Les données sont étiquetées *positives* si elles sont de classe 1 et *négatives* si elles sont de classe -1.

Quand le problème de classification n'est pas linéairement séparable dans l'espace originel, il peut le devenir en réalisant un *déplacement* des données dans un espace de dimension plus élevée (Cover, 1965).

Dans l'exemple donné en 10.3, l'espace initial de représentation des données est $\mathbb{R}^2$ dans lequel les données d'entraînement ne sont pas linéairement séparables. Après déplacement dans $\mathbb{R}^3$, la séparation linéaire des données est réalisée par un hyperplan de $\mathbb{R}^3$.

1. Un espace de Hilbert est un espace vectoriel muni d'un produit scalaire, complet pour la norme associée. $\mathbb{R}^p$ muni du produit scalaire est un espace de Hilbert.

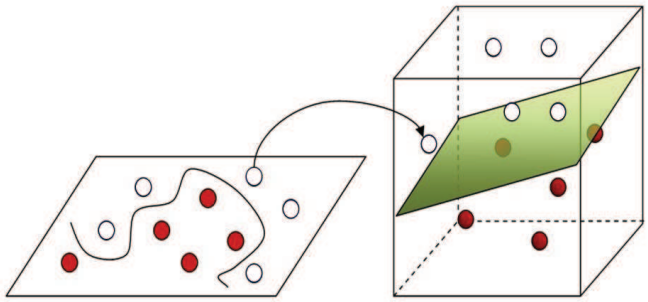


FIGURE 10.3 – *Déplacement de l'espace de représentation vers un espace de dimension supérieure*

Si le problème de classification est linéairement séparable, il existe une famille infinie de formes linéaires discriminantes qui peut lui être associée.

Toute forme de cette famille s'écrit $C(x) = \vec{w}.\vec{x} + w_0$ et $\forall i \in [\![1, n]\!]$ on a :

$$C(x_i) > 0 \Rightarrow u_i = 1$$
$$\text{et}$$
$$C(x_i) < 0 \Rightarrow u_i = -1$$

$$\text{soit} : \forall i \in [\![1, n]\!], \quad u_i C(x_i) > 0.$$

Il existe ainsi une infinité d'hyperplans capables de séparer les données positives des données négatives, chacun de ces hyperplans étant le noyau d'un forme linéaire discriminante associée au problème de classification.

Soit H un hyperplan séparateur d'équation $y = \vec{w}.\vec{x} + w_0$ ($\vec{w}$ normal à H). Soient H^+ et H^- les hyperplans parallèles à H contenant respectivement les éléments positifs et négatifs de X les plus proches de H.

La figure 10.4 illustre cette situation en dimension 2. Les données positives sont représentées en noir tandis que les données négatives sont en clair. H sépare les données positives des négatives, H^+ et H^- sont tous deux parallèles à H. A et B appartiennent respectivement à H^+ et H^- tels que la distance AB est minimale.

Un classifieur à base de séparateurs à vaste marge (SVM) linéaires détermine H tel que la distance entre H^+ et H^-, appelée la *marge*, est maximale. Les SVM font partie des méthodes à noyaux, inspirées de la théorie mathématique de l'apprentissage développée depuis les années 1960 par Vapnik et Chervonenkis (théorie VC) (Vapnik, 1995, 1998).

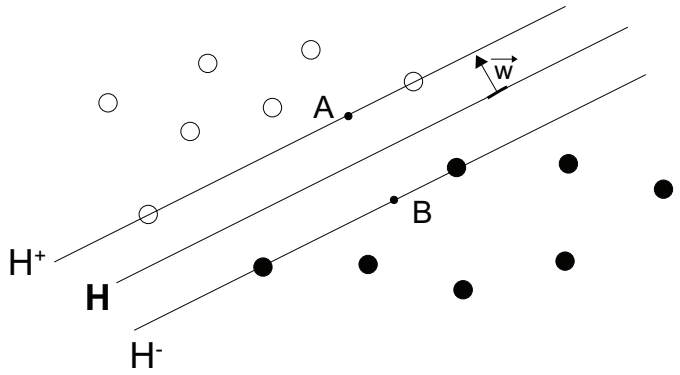


FIGURE *10.4* – *Schéma représentant la séparation de données par un hyperplan H*

Le problème étant linéairement séparable, on peut choisir :
$H^+ : \quad \vec{w}.\vec{x} + w_0 = 1$
$H^- : \quad \vec{w}.\vec{x} + w_0 = -1$ sous les contraintes $\forall i \in [\![1, n]\!] \qquad u_i C(x_i) \geq 1.$

La distance de tout point $x^- \in H^-$ à H est alors $d(x^-, H) = \dfrac{|\vec{w}.\vec{x^-} + w_0|}{\|\vec{w}\|}$ et de même la distance de tout point $x^+ \in H^+$ à H est alors $d(x^+, H) = \dfrac{|\vec{w}.\vec{x^+} + w_0|}{\|\vec{w}\|}$.

La marge λ s'écrit alors : $\lambda = \dfrac{2}{\|\vec{w}\|}$.

Maximiser λ sous les contraintes $\forall i \in [\![1, n]\!] \qquad u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$ revient donc à minimiser $\|\vec{w}\|$ ou encore $\dfrac{1}{2}\|\vec{w}\|^2$ sous les mêmes contraintes.

La recherche de l'hyperplan optimal se ramène à résoudre le problème d'optimisation sur $\vec{w}$ et w_0 :

Minimiser	$\frac{1}{2}\|\vec{w}\|^2$	
sous les contraintes	$\forall i \in [\![1, n]\!]$	$u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$

Le problème ainsi énoncé en forme primale impose une résolution en dimension $p + 1$, les données étant décrites dans $\mathbb{R}^p$, ce qui est d'autant plus complexe que p est grand. Cela compromet l'obtention de solution dans l'espace de grande dimension dans lequel les données ont été projetées pour obtenir un problème de classification linéairement séparable.

L'expression du problème d'optimisation dans sa forme duale permet de contourner cet écueil. Les contraintes du problème étant toutes linéaires, on peut appliquer la méthode des multiplicateurs de Lagrange pour transformer le problème d'optimisation sous contraintes en un problème d'optimisation sans contrainte ayant la même solution. L'application de cette méthode est détaillée dans l'Annexe D.

Ainsi, l'utilisation de la méthode de Lagrange permet de démontrer que **seules les données correspondant aux vecteurs supports sont utiles à l'apprentissage**.

Parmi les évolutions récentes des méthodes à base de SVM, il est intéressant de signaler l'introduction des modèles *soft margin* pour lequels la contrainte de marge est assouplie.

Cette contrainte $u_i(\vec{w}.\vec{x_i} + w_0) \geq 1$, utilisée dans le modèle classique précédement présenté, devient $u_i(\vec{w}.\vec{x_i} + w_0) \geq 1 - \xi_i$ avec ξ_i proche de 0 variables d'erreurs.

L'introduction de ces variables permet de séparer linéairement les données au mieux tout en ignorant quelques exemples mal classés.

Dans le cas de problèmes non linéairement séparables dans l'espace initial, il est intéressant de remarquer que l'heuristique de déplacement vers un espace de grande dimension est indépendante du choix de l'algorithme de classification. Cependant, les classifieurs à base de SVM sont particulièrement bien adaptés à cette approche.

En effet, la classification dans l'espace de dimension supérieure nécessite seulement la connaissance :
- de la fonction de déplacement Φ (non linéaire) ;
- du mode de calcul des produits scalaires dans l'espace d'arrivée en fonction des vecteurs de l'espace initial ($\Phi(\vec{x}).\Phi(\vec{y})$).

Si l'on suppose l'existence d'une fonction **noyau** K telle que $\Phi(\vec{x}).\Phi(\vec{y}) = K(\vec{x}, \vec{y})$, il n'est plus nécessaire de connaitre la fonction de déplacement Φ. L'*astuce du noyau* (*Kernel Trick*) est attractive puisqu'elle permet d'utiliser des noyaux variés [2].

En conclusion, les méthodes à base de SVM permettent de traiter des problèmes de grande dimension. Essentiellement dépendantes des vecteurs supports, elles produisent des résultats pertinents même si les données d'apprentissage sont peu nombreuses. Elles offrent ainsi un bon compromis entre capacité de généralisation et complexité.

De nombreuses librairies libres implémentent les méthodes à base de SVM [3].

10.4 Conclusion

La notion d'arbre définie dans ce chapitre permet de modéliser et de manipuler les fragments sémantiques. Le chapitre suivant expose comment les opérations de composition des fragments produits par les DBN sont effectuées sur des structures d'arbres.

Pour appliquer les opérations de composition, nous proposons une approche heuristique et un algorithme de décision. Notre algorithme de décision repose sur l'utilisation des classifieurs SVM. Le modèle théorique des SVM, décrit dans ce chapitre, met en évidence leur capacité discrimination. En effet, la dépendance des paramètres de ces modèles aux seuls vecteurs supports a pour avantage qu'un ensemble de données

2. Tout noyau respectant la condition de Mercer est admissible(Vapnik, 1998). Cette condition s'écrit $\forall g$ tq $\int g(x)^2 dx$ est finie, $\iint K(x,y)g(x)g(y)dxdy \geq 0$ et elle garantit l'existence d'une solution au problème quadratique duale

3. Une liste non exhaustive de ces librairies peut être consultée à l'addresse `http://www.support-vector-machines.org/SVM_soft.html`.

d'apprentissage de taille restreinte ne compromet pas le niveau de performance de ces méthodes.

Approche heuristique et algorithme de décision à base de SVM sont présentés dans le chapitre suivant 11. L'utilisation des SVM dans notre contexte applicatif y est également détaillée.

Chapitre 11

Approches pour la recomposition de fragments sémantiques

Sommaire

Résumé

Ce chapitre présente les algorithmes de recomposition d'arbres utilisés pour finaliser le processus de compréhension à partir des fragments de structures proposés par le décodage en DBN. Deux approches sont proposées et évaluées dans ce chapitre : la méthode de connection forte et la méthode de connection par classifieur.

11.1 Introduction

A l'issue du décodage sémantique réalisé par les modèles DBN présentés au chapitre 8, le système de compréhension dispose de fragments structures composés de frames et de FE. Ces fragments sont des entités sémantiques composées qui représentent des situations élémentaires incluses dans le message à interpréter. Les relations sémantiques liant ces situations élémentaires ne sont pas toujours décodées par les modèles DBN. En effet, ces relations sont souvent portées par des dépendances "longue-distance", non modélisables par les DBN.

La capture de ces relations nécessite donc l'examen global du message. Dans le contexte de dialogue du corpus MEDIA, le message de l'utilisateur est borné par les interventions de l'opérateur simulant le serveur vocal. Les relations entre les fragments sémantiques produits par les DBN sont donc recherchées en considérant un tour de parole complet de l'utilisateur.

Deux algorithmes de recomposition des arbres sémantiques dont les fragments (i.e. branches, ou sous-arbres) sont produits par les DBN ont été évalués. Ils sont détaillés dans la section 11.2. La première approche est déterministe, la seconde s'appuie sur les décisions de classifieurs SVM.

11.2 Composition d'arbres

Les arbres étiquetés ont été définis en 10.2. Leur utilisation pour représenter les connaissances sémantiques et leur manipulation dans ce travail nécessitent la définition des opérations pouvant être réalisées sur ces arbres.

Comme décrit en 8.5, l'apprentissage des paramètres des modèles DBN nécessite la projection de l'arbre de frames et FE associé à la phrase complète pour relier sous-branches sémantiques et concepts de base. Lors de cette projection, des opérations de deux types sont réalisées. Les opérations de *séparation* rompent des liens entre frames et FE selon les concepts qui leurs sont associés. Les opérations de *duplication* des objets sémantiques (frames ou FE) sont nécessaires lorsque ces objets sont présents dans plusieurs sous-branches distinctes.

L'algorithme de recomposition est développé pour rassembler les fragments sémantiques produits par les DBN et rétablir l'arbre sémantique associé à la globalité du message. Il décide des opérations réciproques de celles effectuées lors de la projection, soit des opérations de *liaison* entre frames et FE et des opérations de *regroupement* entre frames ou FE.

Les liaisons potentielles inter-fragments s'appuient sur l'ontologie en frames et FE développée pour le domaine du corpus MEDIA : deux objets sémantiques ne peuvent être reliés que s'ils le sont dans l'ontologie. Ainsi, deux sous-branches sémantiques ne peuvent être connectées que si elles portent des nœuds dont les labels sont reliés dans l'ontologie. Les opérations de liaison consistent donc en l'ajout d'arêtes entre des

nœuds de sous-branches sémantiques distinctes pour les rassembler sous un arbre sémantique unique.

Les identifications potentielles concernent les objets sémantiques semblables présents au sein de plusieurs fragments associés à un même message. L'algorithme de recomposition considère ces objets et décide de la pertinence de leur présences multiples. Les opérations de regroupement ont ainsi pour rôle de supprimer les objets sémantiques redondants produits par les DBN. Lorsque deux nœuds de sous-branches sont identifiés, un seul est conservé dans l'arbre sémantique global et les nœuds fils du nœud supprimé sont reliés au nœud conservé.

L'exemple donné dans la figure 11.1 illustre ces processus. L'arbre sémantique associé au message "réserver un hôtel à Bourg-en-Bresse" est reproduit sur la gauche de la figure. Les branches de l'arbre décomposé sont reproduites à droite. Lors de la décomposition pour l'apprentissage des paramètres des modèles DBN, la frame `LODGING` est dupliquée et le FE `reserve_theme` est séparé d'une des deux instances de cette frame.

L'algorithme de recomposition doit être à même de recomposer l'arbre complet en disposant du message, des concepts associés et des branches générées par les DBN à partir de ces connaissances. Dans notre exemple, les deux frames `LODGING` doivent être regroupées. La liaison entre le FE `reserve_theme` et la frame `LODGING` résultante est naturellement réalisée dans ce cas. On remarquera que le message "réserver un séjour à Bourg-en-Bresse" aurait généré les branches `RESERVE-reserve_theme` et `LODGING-lodging_location-LOCATION-location_town`. L'algorithme de recomposition aurait eu dans ce cas à décider de la liaison entre le FE `reserve_theme` et la frame `LODGING`.

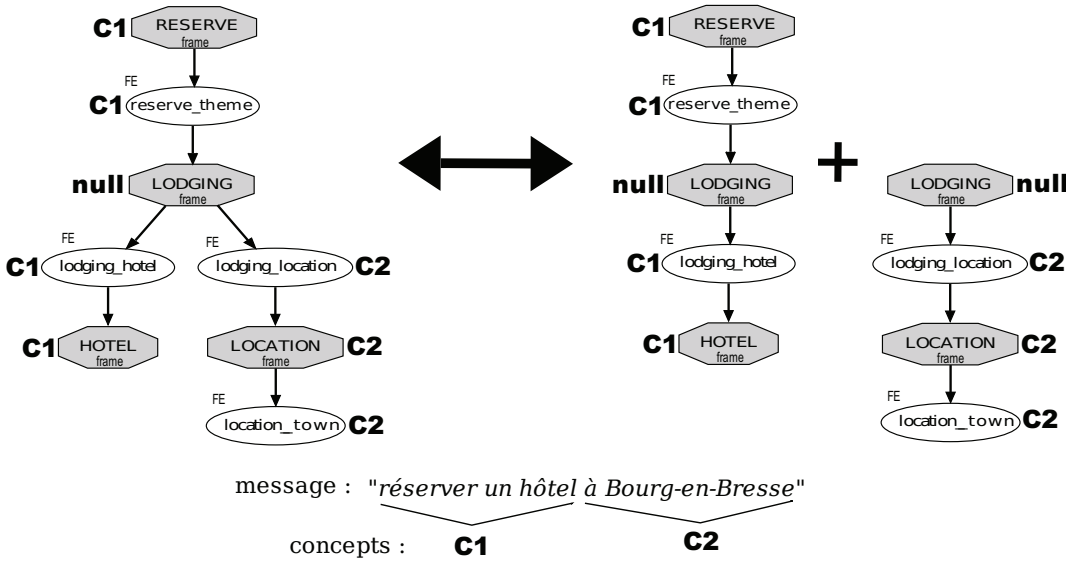


FIGURE 11.1 – *Décomposition et recomposition de l'arbre sémantique associé au message "réserver un hôtel à Bourg-en-Bresse"*

La pertinence de l'arbre sémantique recomposé est donc directement dépendante de

la pertinence des décisions de liaisons et de regroupement. Deux stratégies de décision sont évaluées dans ce travail. Elles sont présentées ci-après.

11.3 Stratégies de décision

11.3.1 Méthode de connexion forte

La première stratégie évaluée est une heuristique visant à obtenir pour chaque message une représentation sémantique compacte dans le cadre autorisé par l'ontologie. Dans cette méthode de **connexion forte**, toute liaison ou regroupement, possible selon l'ontologie, est réalisée.

Cette approche est a priori efficace pour les messages simples contenant des phrases courtes et peu ambigües. En revanche, elle ne prend pas en compte les mots et les concepts associés au message. Elle n'est donc pas très adaptée aux messages complexes dont la représentation sémantique peut contenir de nombreux sous-structures non connectées. Le principe de cette heuristique est présenté dans la figure 11.2.

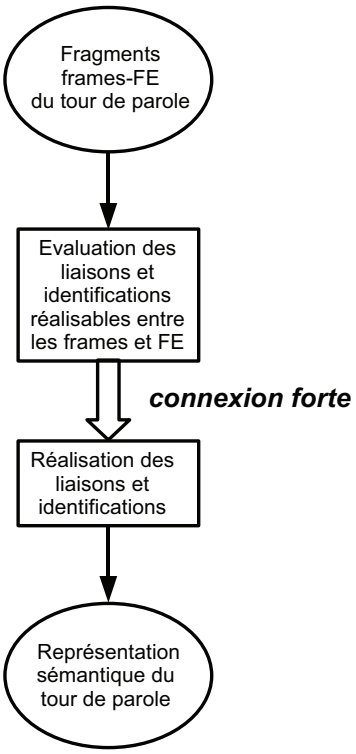


FIGURE 11.2 – *Principe d'application de la méthode de* ***connexion forte***

11.3.2 Méthode de connexion par classifieur

La seconde stratégie évaluée s'appuie sur une méthode de connexion basée sur l'apprentissage de classifieurs SVM, que nous avons nommée **connexion SVM**. Le choix du type de classifieurs linéaires employés est dicté par plusieurs considérations : la quantité de données disponibles, la rapidité de réponse ou encore les performances obtenues sur des données comparables. En raison de leurs propriétés, décrites en 10.3, les classifieurs SVM s'adaptent parfaitement au contexte applicatif de ce travail.

Apprentissage
Les opérations de séparation et de duplication réalisées lors de la projection des arbres nécessaire à l'apprentissage des paramètres des DBN sont recensées. L'algorithme utilisé est donné ci-après (Algorithme 2).

t] Algorithme de projection d'arbre avec extraction des opérations réalisées

Entrée : $\{c_i\}$ séquence de concepts associés à la phrase, $\mathcal{T}$ arbre de frames et FE représentant le message
Sortie : $\mathcal{B}$ ensemble des branches

$\mathcal{B} \leftarrow \emptyset$
pour tout $c \in \{c_i\}$ **faire**
  branche $b_c \leftarrow \emptyset$
  Génération des branches principales
  pour tout $l \in$ feuilles$(\mathcal{T}, c)$ **faire**
    si l.concept() $==$ c_i **alors**
      b_c.ajouter(extraire_branche(l))
    fin si
  fin pour
  Contrôle des branches internes
  pour tout $n \in$ noeuds$(\mathcal{T}, c)$ **faire**
    si $n \notin b_c$ ET n.concept() $==$ c_i **alors**
      b_c.ajouter(extraire_branche(n))
    fin si
  fin pour
  $\mathcal{B} \leftarrow b_c$
fin pour
retourner $\mathcal{B}$

Fonction extraire_branche
Entrée : c_i concept, n noeud
Sortie : branche $b \leftarrow \emptyset$
répéter
  $b. = n$
  si $\mathcal{B}$.contient(n) **alors**
    duplication/regroupement(n)
  fin si
  $n \leftarrow n$.père()
jusqu'à !$(n$ ET $(n$.concept $\in \{c_i, null\})$
si n **alors**
  séparation/liaison(b.last(),n)
fin si

27: **retourner** b

end

A chaque opération est associé l'ensemble des exemples du corpus d'entraînement contenant les objets sémantiques qu'elle fait intervenir. Ces messages sont répartis en deux classes selon qu'ils ont ou non *déclenché* l'opération.

On dispose de $\mathcal{T}$, ensemble des exemples d'apprentissage annotés en arbres sémantiques par le système à base de règles décrit au chapitre 6.

Soit $\mathcal{A}$ l'ensemble construit à partir de $\mathcal{T}$ tel que tout élément de $\mathcal{A}$ est composé des mots, des concepts et de l'arbre sémantique associés à un exemple de $\mathcal{T}$.

Soit $\mathcal{A}^p$ l'ensemble construit à partir de $\mathcal{A}$ tel que tout élément de $\mathcal{A}^p$ est composé :
- des mots,
- des concepts,
- des fragments sémantiques obtenus après projection de l'arbre sémantique,
- des opérations de projection (séparation, duplication) réalisées lors de la projection de l'arbre sémantique

d'un exemple de $\mathcal{A}$.

Soient $\mathbb{O}$ l'ensemble des opérations observées dans $\mathcal{A}^p$. Par souci de simplification, une opération de projection et sa réciproque de recomposition (ou *regroupement*) seront également notées $\mathcal{O}_i$, le contexte d'application levant toute ambigüité.

Chaque opération de projection $\mathcal{O}_i \in \mathbb{O}$ met en jeu deux objets sémantiques f_{i1} et f_{i2} et on notera $\mathcal{O}_i = f_{i1} \mathcal{R} f_{i2}$.

Pour chaque paire $\{f_{i1}, f_{i2}\}$ associée à une opération de $\mathbb{O}$, on construit l'ensemble $\mathcal{A}_i^p$ des exemples de $\mathcal{A}^p$ contenant f_{i1} et f_{i2}.

Les exemples de $\mathcal{A}_i^p$ pour lesquels l'opération $\mathcal{O}_i$ s'est appliquée lors de la projection sont dits "positifs" pour $\mathcal{O}_i$.
Les exemples $\mathcal{A}_i^p$ contenant s_{i1} et s_{i2} pour lesquels $\mathcal{O}_i$ n'a pas été appliquée sont "négatifs" pour $\mathcal{O}_i$.

On dispose pour chaque opération $\mathcal{O}_i$ de la partition $\{\mathcal{A}_i^{p+}, \mathcal{A}_i^{p-}\}$ de $\mathcal{A}_i^p$ où $\mathcal{A}_i^{p+}$ et $\mathcal{A}_i^{p-}$ sont respectivement les sous-ensembles d'exemples positifs et négatifs de $\mathcal{A}_i^p$.

Pour permettre l'emploi de la méthode de classification SVM, telle que décrite en 10.3, il est nécessaire de plonger les données dans $\mathbb{R}^n$. Un exemple $\mathcal{E}$ est représenté dans $\mathbb{R}^n$ par un point E dont les coordonnées sont les index numériques :
- des mots et trigrammes de mots de l'exemple ;
- de la séquence de concepts associée à l'exemple ;
- des frames et FE présents dans les fragments sémantiques associés à cet exemple.

L'introduction des n-grammes de mots dans le point caractérisant un exemple permet de prendre en compte une information séquentielle.

Les paramètres d'un classifieur linéaire binaire à base de SVM sont appris sur les points représentant les exemples de chaque ensemble $\mathcal{A}_i^p$. A l'issue de cette procédure, on

dispose d'un classifieur $\mathcal{S}_i$ par opération $\mathcal{O}_i$ et on a $|\mathbb{O}| = |\mathbb{S}| = I$, avec $\mathbb{S}$ l'ensemble des classifieurs entrainés.

Le tableau 11.1 donne un exemple de l'élaboration des caractéristiques des cas positif et négatif pour l'opération de regroupement de deux occurences de la frame HOTEL.

	Message **POSITIF**	Message **NÉGATIF**
	Caractéristiques du point représentant le message	
mots	"Ibis Prado"	"Ibis ou Mercure"
concepts	hotel-marque nom-hotel	hotel-marque connectattr hotel-marque
frames et FE	HOTEL hotel_name hotel_marque	HOTEL hotel_marque
	Fragments sémantiques associés au message	
avant projection	HOTEL-hotel_name-hotel_marque	HOTEL-hotel_marque HOTEL-hotel_marque
après projection	HOTEL-hotel_name HOTEL-hotel_marque	HOTEL-hotel_marque HOTEL-hotel_marque

TABLE 11.1 – *Caractéristiques des messages positif et négatif dans le cas du regroupement de deux occurences de la frame HOTEL.*

Les deux exemples ont en commun de posséder deux frames HOTEL dans les fragments projetés qui leur sont associés. Avant projection, le fragment sémantique du cas positif possède une unique frame HOTEL. En effet, le message mentionne un unique hôtel, l'Ibis Prado. Après projection, la frame HOTEL est dupliquée. Le cas est donc positif pour l'opération de regroupement de deux frames HOTEL.

Les frames HOTEL du cas négatif sont présentes dans les fragments sémantiques *avant* projection. Elles sont distinctes et symbolisent les deux (marques d') hôtels différent(e)s mentionné(e)s par l'utilisateur, Ibis et Mercure. L'exemple appartient donc à l'ensemble des cas négatifs pour l'opération de regroupement de deux frames HOTEL.

Application aux exemples de l'ensemble de test
Pour chaque exemple $\mathcal{E}$ de l'ensemble de test, annoté en fragments sémantiques, on construit l'ensemble des opérations pouvant le concerner en fonction des paires d'objets sémantiques contenues dans ses fragments.

Soit $\mathbb{O}^{\mathcal{E}}$ cet ensemble, on a :

$$\mathbb{O}^{\mathcal{E}} = \{\mathcal{O}_{i\in I} = f_{i1}\mathcal{R}f_{i2} \text{ tq } f_{i1} \text{ et } f_{i2} \text{ appartiennent aux fragments sémantiques associés à } \mathcal{E}\}$$

et $\mathbb{O}^{\mathcal{E}} \subset \mathbb{O}$.

Pour toute opération $\mathcal{O}_i \in \mathbb{O}^{\mathcal{E}}$, le point E représentant l'exemple $\mathcal{E}$ est soumis au classifieur S_i. La réponse de S_i quant à la classe de E détermine la pertinence de la réalisation de $\mathcal{O}_i$ sur les objets sémantiques de l'exemple.

A l'issue de ce processus, la phase de composition sémantique est achevée par la réalisation sur les objets sémantiques de $\mathcal{E}$ de toutes les opérations jugées pertinentes par les $S_{i \in I}$.

Le principe général d'application de cette méthode est présenté dans la figure 11.3.

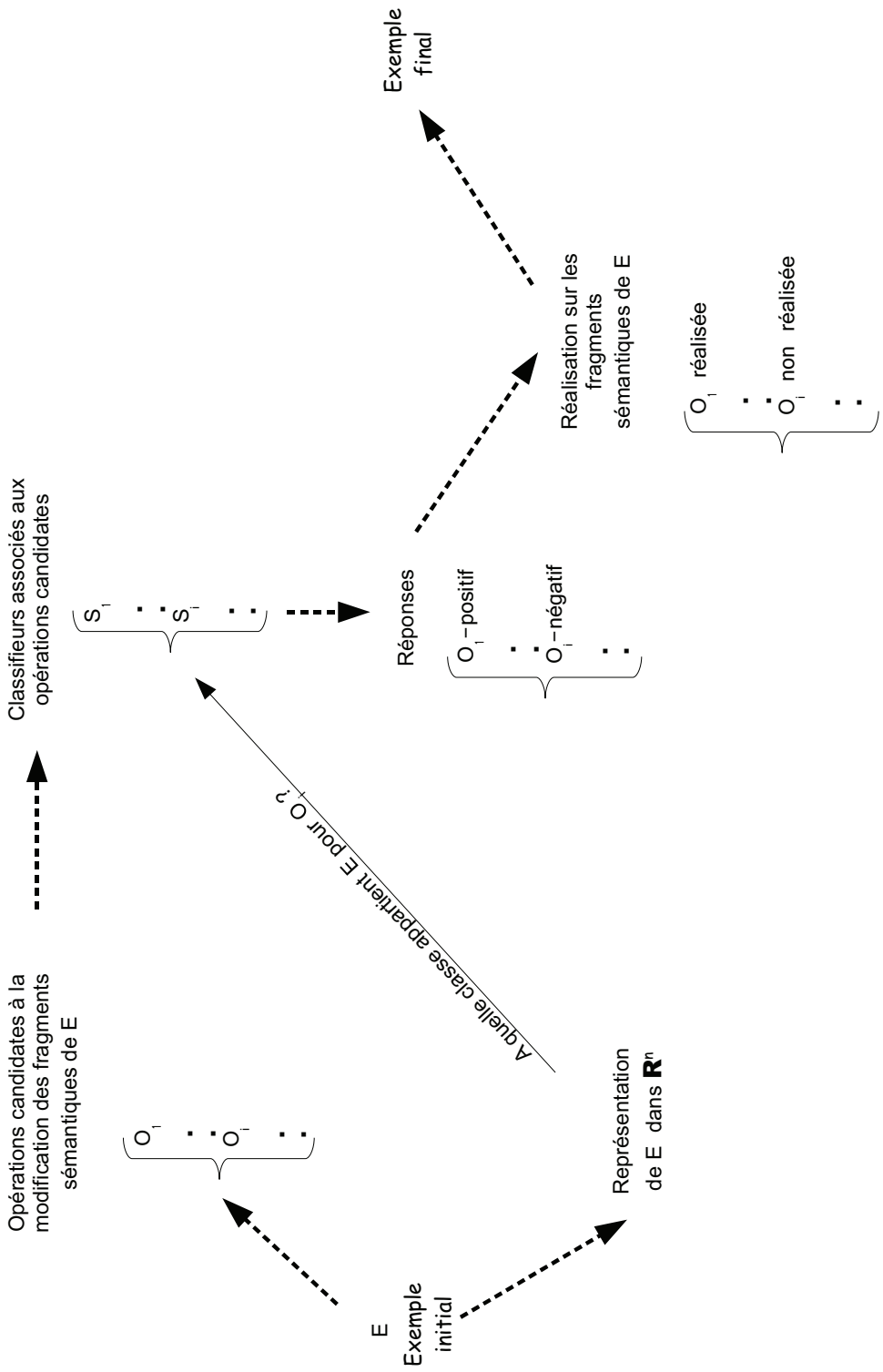


FIGURE *11.3* – *Principe d'application de la méthode de* ***connexion par classifieur SVM***

11.4 Conclusion

Les méthodes de recomposition sémantiques que nous développons dans ce travail sont basées sur le formalisme des arbres sémantiques. Les arbres utilisés sont non orientés. Deux types d'opérations permettent de les recomposer : le regroupement de deux nœuds de même étiquette sémantique et la liaison de deux nœuds d'étiquettes reliées dans la base de connaissances.

Deux méthodes sont proposées pour décider de la réalisation des opérations de recomposition. La méthode de **connexion forte** est déterministe. Elle considère les objets des fragments sémantiques d'un message et réalise toutes les opérations de recomposition compatibles avec les contraintes relationnelles de la base de connaissances.

La méthode de **connexion par classifieur SVM** considère toutes les informations caractérisant un message (mots, concepts, fragments sémantiques). Un ensemble de classifieurs SVM dediés lui permet de décider des opérations à réaliser sur les fragments sémantiques de chaque message.

Ces deux méthodes sont évaluées dans le chapitre 12 sur les fragments sémantiques produits par le modèle DBN compact.

Chapitre 12

Expériences et résultats

Sommaire

Résumé

Ce chapitre décrit les expériences menées pour évaluer les algorithmes de composition des fragments sémantiques présentés dans le chapitre 11. Les deux algorithmes proposés sont appliqués à la composition des fragments sémantiques produits par le modèle DBN compact sur l'ensemble de test de MEDIA. *Les résultats obtenus par chaque algorithme sont détaillés selon la nature manuelle ou automatique des données de test utilisées.*

12.1 Introduction

Deux méthodes de composition des fragments sémantiques sont présentées dans le chapitre précédent 11 : la méthode de connexion forte et la méthode de connexion SVM. Ces deux méthodes sont évaluées sur les tours de parole utilisateur annotés en frames et FE de l'ensemble de test MEDIA.

Les messages d'entraînement annotés par le système à base de règles décrit en 6 permettent l'entraînement des classifieurs SVM utilisés par la méthode de connexion SVM.

Les expériences réalisées sont décrites dans la section 12.2 et les résultats obtenus sont donnés en 12.3.

12.2 Expériences

De même que lors de l'évaluation des systèmes DBN (Chapitre 9), les expériences sont menées sur l'ensemble des 3005 tours de parole de test MEDIA dans trois conditions différentes, fonctions de la nature des données utilisées :

- MAN : les tours de parole du locuteur sont manuellement transcrits et annotés en concepts ;
- SLU : les concepts de base sont décodés à partir des transcriptions manuelles des tours de parole locuteur, en utilisant le modèle SLU à base de DBN décrit dans (Lefèvre, 2006) ;
- ASR+SLU : les concepts sont décodés par le modèle de compréhension en utilisant la meilleure hypothèse (1-best) de séquence de mots générée par un système ASR conforme à (Barrault et al., 2008).

Les données SLU et ASR+SLU comportent des erreurs de transcription et d'annotation conceptuelle liées à l'imperfection des systèmes qui les produisent. Les taux d'erreurs observés sur le test sont rappelés dans le tableau 12.1.

Type de données	SLU	ASR + SLU
Taux d'erreurs mots (%)	0,0	27
Taux d'erreurs concepts (%)	10,6	24,3

TABLE 12.1 – *Taux d'erreurs en mot et en concept observés sur les données SLU et ASR+SLU de l'ensemble de test* MEDIA.

Pour chaque ensemble de données MAN, SLU et ASR+SLU, les fragments sémantiques sont générés par le modèle DBN compact du chapitre 8.

Les différents niveaux d'évaluation sont détaillés ci-dessous :

- **Frames** : les hypothèses de frames sont considérées comme correctes dès lors que les frames correspondantes sont présentes dans la référence.
- **FE** : les hypothèses de FE sont considérées comme correctes dès lors que les FE correspondants sont présents dans la référence ;

– **FE{Frames}** : seules les hypothèses de FE appartenant à des hypothèses de frames correctes sont examinées. L'ensemble de référence est restreint aux FE appartenant aux frames correspondantes dans la référence ;
– **Liens** : les hypothèses de liens sont considérées comme correctes dès lors que les liens correspondants sont présents dans la référence.
– **Liens{Frames}** : seules les hypothèses de liens reliant des hypothèses de frames et FE correctes sont examinées. L'ensemble de référence est restreint aux liens reliant les frames et FE correspondants dans la référence.

Toutes les expérimentations reportées dans ce document ont été réalisées en utilisant la librairie `libSVM` (Chang et Lin, 2001; EL-Manzalawy et Honavar, 2005) pour WEKA (Witten et Frank, 2005; Bouckaert et al., 2008).

12.3 Résultats

Le tableau 12.2 regroupe les résultats issus de l'application des méthodes de connexion forte et SVM sur les fragments sémantiques issus du système basé sur le modèle DBN compact présenté au chapitre 8.

Les résultats sont mesurés en termes de précision, rappel, F-mesure et leurs valeurs moyennes sur un tour de parole pour les trois ensembles de données (MAN, SLU et ASR+SLU)

La méthode de connexion SVM fait usage de 105 classifieurs appris sur le corpus d'entraînement et répartis comme suit :
– 44 classifieurs sont dédiés à l'identification de frames (18) ou de FE (26),
– 61 classifieurs sont dédiés à la liaison de frames et FE.

Les deux algorithmes proposés obtiennent des résultats similaires sur l'ensemble de test MEDIA. Ces résultats confirment l'aptitude de ces algorithmes à composer les fragments sémantiques pour former une représentation complète consistante du message de l'utilisateur.

La structure de la base de connaissance et le contexte relativement fermé des messages de test peuvent expliquer les performances quasi identiques des deux méthodes. En effet, les opérations de regroupement et de liaison des objets sémantiques contenus dans les fragments étant presque toujours justifiées, la méthode de connexion forte commet finalement peu d'erreurs.

Les résultats obtenus par la méthode de connexion SVM permettent d'envisager l'évaluation de cette méthode selon plusieurs axes. L'influence de l'augmentation du dimensionnement de l'espace de travail des classifieurs SVM sur les performances de la méthode pourra être évaluée grâce à la base connaissance LUNA, plus vaste que celle utilisée dans ce travail. Parallèlement, la sélection des tours de parole les plus complexes de l'ensemble de test est en cours. Les deux méthodes pourront prochainement être évaluées sur ce sous-ensemble de tours de parole.

Données	Connexion forte			Connexion SVM		
	MAN					
	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$
Frames	0.93/0.88	0.95/0.88	*0.93/0.88*	0.95/0.92	0.94/0.86	*0.93/0.89*
FE	0.86/0.73	0.94/0.88	*0.88/0.80*	0.87/0.77	0.94/0.87	*0.88/0.81*
FE{Frames}	0.91/0.84	0.99/0.99	*0.94/0.91*	0.91/0.84	0.99/0.98	*0.94/0.90*
Liens	0.82/0.64	0.91/0.77	*0.82/0.70*	0.82/0.66	0.91/0.76	*0.82/0.71*
Liens{Frames}	0.88/0.73	0.98/0.96	*0.91/0.83*	0.88/0.75	0.98/0.96	*0.91/0.84*
Données	**SLU**					
	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$
Frames	0.90/0.84	0.92/0.85	*0.89/0.85*	0.91/0.87	0.92/0.83	*0.89/0.85*
FE	0.83/0.70	0.91/0.84	*0.84/0.76*	0.84/0.73	0.91/0.82	*0.84/0.77*
FE{Frames}	0.91/0.84	0.98/0.97	*0.93/0.90*	0.91/0.83	0.98/0.96	*0.93/0.89*
Liens	0.80/0.61	0.90/0.74	*0.79/0.67*	0.80/0.63	0.89/0.73	*0.79/0.67*
Liens{Frames}	0.88/0.74	0.98/0.95	*0.90/0.83*	0.88/0.75	0.98/0.96	*0.91/0.84*
Données	**ASR+SLU**					
	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$	$\bar{r}$ / r	$\bar{p}$ / p	$\overline{F\text{-}m}$ / $F\text{-}m$
Frames	0.82/0.77	0.86/0.78	*0.80/0.77*	0.83/0.78	0.86/0.76	*0.80/0.77*
FE	0.79/0.61	0.86/0.75	*0.78/0.67*	0.80/0.62	0.86/0.73	*0.78/0.67*
FE{Frames}	0.90/0.78	0.97/0.93	*0.92/0.85*	0.90/0.78	0.97/0.93	*0.92/0.85*
Liens	0.77/0.53	0.88/0.68	*0.75/0.59*	0.77/0.53	0.87/0.66	*0.75/0.59*
Liens{Frames}	0.87/0.68	0.97/0.94	*0.90/0.79*	0.87/0.68	0.97/0.94	*0.90/0.79*

Table 12.2 – *Précision (p), rappel (r), F-mesure ($\overline{F\text{-}m}$), précision moyenne ($\bar{p}$), rappel moyen (r) et F-mesure moyenne ($\overline{F\text{-}m}$) sur l'ensemble de test* MEDIA *après application des méthodes de connexion forte et SVM aux fragments sémantiques générés par le système basé sur le modèle DBN compact. Trois type de données ont été considerés : MAN, SLU et ASR+SLU.*

		Connexion forte		Connexion SVM	
Données			**MAN**		
Frames	Total REF	Insertions	Suppressions	Insertion	Suppressions
FACILITY	299	1	59	10	5
HOTEL	604	5	25	97	19
LODGING	859	61	286	148	278
LOCATION	574	0	43	7	34
NUMBER	82	3	2	7	1
RESERVE	478	65	4	76	2
ROOM	641	39	35	39	35

		Connexion forte		Connexion SVM	
Données			**SLU**		
Frames	Total REF	Insertions	Suppressions	Insertion	Suppressions
FACILITY	299	10	78	24	30
HOTEL	604	22	42	95	36
LODGING	859	75	302	184	293
LOCATION	574	21	58	27	53
NUMBER	82	43	20	45	19
RESERVE	478	78	13	86	11
ROOM	641	40	58	40	58

		Connexion forte		Connexion SVM	
Données			**ASR+SLU**		
Frames	Total REF	Insertions	Suppressions	Insertion	Suppressions
FACILITY	299	25	95	39	57
HOTEL	604	46	88	124	85
LODGING	859	84	324	182	318
LOCATION	574	84	111	87	108
NUMBER	82	126	23	143	22
RESERVE	478	91	39	98	39
ROOM	641	96	70	96	70

Table 12.3 – *Insertions et suppressions pour 7 types de frames sur l'ensemble de test* MEDIA *après application des méthodes de connexion forte et par classifieur SVM aux fragments sémantiques générés par le système basé sur le modèle DBN compact. Trois types de données ont été considerés : MAN, SLU et ASR+SLU.*

Le tableau 12.3 présente pour les deux méthodes quelques résultats détaillés en termes d'occurences de frames insérées et supprimées pour sept frames instanciées sur l'ensemble de test MEDIA. Ces frames offrent des situations très différentes. Si au niveau global les systèmes affichent des niveaux d'insertions et d'omissions assez équilibrés (comme le révèlent les valeurs de précision/rappel du tableau 12.2), on con-

state que les situations sont très variables selon les frames. Ainsi LODGING est supprimée massivement, RESERVE est essentiellement insérée et ROOM est autant insérée que supprimée (avec toutefois un taux d'identification très elevé, 90% sur les données manuelles). Une rapide analyse des erreurs nous a aussi permis de constater, sans surprise, que les principales difficultés concernaient les frames les plus "éloignées" des unités de base (i.e. celles qui doivent être inférées et ne peuvent être simplement déduites d'un concept présent dans l'hypothèse). Par exemple, LODGING a un taux d'identification de 60% sur les données manuelles tandis que celui de la frame HOTEL est de 95% .

Il est intéressant de remarquer que, contrairement aux résultats globaux, les différences de comportement entre les deux méthodes de connexion sont très visibles. Pour un même niveau de performance global, les deux méthodes prennent des décisions très différentes. Ce constat tend à renforcer notre hypothèse que la nature des données de test ne permet pas encore de mettre en avant plus clairement l'avantage de la méthode de connexion par classification SVM sur la méthode de connexion forte.

12.4 Conclusion

Les algorithmes de composition sémantiques proposés dans ce travail ont été évalués sur l'ensemble des 3005 dialogues du test MEDIA dans trois conditions de transcription et d'annotation conceptuelle différentes (tâches réalisées manuellement et/ou automatiquement). Les fragments sémantiques associés aux messages de test sont produits par le modèle DBN compact exposé au chapitre 8.

Les résultats obtenus par les deux algorithmes sont similaires. Les deux algorithmes s'avèrent capables de produire des représentations sémantiques complètes des messages utilisateur à partir des fragments sémantiques générés par le décodage séquentiel à base de DBN. Ils confirment la viabilité de l'approche de composition par combinaison d'arbres sémantiques partiels.

La méthode de connection par classifieur SVM doit permettre une détection plus fine des relations entre les composants sémantiques au sein de la phrase. L'absence d'amélioration de performance dans nos expériences est en grande partie imputable à la nature des données utilisées qui comportent encore trop peu de situations suffisamment complexes en terme de représentation sémantique pour ne plus se satisfaire uniquement des relations déduites de l'ontologie.

Conclusion et perspectives

CONCLUSION

La genèse d'un module de compréhension stochastique complet a été présentée au cours de ce document.

Base de connaissances sémantiques

Ce module s'appuie sur une représentation sémantique issue du formalisme FrameNet. Le choix de ce paradigme a été motivé par la capacité des frames et de leurs éléments (FE) à représenter des situations de négociations et à s'adapter aux actions complexes du gestionnaire de dialogue.

La base de connaissances en frames a été conçue pour pouvoir s'adapter au domaine du corpus d'expérimentation MEDIA tout en conservant la plus grande généricité possible. Sa structuration a été pensée pour permettre une représentation sémantique arborescente des messages de l'utilisateur du système de dialogue.

Construite manuellement, la base de connaissances contient 21 frames et 86 FE. Son développement a été conduit avec l'objectif d'atteindre une couverture sémantique maximale du domaine MEDIA : chaque message du locuteur doit pouvoir être représenté par un ensemble de frames et FE combinés au sein d'un arbre sémantique. Pour tenter de mener à bien cet objectif, nous avons procédé par enrichissements successifs de la base. L'annotation manuelle de 225 tours de parole utilisateur avec les frames et FE de la base en construction a été réalisée itérativement et la base complétée de nouveaux frames et FE jusqu'à obtention d'une couverture sémantique complète de tous les messages.

Annotation des données d'apprentissage

L'utilisation de modèles stochastiques dans un cadre applicatif nécessite la connaissance des distributions de probabilités conditionnelles liant les données auxquelles on s'intéresse. Pour notre module de compréhension, ces données sont les mots, les concepts de base, les frames et les FE associés à chaque message du locuteur. Les distributions conditionnelles décrivant le conditionnement mutuel des mots, concepts, frames et FE, peuvent être apprises sur des ensembles de données observées.

Le corpus MEDIA n'étant pas annoté en frames, un système à base de règles a été développé pour permettre l'annotation des données d'apprentissage. Ce système crée tout d'abord frames et FE par reconnaissance de modèles puis associe ces objets sémantiques lors d'une étape d'inférence logique.

Les modèles définissant les frames et leurs FE sont composés d'unités conceptuelles (CU) et lexicales (LU). Ils sont conçus selon l'approche FrameNet. La présence de ces CU et/ou LU dans les tours de parole à annoter déclenche l'instanciation des frames et FE qui leur correspondent. L'étape d'inférence soumet les frames et FE instanciés

par reconnaissance de modèles à un ensemble de règles logiques. Pour chaque tour de parole, frames et FE présents fixent les valeurs de vérité des prédicats de ces règles. Selon ces valeurs de vérité, des frames et des FE sont créés, supprimés, modifiés ou reliés. Le système à base de règles ainsi construit a permis d'annoter automatiquement en frames et FE les 12.000 messages utilisateur du corpus MEDIA non dédiés aux tests.

Pour être employées comme données d'apprentissage des distributions conditionnelles de nos modèles stochastiques, les données annotées automatiquement doivent être évaluées en termes de fiabilité. Cette évaluation a été réalisée sur les 3005 messages utilisateur MEDIA dédiés aux tests. Ces messages, manuellement transcrits et annotés en concepts de base, ont été automatiquement annotés en frames et FE par le système à base de règles puis corrigés manuellement par un linguiste expert.

Les annotations produites par le système à base de règles sur les 3005 messages utilisateur MEDIA dédiés aux tests ont été évaluées par comparaison à cet ensemble de référence. Les résultats obtenus confirment la fiabilité du système à base de règles. Les 12.000 messages utilisateur annotés automatiquement en frames et FE par ce système peuvent dès lors constituer l'ensemble des données d'apprentissage des distributions conditionnelles des modèles stochastiques.

Génération de fragments sémantiques

Les modèles stochastiques proposés, dédiés à la génération de fragments sémantiques, sont des réseaux bayésiens dynamiques (DBN). Ce sont des modèles d'une grande flexibilité permettant de représenter des systèmes stochastiques complexes. Nous avons proposé et évalué trois modèles de structures différentes.

Le modèle compact représente frames et FE à l'aide d'une variable aléatoire unique. La complexité du décodage en est réduite, au prix de l'instauration de liens déterministes entre frames et FE. Dans le modèle factorisé, deux variables aléatoires distinctes sont associées aux frames et FE. Les FE sont conditionnés par les frames. Chaque association frames - FE est évaluée au cours du décodage simultané des frames et des FE. La complexité de ce modèle impose l'emploi d'un algorithme sous-optimal de décodage.

Dans le modèle à deux niveaux, les frames sont décodées en premier lieu puis considérées comme observées lors du décodage des FE. Ce modèle présente l'avantage d'une approche non déterministe des liens frames - FE tout en ayant une complexité inférieure à celle du modèle factorisé. Pour ces trois modèles, les observations sont composées des séquences de mots et concepts du tour de parole utilisateur.

L'apprentissage des paramètres de ces modèles s'appuie sur un algorithme de décomposition des arbres sémantiques en branches conceptuelles. Il est appliqué à l'annotation en frames et FE de la phrase complète. Cette annotation étant structurée en arbre, l'algorithme permet de définir des sous-branches de l'arbre associées à un seul concept. Les sous-branches sont adaptées aux différents modèles DBN. Dans le cas du modèle compact, elles sont considérées comme des classes composées. Dans les modèles factorisés et à deux niveaux, frames et FE sont séparés pour produire deux ensembles de

classes distinctes.

Les expériences ont été menées sur le test MEDIA dans trois conditions différentes, fonctions de la qualité des données initiales. Les performances des trois systèmes sur les données bruitées (sorties des modules d'ASR et de SLU) attestent de leur robustesse à l'incertitude. Ces résultats confirment que les modèles à base de DBN peuvent être utilisés pour générer des fragments sémantiques consistants. Ces fragments sémantiques sont générés séquentiellement par l'observation des séquences successives mots-concept qui composent chaque message. Ils fournissent une représentation sémantique partielle du message global du locuteur.

Composition de fragments sémantiques

Une représentation sémantique complète est obtenue grâce à une étape de recomposition finale. Les relations entre les fragments sémantiques produits par les DBN sont recherchées en considérant un tour de parole complet de l'utilisateur. Deux algorithmes de recomposition ont été évalués. Ils considèrent les fragments issus des DBN et les rassemblent pour rétablir l'arbre sémantique associé à la globalité du message. Ces algorithmes s'appuient sur la représentation arborescente des objets sémantiques frames et FE et de leurs relations. Ils décident des opérations de liaison entre frames et FE et des opérations de regroupement entre frames ou FE.

Le premier algorithme de "connexion forte" est une heuristique déterministe qui vise à obtenir pour chaque message une représentation sémantique compacte dans le cadre autorisé par l'ontologie. Le second algorithme s'appuie sur une méthode de classification à base de séparateurs à vaste marge (SVM) pour décider des opérations à effectuer. Les classifieurs apprennent les opérations réciproques de séparation et de duplication réalisées sur les annotations des messages d'entraînement lors de la construction des données d'apprentissage pour les DBN. Ils sont ensuite interrogés sur la pertinence de chaque opération potentiellement réalisable lors de la phase de recomposition.

Les expériences, menées sur le test MEDIA annoté en fragments sémantiques par le modèle DBN compact, ont montré que les deux algorithmes de recomposition produisent des résultats pertinents.

Atouts du module de compréhension proposé

Ce module repose sur un système ouvert. Il n'utilise pas de grammaire manuelle et peut être adapté au traitement de tout type de dialogue oral. L'apprentissage des modèles utilisés est entièrement automatique (une fois l'annotation d'un corpus de référence réalisée).

Dans le module proposé, la base de connaissances sur laquelle repose la représentation sémantique est facilement modifiable pour s'adapter aux besoins de domaines

variés. L'emploi des structures de frames et FE contribue à la richesse de cette représentation.

Les modèles stochastiques à base de DBN employés pour générer la représentation sémantique des messages n'imposent aucune contrainte quant à la nature des informations observées. Il est aisé d'introduire de nouvelles variables dans ces modèles et donc de prendre en compte, sous réserve de calculabilité, de nouvelles informations pertinentes pour la compréhension (informations liées au contexte de dialogue par exemple).

L'approche par DBN offre un cadre homogène pour l'ensemble des étapes de décodage sequentiel (unités conceptuelles de base puis sous-structures arborescentes). Cette uniformité favorise à terme l'intégration des décodages. Ainsi, sous réserve d'en maîtriser la complexité, on peut envisager de combiner les modèles DBN associés aux différentes étapes au sein d'un modèle DBN global.

Grâce à la collaboration de modèles génératifs DBN et de modèles discriminants SVM, le module de compréhension proposé possède de réelles aptitudes à la généralisation, tout en bénéficiant d'un module de décision efficace. La séparation en deux étapes permet une prise en compte globale des dépendances à long terme tout en restant fortement appuyée sur l'information séquentielle de base (mots-concepts). De plus, la complexité totale du module reste faible par comparaison aux approches traditionnelles et permet d'envisager son utilisation dans le contexte d'un système de dialogue réel. La generalité de l'approche permet aussi l'intégration de sources d'information supplémentaires sans difficulté. Ainsi, la prise en compte d'un contexte plus général de dialogue pourra se faire par adjonction d'une structure de frames inter-tours de parole.

Un module basé intégralement sur des approches probabilistes présente l'intérêt de pouvoir proposer plusieurs hypothèses d'interprétation et de classer ces hypothèses selon leur probabilité (après une reformulation en score de confiance et l'intégration des différents niveaux de décodage). Aussi, bien que non souhaitée, l'utilisation de règles manuelles ad hoc est toujours envisageable en post-traitement, afin de corriger des cas particuliers difficiles à modéliser correctement dans les approches stochastiques.

Les résultats obtenus sur les données expérimentales permettent de considérer que le module de compréhension construit au cours de ce travail est robuste aux données incertaines.

Limites du module de compréhension proposé

La nécessité de disposer d'un corpus pour l'apprentissage des modèles peut constituer un frein important au développement de l'approche pour de nouvelles tâches. Non seulement cette étape peut se révéler coûteuse (temps passé par les annotateurs à traiter un ensemble de données de taille suffisante) mais les étapes préalables peuvent apparaître très complexes. En effet, comme nous l'avons déjà mentionné dans le chapitre 5, définir l'ontologie du domaine à traiter et le manuel de référence indispensable à une production cohérente des données annotées sont des tâches complexes et au

résultat incertain.

L'approche proposée repose en grande partie sur l'utilisation des modèles à base de DBN pour le décodage séquentiel. Or, récemment, de nouvelles approches discriminantes ont permis d'obtenir de très bons résultats sur les tâches traitées. Ainsi, on notera dans (Hahn et al., 2008a,b) les résultats prometteurs de l'application des champs de Markov aléatoires (*Conditional Random Field*, CRF (Lafferty et al., 2001)) au décodage conceptuel. Toutefois, la substitution des CRF aux DBN, ne remettrait pas en cause le principe de l'approche progressive de la composition qui est un des points clés de notre proposition.

PERSPECTIVES

Dans la continuité directe des travaux présentés dans cette thèse, un certain nombre d'évolutions semblent s'imposer et seront réalisées dans un futur proche :

- **listes d'hypothèses scorées (n-best)** : l'objectif est de fournir une liste d'hypothèses d'interprétation au lieu d'une hypothèse unique. A chaque hypothèse est associée une mesure de confiance qui tient compte de l'ensemble des informations utilisées pour l'engendrer (mesure de confiance acoustique, mesure de confiance des concepts de base et mesure de confiance de l'étape de composition).

- **contexte de dialogue** : la prise en compte du contexte de dialogue lors de l'élaboration de l'interprétation sémantique est également une perspective d'évolution du module de compréhension. Des travaux préliminaires utilisant les réseaux markoviens logiques ont déjà été réalisés dans cet objectif (Meurs et al., 2008). L'intégration d'un contexte de dialogue est aisée dans le cadre proposé ici : schématiquement, il suffira d'ajouter la structure arborescente globale contextuelle parmi les hypothèses considérées par l'algorithme de recomposition des sous-arbres (seconde étape). Cette structure sera alors prise en compte naturellement par l'algorithme proposé (quelle que soit la variante, connexion forte ou par classifieur) et les nouvelles branches seront greffées sur l'existant. Il sera toutefois nécessaire d'ajouter des opérations supplémentaires, comme la suppression d'objets sémantiques dans le cas de modalités négatives. La détection fine de ces nouvelles opérations peut présenter quelques difficultés.

- **intégration longitudinale** : l'uniformité des techniques utilisées (en l'occurrence le recours aux DBN) est un point fort qui devrait permettre d'intégrer les étapes de décodage successives (concepts de bases, valeurs normalisées et fragments sémantiques). La réduction de la complexité des algorithmes en jeu est toutefois nécessaire pour atteindre cet objectif.

La portabilité multilingue de notre module pourra prochainement être évaluée grâce aux corpus de dialogue en italien et en polonais développés dans le cadre du projet LUNA. De plus, ces corpus nous permettront d'appliquer notre système à des do-

maines de connaissance potentiellement plus vastes que le domaine du corpus MEDIA.

Un autre résultat important de cette étude est la mise en avant de la difficulté d'établir une ontologie pertinente et de bonne qualité. Même pour une tâche aussi simple que celle de MEDIA, la solution à laquelle nous avons abouti semble largement perfectible. L'examen des ontologies développées dans le cadre du Web Sémantique confirme ce constat (voir par exemple (Aussenac-Gilles et al., 2000)). Ainsi dans le projet ANR PORT-MEDIA [1] un sous-projet spécifique a été planifié, dédié à la proposition et au développement de structures riches pour la représentation des connaissances sémantiques de haut-niveau.

Des fondements solides ont été établis et testés lors de la campagne d'évaluation MEDIA en ce qui concerne les unités conceptuelles de bases. La représentation sémantique de MEDIA doit maintenant être enrichie par une représentation standardisée de haut-niveau permettant de tenir compte de la composition sémantique au sein des tours de parole et au cours du dialogue. Ce travail, dont une grande partie a été déjà réalisée dans le cadre de cette thèse, doit s'inscrire dans un débat plus large, utilisant notamment les compétences de spécialistes en linguistique.

Une perspective importante de ce travail est l'intégration de notre module de compréhension dans un système de dialogue complet fonctionnel. En effet il convient d'évaluer l'intérêt de l'approche en relation avec son but premier : fournir une information riche et structurée au gestionnaire de dialogue ou à tout autre processus de décision intervenant lors de l'interaction orale.

La pertinence de l'information fournie par notre module de compréhension devra être mesurée à l'aune des capacités d'un gestionnaire de dialogue à utiliser correctement et pleinement cette information. En effet, notre module de compréhension s'adresse idéalement à un gestionnaire de dialogue apte non seulement à analyser des données sémantiques complexes mais également à intégrer la mesure de leur incertitude dans son processus de décision.

Le travail de construction d'un tel gestionnaire a déjà été entamé et présenté dans (Pinault et al., 2009) où est décrite la première version d'un module de gestion du dialogue basé sur des modèles probabilistes (modèles de décision markoviens partiellement observables, POMDP). Ce module utilise en entrée les frames sémantiques proposées par le système présenté dans notre travail. Les résultats préliminaires publiées dans (Pinault et al., 2009) montrent la faisabilité de l'approche.

Le système de dialogue complet sera composé du module de reconnaissance de la parole du LIA (Nocera et al., 2002), du module de compréhension stochastique proposé dans ce travail, du gestionnaire de dialogue à base de POMDP et d'un module de synthèse vocale basé sur le logiciel libre Festival (adapté au français grâce au phonétiseur LIA_PHON (Béchet, 2001)). Des expériences impliquant des utilisateurs réels doivent avoir lieu très prochainement.

1. http ://www.port-media.org/

Annexes

Annexe A

Base de connaissances sémantiques

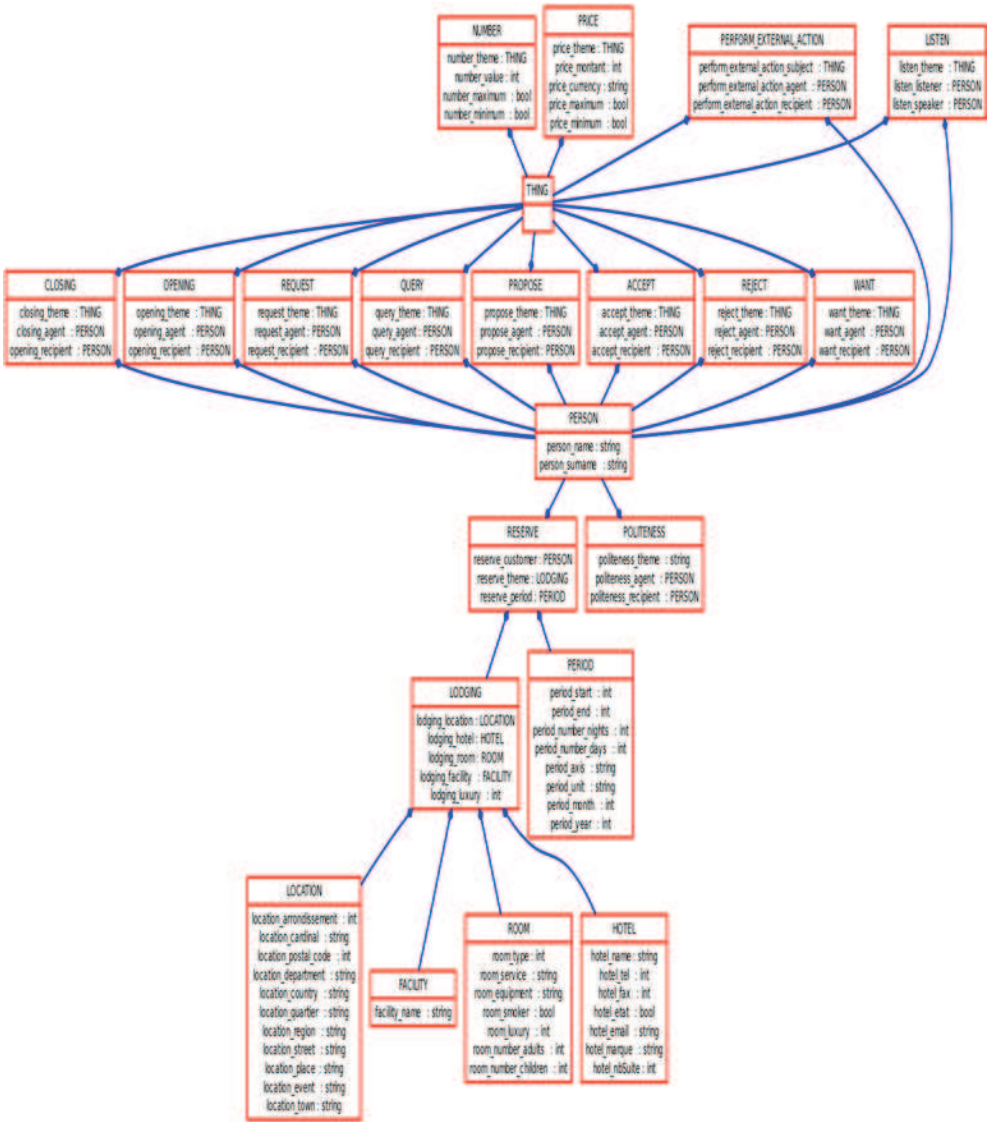


FIGURE *A.1* – *Base de connaissances associée au corpus* MEDIA

Annexe B

Extrait de corpus MEDIA annoté

```
<turn id="209_12_spk start="144" end="155" speaker="spk" audio="08_209.wav">
  <semAnnotation withContext="true" origin="ELDA" manual="true" tool="semantizer">
    <sem id="262" mode="+" concept="reponse" specif="" value="oui">
      <transcription origin="ELDA" manual="true" tool="transcriber">
        <Sync time="144" />
        <Event desc="bb" extent="instantaneous" type="noise" />
        oui oui
      
```

Annexe C

Modèles DBN - format GMTK

Modèle compact

```
GRAPHICAL_MODEL FFE_Decode

#include "commonParams"

frame : 0 {
 variable : startTrans {
   type: discrete observed value 0 cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: startTrans(0)
   using DeterministicCPT("transCopyCPT");
 }

 variable : startFr {
   type: discrete observed value FS_ID cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frM {
   type: discrete hidden cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: startFr(0)
```

```
    using DeterministicCPT("frameCopyCPT");
  }

  variable : cptTrans {
    type: discrete observed TRANS_RNG cardinality 2;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : concept  {
    type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : word {
    type: discrete observed WRD_RNG cardinality WORD_SIZE;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }
}

frame : 1 {
  variable : frTrans {
    type: discrete hidden cardinality 2;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("frTransProbs");
  }

  variable : frM  {
    type: discrete hidden cardinality FRAME_SIZE;
    switchingparents: frTrans(0)
    using mapping("directMappingWithOneParent");
    conditionalparents: frM(-1)
    using DeterministicCPT("frameCopyCPT")
    | frM(-1) using FNGramCPT("frameFNgram");
    weight: scale 1.0 | scale FR_WEIGHT;
  }

  variable : cptTrans {
    type: discrete observed TRANS_RNG cardinality 2;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }
```

```
 variable : concept  {
   type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
   switchingparents: cptTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents:  concept(-1)
   using DeterministicCPT("conceptCopyCPT")
   | frM(0) using FNGramCPT("conceptframeFNgram");
   weight: scale 1.0 | scale CPT_WEIGHT;
 }

 variable : startWord {
   type: discrete observed value CS_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : word {
   type: discrete observed WRD_RNG cardinality WORD_SIZE;
   switchingparents: cptTrans(0) using
   mapping("directMappingWithOneParent");
   conditionalparents: word(-1),concept(0),frM(0) using
   FNGramCPT("wordconceptframeFNgram")
   | startWord(0),concept(0),frM(0)
   using FNGramCPT("wordconceptframeFNgram");
   weight: scale 1.0 | scale WORD_WEIGHT;
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: cptTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: nil using DenseCPT("internal:UnityScore")
   | word(-1),concept(-1),frM(-1)
   using FNGramCPT("wordconceptframeFNgram");
 }
}


frame : 2 {
 variable : lastfrTrans {
   type: discrete observed value 1 cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }
```

```
 variable : frTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: lastfrTrans(0)
   using DeterministicCPT("transCopyCPT");
 }

 variable : lastFr {
   type: discrete observed value FE_ID cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: frM(-1) using FNGramCPT("frameFNgram");
   weight : scale FR_WEIGHT;
 }

 variable : frM {
   type: discrete hidden cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: lastFr(0) using DeterministicCPT("frameCopyCPT");
 }

 variable : endConcept {
   type: discrete observed value SE_ID cardinality CONCEPT_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents:  word(-1),concept(-1),frM(-1)
   using FNGramCPT("wordconceptframeFNgram");
  }
}

chunk 1:1
```

Modèle factorisé

```
GRAPHICAL_MODEL F_FE_Decode

#include "commonParamsFe"

frame : 0 {
 variable : startTrans {
```

```
  type: discrete observed value 0 cardinality 2;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : frTrans {
  type: discrete hidden cardinality 2;
  switchingparents: nil;
  conditionalparents: startTrans(0)
  using DeterministicCPT("transCopyCPT");
}

variable : startFr {
  type: discrete observed value FS_ID cardinality FRAME_SIZE;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : frM {
  type: discrete hidden cardinality FRAME_SIZE;
  switchingparents: nil;
  conditionalparents: startFr(0)
  using DeterministicCPT("frameCopyCPT");
}

variable : feTrans {
  type: discrete hidden cardinality 2;
  switchingparents: nil;
  conditionalparents: startTrans(0)
  using DeterministicCPT("transCopyCPT");
}

variable : startFe {
  type: discrete observed value FELS_ID cardinality FEL_SIZE;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : feM {
  type: discrete hidden cardinality FEL_SIZE;
  switchingparents: nil;
  conditionalparents: startFe(0)
  using DeterministicCPT("feCopyCPT");
}
```

```
 variable : cptTrans {
   type: discrete observed TRANS_RNG cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : concept  {
   type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : word {
   type: discrete observed WRD_RNG cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }
}


frame : 1 {
 variable : frTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("frTransProbs");
 }

 variable : frM  {
   type: discrete hidden cardinality FRAME_SIZE;
   switchingparents: frTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: frM(-1)
   using DeterministicCPT("frameCopyCPT")
   | frM(-1) using FNGramCPT("frameFNgram");
    weight: scale 1.0 | scale FR_WEIGHT;
 }

 variable : feTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: frTrans(0)
   using DeterministicCPT("transCopyCPT");
 }

 variable : feM {
```

```
  type: discrete hidden cardinality FEL_SIZE;
  switchingparents: frTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents: feM(-1)
  using DeterministicCPT("feCopyCPT")
  | frM(0) using FNGramCPT("felmtframeFNgram");
  weight: scale 1.0 | scale FEL_WEIGHT;
}

variable : cptTrans {
  type: discrete observed TRANS_RNG cardinality 2;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : concept  {
  type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
  switchingparents: cptTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents:  concept(-1)
  using DeterministicCPT("conceptCopyCPT")
  | frM(0),feM(0) using FNGramCPT("conceptframefelmtFNgram");
  weight: scale 1.0 | scale CPT_WEIGHT;
}

variable : startWord {
  type: discrete observed value CS_ID cardinality WORD_SIZE;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : word {
  type: discrete observed WRD_RNG cardinality WORD_SIZE;
  switchingparents: cptTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents: word(-1),concept(0),frM(0),feM(0)
  using FNGramCPT("wordconceptframefelmtFNgram")
  | startWord(0),concept(0),frM(0),feM(0)
  using FNGramCPT("wordconceptframefelmtFNgram");
  weight: scale 1.0 | scale WORD_WEIGHT;
}

variable : endWord {
  type: discrete observed value CE_ID cardinality WORD_SIZE;
  switchingparents: cptTrans(0)
```

```
   using mapping("directMappingWithOneParent");
   conditionalparents: nil using DenseCPT("internal:UnityScore")
   | word(-1),concept(-1),frM(-1),feM(-1)
   using FNGramCPT("wordconceptframefelmtFNgram");
 }
}


frame : 2 {
 variable : lastfrTrans {
   type: discrete observed value 1 cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: lastfrTrans(0)
   using DeterministicCPT("transCopyCPT");
 }

 variable : lastFr {
   type: discrete observed value FE_ID cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: frM(-1) using FNGramCPT("frameFNgram");
   weight : scale FR_WEIGHT;
 }

 variable : frM {
   type: discrete hidden cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: lastFr(0) using DeterministicCPT("frameCopyCPT");
 }

 variable : lastfeTrans {
   type: discrete observed value 1 cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : feTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents:  lastfeTrans(0)
```

```
   using DeterministicCPT("transCopyCPT");
 }

 variable : lastFe {
   type: discrete observed value FELE_ID cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: frM(0) using FNGramCPT("felmtframeFNgram");
   weight : scale FEL_WEIGHT;
 }

 variable : feM {
   type: discrete hidden cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: lastFe(0) using DeterministicCPT("feCopyCPT");
 }

 variable : endConcept {
   type: discrete observed value SE_ID cardinality CONCEPT_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents:  word(-1),concept(-1),frM(-1),feM(-1)
   using FNGramCPT("wordconceptframefelmtFNgram");
 }
}

chunk 1:1
```

Modèle à deux niveaux

Premier niveau

```
GRAPHICAL_MODEL Frame_Decode

#include "commonParams"

frame : 0 {
 variable : startTrans {
   type: discrete observed value 0 cardinality 2;
   switchingparents: nil;
```

```
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : frTrans {
    type: discrete hidden cardinality 2;
    switchingparents: nil;
    conditionalparents: startTrans(0)
    using DeterministicCPT("transCopyCPT");
  }

  variable : startFr {
    type: discrete observed value FS_ID cardinality FRAME_SIZE;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : frM {
    type: discrete hidden cardinality FRAME_SIZE;
    switchingparents: nil;
    conditionalparents: startFr(0)
    using  DeterministicCPT("frameCopyCPT");
  }

  variable : cptTrans {
    type: discrete observed TRANS_RNG cardinality 2;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : concept  {
    type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : word {
    type: discrete observed WRD_RNG cardinality WORD_SIZE;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }
}


frame : 1 {
  variable : frTrans {
```

```
  type: discrete hidden cardinality 2;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("frTransProbs");
}

variable : frM  {
  type: discrete hidden cardinality FRAME_SIZE;
  switchingparents: frTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents: frM(-1)
  using DeterministicCPT("frameCopyCPT")
  | frM(-1) using FNGramCPT("frameFNgram");
  weight: scale 1.0 | scale FR_WEIGHT;
}

variable : cptTrans {
  type: discrete observed TRANS_RNG cardinality 2;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : concept  {
  type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
  switchingparents: cptTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents: concept(-1)
  using DeterministicCPT("conceptCopyCPT")
  | frM(0) using FNGramCPT("conceptframeFNgram");
  weight: scale 1.0 | scale CPT_WEIGHT;
}

variable : startWord {
  type: discrete observed value CS_ID cardinality WORD_SIZE;
  switchingparents: nil;
  conditionalparents: nil using DenseCPT("internal:UnityScore");
}

variable : word {
  type: discrete observed WRD_RNG cardinality WORD_SIZE;
  switchingparents: cptTrans(0)
  using mapping("directMappingWithOneParent");
  conditionalparents: word(-1),concept(0),frM(0)
  using FNGramCPT("wordconceptframeFNgram")
  | startWord(0),concept(0),frM(0)
  using FNGramCPT("wordconceptframeFNgram");
```

```
    weight: scale 1.0 | scale WORD_WEIGHT;
  }

  variable : endWord {
    type: discrete observed value CE_ID cardinality WORD_SIZE;
    switchingparents: cptTrans(0)
    using mapping("directMappingWithOneParent");
    conditionalparents: nil using DenseCPT("internal:UnityScore")
    | word(-1), concept(-1), frM(-1)
    using FNGramCPT("wordconceptframeFNgram");
  }
}


frame : 2 {
  variable : lastTrans {
    type: discrete observed value 1 cardinality 2;
    switchingparents: nil;
    conditionalparents: nil using DenseCPT("internal:UnityScore");
  }

  variable : frTrans {
    type: discrete hidden cardinality 2;
    switchingparents: nil;
    conditionalparents: lastTrans(0)
    using  DeterministicCPT("transCopyCPT");
  }

  variable : lastFr {
    type: discrete observed value FE_ID cardinality FRAME_SIZE;
    switchingparents: nil;
    conditionalparents: frM(-1) using FNGramCPT("frameFNgram");
    weight : scale FR_WEIGHT;
  }

  variable : frM {
    type: discrete hidden cardinality FRAME_SIZE;
    switchingparents: nil;
    conditionalparents: lastFr(0)
    using DeterministicCPT("frameCopyCPT");
  }

  variable : endConcept {
    type: discrete observed value SE_ID cardinality CONCEPT_SIZE;
    switchingparents: nil;
```

```
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents:  word(-1), concept(-1), frM(-1)
   using FNGramCPT("wordconceptframeFNgram");
 }
}

chunk 1:1
```

Second niveau

```
GRAPHICAL_MODEL Concept_Decode

#include "commonParamsFe"

frame : 0 {
 variable : frTrans {
   type: discrete observed FRTRANS_RNG cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frM {
   type: discrete observed FRM_RNG cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : startFe {
   type: discrete observed value FELS_ID cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : feM {
   type: discrete hidden cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: startFe(0)
   using DeterministicCPT("feCopyCPT");
 }
```

```
 variable : cptTrans {
   type: discrete observed TRANS_RNG cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : concept  {
   type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : word {
   type: discrete observed WRD_RNG cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }
}


frame : 1 {
 variable : frTrans {
   type: discrete observed FRTRANS_RNG cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frM  {
   type: discrete observed FRM_RNG cardinality FRAME_SIZE;
   switchingparents: frTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: frM(-1)
   using DeterministicCPT("frameCopyCPT")
   | frM(-1) using FNGramCPT("frameFNgram");
   weight: scale 1.0 | scale FR_WEIGHT;
 }

 variable : feM {
   type: discrete hidden cardinality FEL_SIZE;
   switchingparents: frTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: feM(-1)
   using DeterministicCPT("feCopyCPT")
   | frM(0) using FNGramCPT("felmtframeFNgram");
   weight: scale 1.0 | scale FEL_WEIGHT;
```

```
 }

 variable : cptTrans {
   type: discrete observed TRANS_RNG cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : concept  {
   type: discrete observed CPT_RNG cardinality CONCEPT_SIZE;
   switchingparents: cptTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents:  concept(-1)
   using DeterministicCPT("conceptCopyCPT")
   | frM(0),feM(0) using FNGramCPT("conceptframefelmtFNgram");
   weight: scale 1.0 | scale CPT_WEIGHT;
 }

 variable : startWord {
   type: discrete observed value CS_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : word {
   type: discrete observed WRD_RNG cardinality WORD_SIZE;
   switchingparents: cptTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: word(-1),concept(0),frM(0),feM(0)
   using FNGramCPT("wordconceptframefelmtFNgram")
   | startWord(0),concept(0),frM(0),feM(0)
   using FNGramCPT("wordconceptframefelmtFNgram");
   weight: scale 1.0 | scale WORD_WEIGHT;
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: cptTrans(0)
   using mapping("directMappingWithOneParent");
   conditionalparents: nil using DenseCPT("internal:UnityScore")
   | word(-1),concept(-1),frM(-1),feM(-1)
   using FNGramCPT("wordconceptframefelmtFNgram");
 }
}
```

```
frame : 2 {
 variable : lastfrTrans {
   type: discrete observed value 1 cardinality 2;
   switchingparents: nil;
   conditionalparents: nil using DenseCPT("internal:UnityScore");
 }

 variable : frTrans {
   type: discrete hidden cardinality 2;
   switchingparents: nil;
   conditionalparents: lastfrTrans(0)
   using DeterministicCPT("transCopyCPT");
 }

 variable : lastFr {
   type: discrete observed value FE_ID cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: frM(-1) using FNGramCPT("frameFNgram");
   weight : scale FR_WEIGHT;
 }

 variable : frM {
   type: discrete hidden cardinality FRAME_SIZE;
   switchingparents: nil;
   conditionalparents: lastFr(0)
   using DeterministicCPT("frameCopyCPT");
 }

 variable : lastFe {
   type: discrete observed value FELE_ID cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: frM(0) using FNGramCPT("felmtframeFNgram");
   weight : scale FEL_WEIGHT;
 }

 variable : feM {
   type: discrete hidden cardinality FEL_SIZE;
   switchingparents: nil;
   conditionalparents: lastFe(0) using DeterministicCPT("feCopyCPT");
 }

 variable : endConcept {
   type: discrete observed value SE_ID cardinality CONCEPT_SIZE;
   switchingparents: nil;
```

```
   conditionalparents: frM(0),feM(0)
   using FNGramCPT("conceptframefelmtFNgram");
 }

 variable : endWord {
   type: discrete observed value CE_ID cardinality WORD_SIZE;
   switchingparents: nil;
   conditionalparents:  word(-1),concept(-1),frM(-1),feM(-1)
   using FNGramCPT("wordconceptframefelmtFNgram");
 }
}

chunk 1:1
```

Annexe D

Méthode de Lagrange

Application de la méthode des multiplicateurs de Lagrange pour transformer le problème d'optimisation sous contraintes présenté dans la sesction 10.3 en un problème d'optimisation sans contrainte ayant la même solution.

Le lagrangien $L(\vec{w}, w_0, \vec{\alpha})$ associé au problème est la somme de la fonction objectif et de l'opposé d'une combinaison linéaire à coefficients positifs des contraintes. Les coefficients α_i de cette combinaison linéaire sont les *multiplicateurs de Lagrange* appelés également *variables duales*.
On a :

$$L(\vec{w}, w_0, \vec{\alpha}) = \frac{1}{2}\|\vec{w}\|^2 - \sum_{i=1}^{n} \alpha_i (u_i(\vec{w}.\vec{x_i} + w_0) - 1)$$

Le problème primal et sa forme duale ont pour solutions communes les points selle du lagrangien que l'on doit minimiser par rapport aux variables primaires $\vec{w}$ et w_0 et maximiser par rapport aux variables duales α_i.

Toute solution optimale $(\vec{w}^*, w_0^*)$ du problème primal vérifie les conditions de Karush-Kuhn-Tucker(Kuhn et Tucker, 1951) (KKT), ce qui implique notamment que $\forall i \in [\![1, n]\!]$, le multiplicateur de Lagrange α_i associé à la contrainte i est nul lorsque cette contrainte n'est pas saturée.

En effet, les dérivées partielles en $\vec{w}$ et en w_0 du lagrangien $\frac{\partial}{\partial \vec{w}} L(\vec{w}, w_0, \vec{\alpha}) = 0$ et $\frac{\partial}{\partial w_0} L(\vec{w}, w_0, \vec{\alpha}) = 0$ doivent être nulles, ce qui donne $\sum_{i=1}^{n} \alpha_i u_i = 0$ et $\vec{w} = \sum_{i=1}^{n} \alpha_i u_i \vec{x_i}$.

En forme duale, le problème est donc de trouver les multiplicateurs de Lagrange $\alpha_i \geq 0$

tels que :

$$\begin{cases} Max_{\vec{\alpha}} \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j u_i u_j (\vec{x_i}.\vec{x_j}) \right) \\ \alpha_i \geq 0 \forall i \in [\![1, n]\!] \\ \sum_{i=1}^{n} \alpha_i u_i = 0 \end{cases}$$

La dernière condition (KKT) s'écrit $\alpha_i(u_i(\vec{w}.\vec{x_i} + w_0) - 1) = 0 \forall i \in [\![1, n]\!]$ donc $\forall \vec{x_i} \in X$, soit $\alpha_i = 0$, soit $u_i(\vec{w}.\vec{x_i} + w_0) - 1 = 0$ $(\star)$.

L'équation de l'hyperplan séparateur optimal est donc :
$y = \vec{w^*}.\vec{x} + w_0^* = \sum_{i \in M} \alpha_i^* u_i (\vec{x}.\vec{x_i}) + w_0 *$
où les $\alpha_i^*, \quad i \in M$ sont les multiplicateurs de Lagrange non nuls (indicés dans $M \subset [\![1, n]\!]$).

On appelle *vecteur support* une donnée dont le multiplicateur de Lagrange associé est non nul. D'après $(\star)$, les vecteurs supports appartiennent donc aux hyperplans H^- et H^+.

En conséquence, **seules les données correspondant aux vecteurs supports sont utiles à l'apprentissage**.

Pour toute nouvelle donnée $\vec{x}$, le signe de $\sum_{i \in M} \alpha_i^* u_i (\vec{x}.\vec{x_i}) + w_0 *$ décidera de sa classe.

Annexe E

Publications personnelles

Revue internationale

Mohamed DIDI BIHA and Marie-Jean MEURS
Polyhedral approach for the vertex separator problem
En révision

Revue nationale

Marie-Jean MEURS, Fabrice LEFÈVRE, Renato DE MORI
Approche bayésienne de la composition sémantique dans les systèmes de dialogue oral
ISI, 2010
A paraître

Conférences internationales

Marie-Jean MEURS, Fabrice LEFÈVRE, Renato DE MORI
Learning Bayesian Networks for Semantic Frame Composition in a Spoken Dialog System
NAACL-HLT, 2009, Boulder, CO, USA

Marie-Jean MEURS, Fabrice LEFÈVRE, Renato DE MORI
Spoken Language Interpretation : On the Use of Dynamic Bayesian Networks for Semantic Composition
IEEE ICASSP, 2009, Taipei, Taiwan

Marie-Jean MEURS, Fabrice LEFÈVRE, Renato DE MORI
A Bayesian approach to semantic composition for spoken language interpretation
InterSpeech, 2008, Brisbane, Australie

Marie-Jean MEURS, Frédéric DUVERT, Frédéric BÉCHET, Fabrice LEFÈVRE, Renato DE MORI
Semantic Frame Annotation on the French MEDIA corpus
LREC, 2008, Marrakech, Maroc

Marie-Jean MEURS, Frédéric DUVERT, Fabrice LEFÈVRE, Renato DE MORI
Markov Logic Networks for Spoken Language Interpretation
IIS, 2008, Zakopane, Pologne

Frédéric DUVERT, Marie-Jean MEURS, Christophe SERVAN, Frédéric BÉCHET, Fabrice LEFÈVRE, Renato DE MORI
Semantic composition process in a spoken understanding system
IIS, 2008, Zakopane, Pologne

Frédéric DUVERT, Marie-Jean MEURS, Christophe SERVAN, Frédéric BÉCHET, Fabrice LEFÈVRE, Renato DE MORI
Semantic composition process in a speech understanding system
IEEE ICASSP, 2008, Las Vegas, USA

Marie-Jean MEURS, Eric SANJUAN
Combining Optimal and Atomic Decomposition of Terminology Association graphs
MLG, 2008, Helsinki, Finlande

Mohamed DIDI BIHA, Bangaly KABA, Marie-Jean MEURS, Eric SANJUAN, *Graph decomposition approaches for terminology graphs*
MICAI, 2007, Aguascalientes, Mexique

Conférences nationales

Marie-Jean MEURS, Fabrice LEFÈVRE, Renato DE MORI
Interprétation du dialogue oral : pour une approche bayésienne de la composition sémantique.
MajecSTIC, 2008, Marseille, France
Best paper award

Marie-Jean MEURS, Frédéric DUVERT, Frédéric BÉCHET, Fabrice LEFÈVRE, Renato DE MORI
Annotation en Frames Sémantiques du corpus de dialogue MEDIA
TALN, 2008, Avignon, France

Frédéric DUVERT, Marie-Jean MEURS, Christophe SERVAN, Frédéric BÉCHET, Fabrice LEFÈVRE, Renato DE MORI
Composition sémantique pour la compréhension de la parole dans le cadre de dialogue
JEP, 2008, Avignon, France

Communications

Marie-Jean MEURS, Mohamed DIDI BIHA
Polyhedral Approach for the Vertex Separator Problem
The International Conference of NonConvex Programming, 2007, Rouen, France

Marie-Jean MEURS, Mohamed DIDI BIHA
Approche polyédrale pour le problème du séparateur
FRANCORO V - ROADEF, 2007, Grenoble, France

Mémoire

Marie-Jean MEURS
Approche polyédrale pour le problème du séparateur
Mémoire de Master Recherche, LIA, 2006

Annexe F

Liste des acronymes

ASR	*Automatic Speech Recognition* Reconnaissance automatique de la parole
CRF	*Conditional Random Field* Champ de Markov aléatoire
CU	*Conceptual Unit* Unité conceptuelle
DBN	*Dynamic Bayesian Network* Réseau bayésien dynamique
DM	*Dialog Manager* Gestionnaire de dialogue
F	Frame
FE	Frame element
FLM	*Factored Language Model* Modèle de langage factorisé
FSM	*Finite State Machine* Machine à Etats Finis
GPB	*Generalized Parallel Backoff* Repli parallèle généralisé
GRT	Grammaire à base de réseaux de transitions
GRTA	Grammaire à base de réseaux de transitions augmentés

HMM	*Hidden Markov Model* Modèle de Markov caché
IAg	*Inter-annotator Agreement* Accord inter-annotateur
LU	*Lexical Unit* Unité lexicale
MLN	*Markov Logic Network* Réseau markovien logique
NLG	*Natural Language Generation* Génération du langage naturel
POMDP	*Partially Observable Markov Decision Process* Modèle de décision markovien partiellement observable
SLU	*Spoken Language Understanding* Compréhension automatique de la parole
SVM	*Support Vector Machine* Séparateur à vaste marge
TALN	Traitement Automatique de la Langue Naturelle
TTS	*Text-To-Speech synthesis* Synthèse de la parole
WoZ	*Wizard of Oz* Magicien d'Oz

Liste des illustrations

Liste des tableaux

Bibliographie

(Allen, 1988) J. Allen, 1988. *Natural language understanding*. Redwood City, CA, USA : Benjamin-Cummings Publishing Co., Inc. 28

(Aussenac-Gilles et al., 2000) N. Aussenac-Gilles, B. Biebow, et S. Szulman, 2000. Revisiting ontology design : A methodology based on corpus analysis. Dans les actes de *EKAW '00 : Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management*, London, UK, 172–188. Springer-Verlag. 154

(Baker et al., 1998) C. Baker, C. Fillmore, et J. Lowe, 1998. The berkeley framenet project. Dans les actes de *COLING-ACL*, Montreal, Canada. 52

(Barrault et al., 2008) L. Barrault, C. Servan, D. Matrouf, G. Linarès, et R. de Mori, 2008. Frame-based acoustic feature integration for speech understanding. Dans les actes de *IEEE ICASSP*, Las Vegas. 112, 142

(Bellegarda, 2007) J. Bellegarda, 2007. *Latent Semantic Mapping - Principes & Application*. Morgan and Claypool Publishers. 125

(Bellegarda et Silverman, 2003) J. Bellegarda et K. Silverman, 2003. Natural language spoken interface using data-driven semantic inference. *IEEE Transactions on Speech and Audio Processing 11*(3), 267–277. 18

(Bennacef et al., 1994) S. Bennacef, H. Bonneau-Maynard, J.-L. Gauvain, L. Lamel, et W. Minker, 1994. A spoken language system for information retrieval. Dans les actes de *ICSLP*, Yokohama, Japan. 32

(Bennacef et al., 1996) S. Bennacef, L. Devillers, S. Rosset, et L. Lamel, 1996. Dialog in the railtel telephone-based system. Dans les actes de *ICSLP*, Philadelphia. 32

(Bille, 2005) P. Bille, 2005. A survey on tree edit distance and related problems. *Theorical Computional Science 337*, 217–239. 124

(Bilmes, 2000) J. Bilmes, 2000. Dynamic bayesian multinets. Dans les actes de *UAI '00 : Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, 38–45. Morgan Kaufmann Publishers Inc. 96

(Bilmes et Kirchhoff, 2003) J. Bilmes et K. Kirchhoff, 2003. Factored language models and generalized parallel backoff. Dans les actes de *HLT-NAACL*. 97

(Bilmes et Zweig, 2002) J. Bilmes et G. Zweig, 2002. The graphical models toolkit : An open source software system for speech and time-series processing. Dans les actes de *IEEE ICASSP*, Orlando, Florida. 113

(Bod, 2000) R. Bod, 2000. Combining semantic and syntactic structure for language modeling. Dans les actes de *ICSLP*. 33

(Bonneau-Maynard et Lefèvre, 2005) H. Bonneau-Maynard et F. Lefèvre, 2005. A 2+1-level stochastic understanding model. Dans les actes de *ASRU*. 45

(Bonneau-Maynard et al., 2005) H. Bonneau-Maynard, S. Rosset, C. Ayache, A. Kuhn, et D. Mostefa, 2005. Semantic annotation of the french media dialog corpus. Dans les actes de *Eurospeech*, Lisboa, Portugal. 59, 62

(Bouckaert et al., 2008) R. R. Bouckaert, E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, et D. Scuse, 2008. Weka manual for version 3-6-0. User manual, The University of Waikato, New Zealand. 143

(Brachman, 1979) R. J. Brachman, 1979. On the epistemological status of semantic networks. Dans N. V. Findler (Ed.), *Associative Networks : Representation and Use of Knowledge by Computers*, 3–50. Orlando : Academic Press. 50

(Brachman et Schmolze, 1985) R. J. Brachman et J. G. Schmolze, 1985. an overview of the kl-one knowledge representation system. *Cognitive Science : A Multidisciplinary Journal 9 :2*, 171–216. 50

(Bruce, 1975) B. Bruce, 1975. Case systems for natural language. *Artificial Intelligence 6*. 32

(Béchet, 2001) F. Béchet, 2001. Lia_phon : Un système complet de phonétisation de textes. *TAL 42/1*, 47–67. 154

(Candillier, 2006) L. Candillier, 2006. *Contextualisation, visualisation et évaluation en apprentissage non supervisé*. Thèse de Doctorat, Université Charles de Gaulle Lille 3. 124

(Candillier et al., 2007) L. Candillier, L. Denoyer, P. Gallinari, M.-C. Rousset, A. Termier, et A.-M. Vercoustre, 2007. Mining xml documents. Dans M. T. P. Poncelet, F. Masseglia (Ed.), *Data Mining Patterns : New Methods and Applications*, pp. 198–219. Information Science Reference. 124

(Carletta, 1996) J. Carletta, 1996. Assessing agreement on classification tasks : the kappa statistic. *Computional Linguistics 22*(2), 249–254. 62

(Chang et Lin, 2001) C.-C. Chang et C.-J. Lin, 2001. *LIBSVM : a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 143

(Charniak, 1991) E. Charniak, 1991. Bayesian networks without tears : making bayesian networks more accessible to the probabilistically unsophisticated. *AI Mag. 12*(4), 50–63. 89

(Chelba, 1997) C. Chelba, 1997. A structured language model. Dans les actes de *EACL*. 33

(Chomsky, 1957) N. Chomsky, 1957. *Syntactic structures*. Mouton, The Hague. 25

(Chomsky, 1964) N. Chomsky, 1964. A transformational approach to syntax. *The Structure of Language*. 31

(Chomsky et Schützenberger, 1963) N. Chomsky et M. Schützenberger, 1963. The algebraic theory of context-free languages. *Computer programming and formal systems*. 30

(Colmerauer et Roussel, 1996) A. Colmerauer et P. Roussel, 1996. The birth of prolog. *History of programming languages—II*, 331–367. 80

(Corazza et al., 1994) A. Corazza, R. D. Mori, R. Gretter, et G. Satta, 1994. Optimal probabilistic evaluation functions for search controlled by stochastic context-free grammars. *IEEE Transactions on Pattern Analysis and Machine Intelligence 16*(10), 1018–1027. 33

(Cover, 1965) T. Cover, 1965. Geometrical and statistical properties of systems of linear inequalities with application in pattern recognition. *IEEE Transactions on Electronic Computers 14*, 326–334. 125

(Crabbe et al., 2003) B. Crabbe, B. Gaiffe, et A. Roussanaly, 2003. Une plateforme de conception et d'exploitation de grammaire d'arbres adjoints lexicalises. Dans les actes de *TALN*. 33

(Dahl et al., 1994) D. Dahl, M. Bates, M. Brown, W. Fisher, K. Hunicke-Smith, D. Pallett, C. Pao, A. Rudnicky, et E. Shriberg, 1994. Expanding the scope of the atis task : The atis-3 corpus. Dans les actes de *ARPA Spoken Language Systems Technology Workshop*, 3–8. 40

(Damnati et al., 2007) G. Damnati, F. Béchet, et R. de Mori, 2007. Spoken language understanding strategies on the france telecom 3000 voice agency corpus. Dans les actes de *IEEE ICASSP*. 15

(De Mori, 1998) R. De Mori, 1998. *Spoken Dialogues with Computers*. Academic Press. 40

(Dean et Kanazawa, 1988) T. Dean et K. Kanazawa, 1988. Probabilistic temporal reasoning. Dans les actes de *AAAI*. 96

(Dean et Kanazawa, 1989) T. Dean et K. Kanazawa, 1989. A model for reasoning about persistence and causation. *Artificial Intelligence 93*(1-2), 1–27. 88, 96

(Denis et al., 2006) A. Denis, M. Quignard, et G. Pittel, 2006. A deep-parsing approach to natural language understanding in dialogue system : Results of a corpus-based evaluation. Dans les actes de *LREC*. 33

(Devillers et al., 2002) L. Devillers, H. Maynard, et P. Paroubek, 2002. Méthodologie d'évaluation des systèmes de dialogue parlé : réflexions et expériences autour de la compréhension. *TAL 43*(2), 155–184. 58

(EL-Manzalawy et Honavar, 2005) Y. EL-Manzalawy et V. Honavar, 2005. *WLSVM : Integrating LibSVM into Weka Environment*. Software available at `http://www.cs.iastate.edu/~yasser/wlsvm`. 143

(Fillmore, 1968) C. J. Fillmore, 1968. The case for case. *Universals in linguistic theory*. 32, 51, 52

(Fillmore, 1982) C. J. Fillmore, 1982. *Frame Semantics*. Linguistics in the Morning Calm, Seoul. 51, 52

(Fillmore, 1985) C. J. Fillmore, 1985. Frames and the semantics of understanding. *Quaderni di Semantica VI*(2), 222–254. 32, 51, 55

(Fillmore et al., 2003) C. J. Fillmore, C. R. Johnson, et M. R. Petruck, 2003. Background to framenet. *International Journal of Lexicography 16.3*, 235–250. 52, 70, 76

(Geiger et Heckerman, 1996) D. Geiger et D. Heckerman, 1996. Knowledge representation and inference in similarity networks and bayesian multinets. *Artificial Intelligence 82*, 45–74. 96

(Gildea et Jurafsky, 2002) D. Gildea et D. Jurafsky, 2002. Automatic labeling of semantic roles. *Computational Linguistics 28*(3), 245–288. 33, 34

(Gorin et al., 1997) A. L. Gorin, G. Riccardi, et J. H. Wright, 1997. How may i help you ? *Speech Communication 23*(1-2), 113–127. 15

(Hahn et al., 2008a) S. Hahn, P. Lehnen, et H. Ney, 2008a. System combination for spoken language understanding. Dans les actes de *Interspeech*, Brisbane, Australia. 153

(Hahn et al., 2008b) S. Hahn, P. Lehnen, C. Raymond, et H. Ney, 2008b. A comparison of various methods for concept tagging for spoken language understanding. Dans les actes de *Sixth Int. Conf. on Language Resources and Evaluation (LREC)*, Marrakech, Morocco. 153

(Hayes et al., 1986) P. Hayes, A. Hauptman, J. Carbonnell, et M. Tomita, 1986. Parsing spoken language, a semantic caseframe approach. Dans les actes de *COLING-86*. 32

(He et Young, 2003) Y. He et S. Young, 2003. Hidden vector state model for hierarchical semantic parsing. Dans les actes de *IEEE ICASSP*, Hong Kong. 43

(He et Young, 2006) Y. He et S. Young, 2006. Spoken language understanding using the hidden vector state model. *Speech Communication 48 (3-4)*(3-4), 262–275. 19, 43

(Hockey et Miller, 2007) B. A. Hockey et D. P. Miller, 2007. A demonstration of a conversationally guided smart wheelchair. Dans les actes de *Assets '07 : Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility*. 16

(Jackendoff, 1990) R. Jackendoff, 1990. *Semantic Structures*. 50

(Jelinek, 1976) F. Jelinek, 1976. Continuous speech recognition by statistical methods. *IEEE 64*(4), 532–556. 38

(Jordan, 1998) M. I. Jordan, 1998. *Learning in graphical models*. Kluwer Academic Publishers. 88

(Jousse, 2007) F. Jousse, 2007. *Transformations d'Arbres XML avec des Modèles Probabilistes pour l'Annotation*. Thèse de Doctorat, Université Charles de Gaulle - Lille III. 124

(Jurcicek et al., 2008) F. Jurcicek, J. Svec, et L. Müller, 2008. Extension of hvs semantic parser by allowing left-right branching. Dans les actes de *International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, USA. 44

(Kingsbury et Palmer, 2003) P. Kingsbury et M. Palmer, 2003. Propbank : the next level of treebank. Dans les actes de *TreeBank and Lexical Theories*. 52

(Kirchhoff et al., 2008) K. Kirchhoff, J. Bilmes, et K. Duh, 2008. Factored language models tutorial. Rapport technique, Dept of EE, University of Washington. 97

(Klatt, 1977) D. H. Klatt, 1977. Review of the arpa speech understanding project. *Acoustical Society of America Journal 62*, 1345–1366. 31

(Kneser et Ney, 1995) R. Kneser et H. Ney, 1995. Improved backing-off for m-gram language modeling. Dans les actes de *IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 181–184. 105

(Kuhn et Tucker, 1951) H. W. Kuhn et A. Tucker, 1951. Nonlinear programming. Dans J. Neyman (Ed.), *Second Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, Calif, 481–492. University of California Press. 179

(Kuhn et De Mori, 1995) R. Kuhn et R. De Mori, 1995. The application of semantic classification trees to natural language understanding. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 17*(5), 449–460. 40

(Lafferty et al., 2001) J. Lafferty, A. McCallum, et F. Pereira, 2001. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. Dans les actes de *ICML*, 282–289. Morgan Kaufmann, San Francisco, CA. 153

(Lamel et al., 1999) L. Lamel, S. Rosset, J.-L. Gauvain, et S. Bennacef, 1999. The LIMSI ARISE system for train travel information. Dans les actes de *IEEE ICASSP*, Phoenix. 32

(Lamel et al., 2000) L. Lamel, S. Rosset, J.-L. Gauvain, S. Bennacef, M. Garnier-Rizet, et B. Prouts, 2000. The limsi arise system. *Speech Communication 31(4)*, 339–354. 15, 42

(Laroche et al., 2009) R. Laroche, G. Putois, P. Bretier, et B. Bouchon-Meunier, 2009. Hybridisation of expertise and reinforcement learning in dialogue systems. Dans les actes de *Interspeech*, Brighton, UK. 18

(Lefèvre, 2006) F. Lefèvre, 2006. A dbn-based multi-level stochastic spoken language understanding system. Dans les actes de *IEEE/ACL Workshop on Spoken Language Technology*, Aruba. 45, 112, 142

(Lefèvre, 2007) F. Lefèvre, 2007. Dynamic bayesian networks and discriminative classifiers for multi-stage semantic interpretation. Dans les actes de *IEEE ICASSP*, Hawaii, USA. 19, 45, 96

(Lefèvre et de Mori, 2007) F. Lefèvre et R. de Mori, 2007. Unsupervised state clustering for stochastic dialog management. Dans les actes de *IEEE Automatic Speech Recognition and Understanding Workshop*, Kyoto, Japan. 18

(Levin et Pieraccini, 1995) E. Levin et R. Pieraccini, 1995. Concept-based spontaneous speech understanding system. Dans les actes de *EUROSPEECH*. 19

(Levin et al., 1997) E. Levin, R. Pieraccini, et W. Eckert, 1997. Learning dialogue strategies within the markov decision process framework. Dans les actes de *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 72–79. 18

(Litman et Silliman, 2004) D. J. Litman et S. Silliman, 2004. Itspoke : An intelligent tutoring spoken dialogue system. Dans les actes de *HLT-NAACL*. 15

(Marconi, 1995) D. Marconi, 1995. *Filosofia del linguaggio (La philosophie du langage au XXeme siecle)*. LA FILOSOFIA. 1997 pour la traduction française. 23

(Marcus et al., 1994) M. Marcus, B. Santorini, et M. Marcinkiewicz, 1994. Building a large annotated corpus of english : The penn treebank. *Computational Linguistics 19(2)*, 313–330. 52

(Matrouf et al., 1990) A. Matrouf, J.-L. Gauvain, F. Néel, et J. Mariani, 1990. An oral task-oriented dialog for air-traffic controller training. Dans les actes de *SPIE1293, Applications of Artificial Intelligence,VIII*. 32

(Maynard et Lefèvre, 2002) H. Maynard et F. Lefèvre, 2002. Apprentissage d'un module stochastique de compréhension de la parole. Dans les actes de *Journées d'Etudes sur la Parole*. 42, 45

(Maynard et al., 2004) H. Maynard, K. McTait, D. Mostefa, L. Devillers, S. Rosset, P. Paroubek, C. Bousquet, K. Choukri, J. Goulian, J. Antoine, F. Béchèt, O. Bontron, L. Charnay, L. Romary, M. Vergnes, et N. Vigouroux, 2004. Constitution d'un corpus de dialogue oral pour l'évaluation automatique de la compréhension hors et en contexte du dialogue. Dans les actes de *Journées d'Etudes sur la Parole*. 58

(Meurs et al., 2008) M.-J. Meurs, F. Duvert, F. Lefèvre, et R. De Mori, 2008. Markov logic networks for spoken language interpretation. Dans les actes de *IIS*, Zakopane, Pologne. 153

(Mihajlovic et Petkovic, 2001) V. Mihajlovic et M. Petkovic, 2001. Dynamic bayesian networks : A state of the art. *CTIT technical reports series TR-CTIT-34*. DMW-project. 90

(Miller et al., 1994) S. Miller, R. Schwartz, R. Bobrow, et R. Ingria, 1994. Statistical language processing using hidden understanding models. Dans les actes de *HLT '94 : Proceedings of the workshop on Human Language Technology*, Morristown, NJ, USA, 278–282. Association for Computational Linguistics. 40, 41

(Minker et al., 1996) W. Minker, S. Bennacef, et J.-L. Gauvain, 1996. A stochastic case frame approach for natural language understanding. Dans les actes de *ICSLP*. 42

(Mohri et al., 2002) M. Mohri, F. Pereira, et M. Riley, 2002. Weighted finite-state transducers in speech recognition. *Computer, Speech and Language* *16*(1), 69–88. 43

(Moschitti, 2006) A. Moschitti, 2006. Syntactic kernels for natural language learning : the semantic role labeling case. Dans les actes de *NAACL HLT*, New York City, USA, 97–100. Association for Computational Linguistics. 34

(Moschitti et al., 2008) A. Moschitti, D. Pighin, et R. Basili, 2008. Tree kernels for semantic role labeling. *Computational Linguistics* *34*(2), 193–224. 34

(Murphy, 2002) K. Murphy, 2002. *Dynamic bayesian networks : representation, inference and learning*. Thèse de Doctorat. Chair-Russell, Stuart. 90, 91

(Nocera et al., 2002) P. Nocera, G. Linarès, et D. Massonié, 2002. Principes et performances du décodeur parole continue speeral. Dans les actes de *JEP*. 154

(Padó et Lapata, 2005) S. Padó et M. Lapata, 2005. Cross-linguistic projection of role-semantic information. Dans les actes de *HLT '05 : Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Morristown, NJ, USA, 859–866. Association for Computational Linguistics. 70

(Padó et Lapata, 2006) S. Padó et M. Lapata, 2006. Optimal constituent alignment with edge covers for semantic projection. Dans les actes de *ACL-44 : Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, Morristown, NJ, USA, 1161–1168. Association for Computational Linguistics. 70

(Pado et Pitel, 2007) S. Pado et G. Pitel, 2007. Annotation précise du francais en sémantique de roles par projection cross-linguistique. Dans les actes de *TALN*, Toulouse, France. 70, 71, 76

(Pearl, 1986) J. Pearl, 1986. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence* *29*(3), 241–288. 88

(Pearl, 1998) J. Pearl, 1998. *Bayesian networks*. Cambridge, MA, USA : MIT Press. 88, 89

(Pereira et Wright, 1997) F. Pereira et R. Wright, 1997. *Finite State Language Processing*, Chapter Finite-state approximations of phrase structure grammars, 149–173. The MIT Press. 42

(Petruck, 1996) M. Petruck, 1996. Frame semantics. 51

(Pieraccini et al., 1991) R. Pieraccini, E. Levin, et C.-H. Lee, 1991. Stochastic representation of conceptual structure in the atis task. Dans les actes de *HLT '91 : Proceedings of the workshop on Speech and Natural Language*, Morristown, NJ, USA, 121–124. Association for Computational Linguistics. 40

(Pieraccini et al., 1993) R. Pieraccini, E. Levin, et E. Vidal, 1993. Learning how to understand language. Dans les actes de *Eurospeech*. 38

(Pinault et al., 2009) F. Pinault, F. Lefèvre, et R. De Mori, 2009. Feature-based summary space for stochastic dialogue modeling with hierarchical semantic frames. Dans les actes de *Interspeech*. 154

(Pon-Barry et al., 2006) H. Pon-Barry, K. Schultz, E. Owen Bratt, B. Clark, et S. Peters, 2006. Responding to student uncertainty in spoken tutorial dialogue systems. *International Journal of Artificial Intelligence in Education Volume 16, Number 2/2006*, 171–194. 15

(Potamianos et al., 2005) A. Potamianos, S. Narayanan, et G. Riccardi, 2005. Adaptive categorical understanding for spoken dialogue systems. *IEEE Trans. on Speech and Audio 13*(3), 321–329. 19

(Pradhan et al., 2004) S. Pradhan, W. Ward, K. Hacioglu, J. H. Martin, et D. Jurafsky, 2004. Shallow semantic parsing using support vector machines. Dans les actes de *Proceedings of the Human Language Technology Conference/North American chapter of the Association for Computational Linguistic annual meeting*, Boston, MA. Association for Computational Linguistics. 34, 125

(Price, 1990) P. Price, 1990. Evaluation of spoken language systems : the atis domain. Dans les actes de *DARPA Workshop on Speech and Natural Language*. 15

(Raux et al., 2003) A. Raux, B. Langner, A. W. Black, et M. Eskenazi, 2003. Let's go : Improving spoken dialog systems for the elderly and non-natives. Dans les actes de *EUROSPEECH - INTERSPEECH*. 16

(Raymond et al., 2006) C. Raymond, F. Bechet, R. D. Mori, et G. Damnati, 2006. On the use of finite state transducers for semantic interpretation. *Speech Communication 48*(3-4), 288–304. 42

(Riccardi et al., 1995) G. Riccardi, E. Bocchieri, et R. Pieraccini, 1995. Non deterministic stochastic language models for speech recognition. Dans les actes de *IEEE ICASSP*. 40

(Russell, 1905) B. Russell, 1905. On denoting. *Mind, New Series 14 :56*, 479–493. 24

(Sadek et al., 1995) M. D. Sadek, P. Bretier, V. Cadoret, A. Cozannet, P. Dupont, A. Ferrieux, et F. Panaget, 1995. A cooperative spoken dialogue system based on a rational agent model : A first implementation on the ags. Dans les actes de *ESCA Workshop on Spoken Dialogue Systems*. 15

(Seneff, 1989) S. Seneff, 1989. Tina : a probabilistic syntactic parser for speech understanding systems. Dans les actes de *HLT '89 : Proceedings of the workshop on Speech and Natural Language*, Morristown, NJ, USA, 168–178. Association for Computational Linguistics. 33

(Siegel et N.J. Castellan, 1988) S. Siegel et J. N.J. Castellan, 1988. *Nonparametric statistics for the behavioral sciences* (2nd ed.). McGraw Hill. 62

(Singh et al., 2002) S. Singh, D. Litman, M. Kearns, et M. Walker, 2002. Optimizing dialogue management with reinforcement learning. *Journal of Artificial Intelligence Research 16*, 105–133. 18

(Stolcke, 2002) A. Stolcke, 2002. Srilm - an extensible language modeling toolkit. Dans les actes de *IEEE ICASSP*. 113

(Taylor, 2009) P. Taylor, 2009. *Text-to-Speech Synthesis*. Cambridge University Press. 15

(Vapnik, 1995) V. Vapnik, 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag. 126

(Vapnik, 1998) V. Vapnik, 1998. *Statistical Learning Theory*. 126, 128

(Villaneau et al., 2004) J. Villaneau, J.-Y. Antoine, et O. Ridoux, 2004. Logical approach to natural language understanding in a spoken dialogue system. Dans les actes de *Text, Speech and Dialogue, 7th International Conference*. 33

(Walker et al., 2007) M. Walker, A. Stent, F. Mairesse, et R. Prasad, 2007. Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research 30*, 413–456. 15

(Walker, 1998) M. A. Walker, 1998. Learning optimal dialogue strategies : A case study of a spoken dialogue agent for email. Dans les actes de *36th Annual Meeting of the Association of Computational Linguistics, COLING/ACL 98*, 1345–1352. 18

(Ward, 1991) W. Ward, 1991. The PHOENIX system : Understanding spontaneous speech. Dans les actes de *IEEE ICASSP*. 32

(Wielemaker, 2003) J. Wielemaker, 2003. An overview of the swi-prolog programming environment. Dans les actes de *Proceedings of the 13th International Workshop on Logic Programming Environments*. 80

(Williams et Young, 2007) J. Williams et S. Young, 2007. Scaling POMDPs for spoken dialog management. *IEEE Transactions on Audio, Speech and Language Processing 15*(7), 2116–2129. 18

(Witten et Frank, 2005) I. H. Witten et E. Frank, 2005. *Data Mining : Practical machine learning tools and techniques* (2nd ed.). Morgan Kaufmann, San Francisco. 143

(Woods, 1975) W. Woods, 1975. *What's in a Link : Foundations for Semantic Networks*. Bolt, Beranek and Newman. 50

(Woods, 1970) W. A. Woods, 1970. Transition network grammars for natural language analysis. *Computational Linguistics 13*, 591–606. 31

(Woods, 1981) W. A. Woods, 1981. Procedural semantics as a theory of meaning. Rapport technique, NTIS, Bolt, Beranik and Newman Inc. 50, 51, 55

(Woods et al., 1976) W. A. Woods, L. Bates, G. Brown, C. C. Cook, et B. C. Bruce, 1976. Speech understanding systems. Rapport technique. 31

(Young et al., 2010) S. Young, M. Gašić, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, et K. Yu, 2010. The hidden information state model : A practical framework for POMDP-based spoken dialogue management. *Computer Speech & Language 24*(2), 150–174. 18

(Young et al., 1989) S. L. Young, A. G. Hauptmann, W. H. Ward, E. T. Smith, et P. Werner, 1989. High level knowledge sources in usable speech recognition systems. *Commun. ACM 32*(2), 183–194. 31