Université de Montréal

Quelques modèles de langage statistiques et graphiques lissés avec WordNet

par
Christian Jauvin

Département d'informatique et de recherche opérationnelle
Faculté des arts et sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de maîtrise ès sciences (M.Sc.)
en informatique

août, 2003

Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé:

Quelques modèles de langage statistiques
et graphiques lissés avec WordNet

présenté par:

Christian Jauvin

a été évalué par un jury composé des personnes suivantes:

Jian-Yun Nie
président-rapporteur

Yoshua Bengio
directeur de recherche

Philippe Langlais
co-directeur

Douglas Eck
membre du jury

Mémoire accepté le 20 octobre 2003

Sommaire

Un modèle de langage statistique tente d'estimer la distribution jointe d'une séquence de mots en se restreignant typiquement aux dépendances à l'intérieur d'une fenêtre de taille fixe. Afin de contrer le problème de la sous-représentation, ces modèles sont habituellement lissés à l'aide de méthodes de redistribution de la masse de probabilité faisant appel à diverses heuristiques. Nous proposons quelques variantes de cette idée sous la forme de modèles graphiques dans lesquels on retrouve des variables cachées représentant le sens des mots ou des concepts de plus haut niveau. La structure et les valeurs que peuvent prendre les variables de ces modèles sont contraintes par une ontologie extraite de WordNet à partir du vocabulaire. L'usage de ces architectures présente l'avantage de lisser implicitement la distribution en permettant de généraliser à de nouvelles séquences de mots, reliés par des sens ou des concepts partagés avec des mots provenant de séquences déjà rencontrées. Étant donné que la distribution des sens et des concepts n'est pas observable sur la plupart des corpus d'entraînement, les paramètres du modèle sont appris à l'aide de l'algorithme EM.

Mots-clés : **modèle de langage statistique**, **désambiguïsation du sens**, **WordNet**, **modèle graphique**.

Abstract

The goal of statistical language modeling is the estimation of the joint distribution of a word sequence, usually constrained to the dependencies within a fixed size window. To fight data sparseness, language models are often smoothed with probability redistribution mechanisms which rely on different heuristics. With this study we propose some new and related ideas which take the form of graphical models with hidden variables representing word senses or some higher level concepts. The structure of the models and the values for their variables are constrained by an ontology extracted from WordNet. Using these architectures has the benefit of an implicit smoothing which permits generalization to new words sequences, if the words are linked by senses or concepts shared with words from known sequences. Given the fact that training corpora usually contain word occurrences only, learning the parameters is done with the EM algorithm.

Keywords : **statistical language modeling**, **word sense disambiguation**, **WordNet**, **graphical model**.

Table des matières

Liste des figures

Liste des tableaux

À ma mère

Chapitre 1

Introduction

1.1 Qu'est-ce que le sens ?

Dans le flot de la vie quotidienne, un doute quant au sens exact d'un énoncé ou d'une question n'intervient que dans des circonstances particulières. Mais quelle est la nature véritable de ce que nous tentons alors de mieux saisir : s'agit-il d'un *objet*, d'une *propriété*, d'une *sensation* ou d'un *processus mental* particulier ? Le sens, alors qu'il apparaît la plupart du temps naturel et omniprésent (au point d'en être presque invisible), acquiert ainsi un caractère mystérieux et intangible, lorsqu'on l'examine sous cet angle.

À première vue, sa reconnaissance ne semble pourtant pas poser de problème particulier. On peut aisément penser qu'il est possible de s'en remettre à des critères précis pour déterminer si un énoncé possède un sens. Afin de montrer cela, on pourrait imaginer une expérience comme celle-ci. On présente les phrases :

(1a) Le sens est une entité vague et insaisissable.

(1b) Vous régime en sauce et la bateau arsenal.

à un individu dont la compétence linguistique est "normale" (quelqu'un dont le français est la langue maternelle, par exemple). Cette personne n'éprouvera probablement pas de difficulté particulière à établir que la première phrase, (1a), possède une signification claire, tandis que la deuxième, (1b), n'en possède manifestement aucune. L'individu soumis à un tel test n'aura

certainement pas l'impression, ce faisant, d'analyser quoi que ce soit, ou de se livrer à une quelconque "procédure de reconnaissance". Si on l'interroge quant aux raisons de son choix, il est très probable qu'il ne pourra faire autre chose qu'hausser les épaules, et répondre que c'est "l'évidence même !". Une réaction similaire serait également à prévoir dans le cas où on lui demanderait de décrire "ce qui s'est passé en lui" au moment de la décision.

En présence de tels exemples (fortement "polarisés", pour ainsi dire), il devient trivial d'établir que certaines phrases possèdent un sens évident, tandis que d'autres constituent au contraire de purs non-sens. Ce constat est néanmoins peu utile quant à la question qui nous occupe, ce qui nous pousse donc à imaginer une variante plus problématique de l'expérience précédente :

(2a) Le sens est une entité vague et insaisissable.
(2b) La balle chaude parle dans mon champ.

où l'étrangeté de la deuxième phrase, (2b), est évidemment encore une fois en jeu. Il est cependant plausible que le jugement demande ici un temps de réflexion plus long, probablement occupé à faire "tourner et retourner" mentalement la phrase, en tentant d'y discerner *quelque chose* (ne serait-ce qu'une lointaine évocation, comme avec un poème dont le signification serait particulièrement hermétique, par exemple). La conclusion sera néanmoins probablement unanime : le sens de la première phrase est beaucoup plus évident que celui de la deuxième. On poursuit cet examen en proposant à notre individu une troisième expérience :

(3a) Le sens est une entité vague et insaisissable.
(3b) Vous faites une promenade dans les bois.

composée à première vue de deux phrases parfaitement en règle, qu'on pourrait vraisemblablement rencontrer dans une conversation ou un livre, par exemple. Pourtant, en dépit de cette apparente rectitude, il est probable que le sens de la deuxième phrase, (3b), apparaisse "suspect" dans le contexte d'une telle série d'expériences. En effet, étant donné le caractère auto-référentiel de la première phrase (le fait qu'elle traite du *sens* fait clairement référence au cadre de l'expérience en cours), un ombrage est porté sur la signification de

la deuxième, qui semble ainsi, en contraste, "sortir de nulle part". Une justification plus ou moins précise pourra être proposée (faisant possiblement appel à la notion de "contexte" ou de "continuité dans le discours"), mais la variabilité des jugements sera inévitablement plus grande, et on risque évidemment moins d'invoquer l'évidence. Pour compliquer la situation, on remarque que plusieurs justifications différentes pourront être jugées "bonnes". Il est même parfaitement possible d'imaginer un individu n'éprouvant aucun malaise avec le contraste que présentent les deux phrases. Pour lui, les deux phrases possèdent un sens clair, et il n'est pas utile de vouloir en tenter une distinction, ou encore une quantification (la signification d'une phrase étant jugée plus "grande" ou plus importante qu'une autre).

Le lecteur bien informé pourra être tenté de classifier de manière systématique la difficulté croissante de ces expériences en invoquant les *dimensions d'analyse* du langage, dont font abondamment usage la linguistique et la philosophie moderne. La phrase (1b) du premier problème serait ainsi rejetée en tant que non-sens en vertu des règles de la *syntaxe*, qui régit l'agencement des mots en fonction de leur rôle grammatical au sein de la phrase, et dont les exigences structurales sont déterminées et prévisibles dans la plupart des cas. On peut ainsi détecter (et bien souvent simplement *sentir*) la grammaticalité d'une phrase en vérifiant qu'elle ait été ou non produite par une grammaire valide. Une phrase non-grammaticale apparaîtra ainsi naturellement dénuée de sens, étant donné qu'elle ne satisfait même pas les exigences structurales préalables à la possibilité même de signifiance. La phrase (2b) du deuxième problème ne satisferait pas quant à elle aux règles de la *sémantique*, une dimension de l'univers langagier beaucoup plus diffuse et complexe, régissant "ce que l'on peut et ne peut pas dire". C'est la sémantique qui dicte par exemple (de manière implicite) que l'on ne puisse pas dire (ou penser) certaines choses au sujet de ce qui est inanimé : dans des circonstances normales, on ne peut pas dire d'un objet telle qu'une balle qu'il peut parler. Le contraste entre les phrases du troisième exemple se rapporterait plutôt à l'aspect *pragmatique* du langage, encore plus vaste et englobant, et qui concerne son déploiement et son articulation dans un contexte donné. Le fait de se retrouver dans une situation

particulière (un dîner d'affaire, par exemple) a ainsi tendance à "orienter" le contenu sémantique du *jeu de langage* [1] qu'on y joue.

Mais une fois cette rassurante classification établie, le problème du sens demeure néanmoins entier. Que se passe-t-il réellement lors du débat intérieur avec la question de savoir à quoi se rapporte la "chaleur d'une balle", ou encore le fait qu'elle puisse parler ? En arrivons-nous vraiment à nous poser ces questions au moment d'établir le sens de la phrase (2b) ? Et si oui, à quelle instance interne ou externe faisons-nous appel pour trancher ? Quel mécanisme mettons-nous en branle ? Il semblerait que notre puissance explicative soit prise ici en défaut : les critères et les règles en jeu apparaissent trop confus et inextricablement entremêlés pour qu'on puisse distinguer clairement ce qui se passe.

Et c'est de la difficulté d'établir les critères constitutifs du sens dont il sera question ici. Car la complexité du problème, que nous résolvons constamment dans sa version "quotidienne", s'apparente davantage à la troisième expérience (et en partie à la deuxième, pourrait-on dire) qu'à la première. "Posséder un sens" n'est pas une propriété tranchée des énoncés qui composent le langage de la vie quotidienne. La frontière qui sépare nos jugements est dans bien des cas définie de manière imprécise et changeante. Nous avons plutôt affaire à un concept flou et diffus, susceptible de ne pas être systématiquement l'objet d'un consensus, mais que nous manipulons néanmoins avec une aisance étonnante.

1.2 Un peu de philosophie du langage

Le problème du sens et de la signification est devenu un sujet de préoccupation central pour les philosophes avec l'arrivée du XXe siècle. C'est l'avènement de la logique contemporaine, par le biais principalement des logiciens Gottlob Frege (1848-1925) et Bertrand Russell (1872-1970) qui a le plus contribué à attirer l'attention sur les importants problèmes théoriques que pose l'étude du langage. Les nombreuses philosophies qui précèdent ce tournant historique tendaient ainsi à escamoter la question du langage, afin de se concentrer sur

[1] Il sera question un peu plus loin de cette notion particulière.

des aspects soi-disant "supérieurs" de l'esprit humain (la cognition, la pensée, l'égo ou la conscience, par exemple).

Mais les idéaux premiers de ces fondateurs de la nouvelle logique se situaient également bien au-delà de la seule sphère du langage. Le but visé était ni plus ni moins l'unification des différents champs du savoir à l'aide de l'élaboration d'un système symbolique idéal, qui allait devoir permettre de lever une fois pour toutes les difficultés et les ambiguïtés inhérentes aux systèmes de signes dont font abondamment usage la science et les arts en général, et dont on postulait l'imperfection.

On associe souvent cet ambitieux programme avec les tentatives fameuses (mais ayant pour la plupart échoué) de réduction des mathématiques à la logique pure (la plus fameuse tentative étant sans contredit les *Principia Mathematica* (WHITEHEAD et RUSSELL 1962)). C'est en effet en tant que fondement des mathématiques, le système symbolique par excellence, que l'édifice logique a tout d'abord tenté de s'édifier, en tentant de synthétiser les vérités, raisonnements et régularités qui la composent à l'aide d'un ensemble de règles logiques dont les fondements se devaient d'être absolument clairs et ultimement à l'abri de toute interprétation.

Une importante conséquence de cette révolution fut sans doute l'attention subitement attirée sur le caractère symbolique de la pensée, conçue désormais en tant que mécanisme de manipulation de signes. La question qui allait imprégner une grande part des préoccupations philosophiques modernes, soit la nature véritable du langage, venait d'être posée.

Afin de synthétiser deux conceptions antagonistes du sens qui ont prévalu à différents moments de l'histoire de la philosophie du langage, les idées d'un même philosophe particulièrement original et influent, Ludwig Wittgenstein (1889-1951), seront brièvement exposées. Après avoir publié le *Tractatus Logico-Philosophicus* (WITTGENSTEIN 1922), un court ouvrage dont les thèmes et le style [2] continuent d'alimenter les débats philosophiques et d'ins-

[2] Le *Tractatus* est organisé de manière très systématique en sept propositions principales, qui constituent les idées centrales de l'ouvrage, et qui sont ensuite explosées en une multitude de sous-propositions dont l'importance hiérarchiques est clairement établie par le système de numérotation.

pirer de multiples commentateurs, un revirement fondamental s'est opéré dans la pensée de Wittgenstein. Ce schisme est si important qu'on parle souvent du "premier" et du "second" Wittgenstein, afin de bien mettre en évidence le fossé profond qui sépare les deux discours.

1.2.1 Le sens, selon le "premier" Wittgenstein

La philosophie du premier Wittgenstein met clairement l'accent sur la logique. La logique est l'essence du monde, l'armature lui servant de fondement. Ce que doivent partager le langage et le monde, pour que le premier puisse constituer une "image" du second est une même *forme logique*. La liaison du langage au monde (sa mise en correspondance) se fait à travers un système hiérarchique dont les éléments sont articulés dans l'espace logique. Le monde est ainsi tout d'abord constitué d'*objets* simples (§2.02 du TLP), qui en sont les composantes atomiques et indivisibles, qu'il n'est pas possible d'analyser plus avant. Wittgenstein laisse floue et énigmatique la nature exacte de ces objets ; il préfère plutôt laisser la science s'occuper de cette question jugée secondaire [3]. Ces objets simples (dont la nature importe peu, mais dont le postulat est essentiel à la théorie, en tant qu'ils constituent le point où le langage joint le monde) sont ensuite organisés en *états de choses*, une configuration d'objets simples qui y "pendent les uns aux autres comme les maillons d'une chaîne" (§2.03). L'état de choses définit les relations qui unissent entre eux les objets simples : il dit par exemple que $R(a, b)$, soit que l'objet a est en relation R avec l'objet b. Il n'est pas nécessaire qu'un état de choses *existe*, l'essentiel est qu'il soit simplement *possible*. L'état de choses de ce qui est le cas (ce qui existe) est le *fait*. Et c'est ainsi la totalité des faits qui composent le monde (§1).

Pour comprendre comment le langage s'arrime au monde, afin de le représenter, on doit considérer la *proposition*, une articulation structurée de *noms* simples. Le rapport des noms simples à la proposition dans le langage est

[3] On peut certainement penser néanmoins aux particules élémentaires dont parle abondamment la physique moderne, ou encore à la hauteur ("pitch") d'une note de musique, ou un simple point de couleur dans l'espace visuel.

équivalent au rapport des objets simples à l'état de choses dans le monde. La proposition nous dit qu'un certain état de choses est un fait. Mais pour qu'elle puisse manifester ce pouvoir d'assertion, pour qu'elle puisse *dire* quelque chose, elle doit d'abord avoir en commun avec l'état de choses une même structure, une même forme. Cette structure, ce n'est pas ce que la proposition *dit*, c'est plutôt ce qu'elle *montre*. La proposition "*l'enfant mange une pomme*" ne *dit* pas que l'enfant mange une pomme, elle *dit* qu'un certain état de choses est vrai, qu'il est bel et bien ce qui est le cas. Que l'enfant mange une pomme, c'est plutôt ce que *montre* la proposition...

Cette subtile mais fondamentale distinction sera rendue plus claire par l'analogie suivante : les noms dans la proposition doivent être liés entre eux d'une manière comparable aux éléments picturaux dans une image ou un tableau. Bien que les éléments du tableau ne soient pas nécessairement identiques aux objets du monde qu'ils tentent de dépeindre (on peut penser aux effets de la perspective, par exemple), un ensemble de conventions et de règles de représentation font en sorte qu'il est possible de "comprendre" le tableau, savoir ce qu'il représente. La proposition est ainsi comparée à une image ou un tableau du monde. Bien que les formes de représentation diffèrent, il n'y a pas de différence essentielle entre l'image d'un enfant mangeant une pomme, illustrant de manière picturale (ou spatiale) cet état de choses, et la proposition "*l'enfant mange une pomme*", qui l'illustre de manière logique.

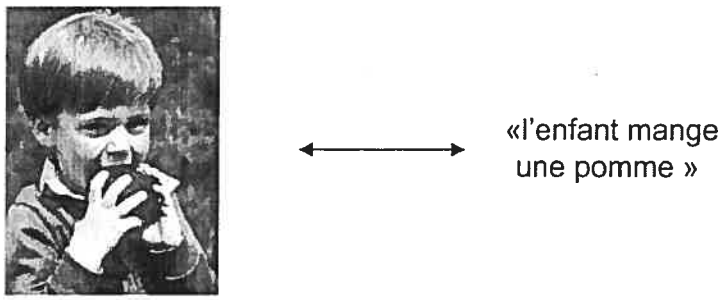


Figure 1.1 – Une proposition et une image illustrant de manière équivalente un même état de choses.

Pour qu'une proposition ait un sens, il faut que je puisse me représenter (imaginer) ce qui serait le cas si ce qu'elle dit est vrai (ou faux). La proposition

montre ainsi ce qui serait le cas si ce qu'elle dit est vrai. Une proposition ne peut pas *dire* qu'elle a un sens, elle ne peut que l'exhiber. Il est clair que la compréhension de la proposition "*cette proposition a un sens*" (entendue en tant que "pointant" vers elle-même) n'est d'aucune utilité quant à la question de savoir si elle en possède un ou pas. Nous n'écoutons pas pour ainsi dire la "voix" de la proposition, nous tournons plutôt notre "regard" vers elle, vers ce qu'elle nous désigne [4]. La même proposition, dénuée de son caractère auto-référentiel, aurait par contre un sens clair dans plusieurs circonstances [5].

Dans le cadre de ce système, il apparaît que le concept de signification ne peut s'accorder qu'avec ce qui possède une structure ou une articulation (une proposition ou un tableau, par exemple). On ne pourrait pas en effet imaginer à quoi se rapporterait le fait qu'un nom ou un objet possède un sens. Dans les problèmes pratiques dont il sera question dans cette étude, nous ferons pourtant une utilisation fréquente du concept de "sens des mots". Il doit donc être clair d'emblée que cette formulation est quelque peu fallacieuse : en général, le sens des mots ne peut se rapporter qu'à leur articulation dans un contexte donné, et non à leurs occurrences isolées.

Cette théorie de la signification est complètement indépendante de toute forme d'adéquation à la réalité, à ce qui est ou n'est pas le cas dans le "vrai monde", car elle ne constitue en aucun cas une méthode de vérification des faits. Comprendre une proposition n'a donc rien à voir avec le fait de savoir si elle est vraie ou fausse. Le sens est une exigence purement structurale, d'une nature complètement différente de celle de la vérité ou de l'adéquation au réel. Il est en outre intéressant de souligner que des énoncés "nécessairement vrais" ou "nécessairement faux" (une tautologie ou une contradiction, telle que l'antinomie du menteur) ne sont pas significatifs, dans les termes de cette théorie. Un énoncé du langage peut donc être vrai ou faux, tout en étant

[4] Ceci jette soit dit en passant une lumière nouvelle sur la fameuse antinomie du menteur : une proposition ne peut rien dire la concernant, elle ne peut que le montrer. Et ce que les propositions "*je mens*" ou "*cette phrase est fausse*" montrent, c'est simplement qu'elles n'ont pas de sens.

[5] On peut par exemple imaginer le cas d'un individu devant se soumettre à l'une des expériences de la section 1.1 et disant "*cette proposition a un sens*" (en en désignant une du doigt, par exemple).

complètement dénué de signification. Dans cette catégorie particulière sont ainsi rangées les "vérités logiques", qui n'énoncent rien sur le monde (mais qui en exhibent la structure) ainsi que l'ensemble des mathématiques. Ces systèmes ne produisent pas des "vérités" semblables à celles produites par les sciences de la nature, par exemple. Les vérités qu'elles produisent sont d'un autre ordre, structural et indépendant du monde, et c'est pour cette raison qu'on les exclut du domaine de ce qui possède une signification. La vérité d'un théorème n'est pas du même ordre que la vérité d'un fait de la vie quotidienne.

La *théorie picturale de la signification*, que nous venons d'évoquer brièvement, constitue en fait un des nombreux fondements théoriques sur lesquels reposent la philosophie du premier Wittgenstein, tout entière contenue dans le *Tractatus Logico-Philosophicus*, qui prétend un peu pompeusement avoir "définitivement réglé les problèmes philosophiques" (préface du TLP), qui sont presque toujours fondés sur des malentendus et de l'incompréhension quant à la logique de notre langage. Le "silence" de Wittgenstein, dans la dizaine d'années qui ont suivi le *Tractatus* et qui séparent sa première et sa seconde philosophie est ainsi particulièrement éloquent [6].

1.2.2 Le sens, selon le "second" Wittgenstein

Une anecdote, rapportée par Norman Malcolm, un des nombreux commentateurs de Wittgenstein, résume et illustre très bien l'importante modification qui s'est opérée dans sa pensée, et qui marque le tournant vers sa second philosophie : alors qu'il voyageait avec l'économiste Piero Sraffa, Wittgenstein soutenait avec beaucoup d'emphase, au cours d'une discussion animée, qu'une proposition et ce à quoi elle se rapporte devaient posséder le même degré de *multiciplicité logique*. Sraffa répliqua en ébauchant un geste typique des napolitains, dénotant le dégoût, et consistant en "un mouvement des doigts brossant le menton vers l'extérieur", tout en demandant "quelle était alors la forme logique de ceci ?". Il semblerait que ce simple événement ait contribué

[6] La septième et dernière proposition du *Tractatus* a été d'ailleurs à ce propos étonnamment prophétique : "Ce dont on ne peut parler, il faut le taire".

à ébranler chez Wittgenstein la certitude que le langage et le monde devaient nécessairement posséder "quelque chose" en commun...

Pour le second Wittgenstein, ce n'est que dans le contexte d'un *jeu de langage* qu'une proposition possède un sens. La question de savoir si cette proposition s'accorde ou non avec la réalité perd ainsi son aspect fondateur et central quand on reconnaît qu'elle ne s'applique que dans le jeu de langage de la *description* (ou à un qui s'y apparente). Et bien sûr une infinité d'autres jeux de langage d'égale importance composent le "langage de la vie quotidienne" (comme l'illustre Sraffa avec son geste particulier), et l'erreur du *Tractatus* aura été d'en faire abstraction. Tout comme il existe plusieurs manières de "jouer à un jeu" (jouer aux échecs, jouer au ballon, etc.), il existe une multitude de manière d'utiliser le langage, qui ne sont souvent pas plus apparentées entre elles que ne peuvent l'être des jeux différents, en dépit du fait que nous aurons tendance (et ce pour plusieurs raisons valables) à les regrouper sous la bannière d'un même concept. Et on ne voudra certes pas nier la signifiance et l'importance de ces autres aspects du langage, qui possédent des règles et des raisons d'être qui leur sont propres. Il faudra donc se résoudre à détourner le regard de l'idée séduisante selon laquelle il existe une formule magique expliquant et synthétisant le langage dans sa totalité. Afin de ne pas retomber dans les erreurs de la première philosophie, causées par le postulat à priori raisonnable de l'existence d'une forme générale de la signification, on devra construire des outils de description moins englobants, qui devront manipuler plus aisément les détails parfois contradictoires du langage ordinaire.

La notion de jeu de langage, probablement la plus importante et représentative du second corpus wittgensteinien, ne fait donc plus du tout appel à l'idéal logique du *Tractatus*, ou à toute autre forme d'abstraction ontologique se situant en-dehors de l'espace et du temps. Elle s'enracine plutôt dans le quotidien et l'ordinaire, qu'elle ne tente plus d'expliquer ou de fonder, mais simplement de décrire.

Un slogan fameux, tiré des *Investigations Philosophiques* (WITTGENSTEIN 1958), proclame que "la signification, c'est l'usage" (§43 de IP). Le sens perd ainsi l'aspect mystérieux qui en faisait une sorte d'"aura" intangible accom-

pagnant les mots d'une manière quelque peu comparable, en un certain sens, à l'éther dans les théories naïves expliquant la nature ondulatoire de le lumière, avant l'avènement de la Théorie de la Relativité. Il n'est plus dès lors que **l'expérience de l'application d'un consensus**, visant à faire d'un usage linguistique particulier un phénomène public et partagé, ayant une utilité propre, et n'ayant pour ce faire nul besoin de se rapporter à une réalité objective immuable et extérieure. Wittgenstein reconnaît néanmoins que l'idée du sens en tant qu'entité est puissante et bien enracinée. Il est ainsi très difficile de ne pas considérer que la proposition "*il faut que je répare mon ordinateur*" contienne *quelque chose de plus* que la proposition "*table voir tremble hiver*". Mais on ne doit pas perdre de vue la nature véritable de cette différence, soit une simple conséquence du fait que cet usage ait été utilisé et répété un nombre incalculable de fois dans le tissu de la vie quotidienne. La première proposition a une fonction propre et bien définie, tandis que la deuxième n'en offre apparemment aucune.

1.3 Un peu de psycholinguistique

Parallèlement à ces activités philosophiques, la psycholinguistique a également pris son essor véritable avec le bouillonnement culturel et scientifique du XXe siècle. Située au confluent de plusieurs disciplines, dont la linguistique, la psychologie et la neurologie, cette discipline moderne aborde la question du sens et du langage de manière un peu plus pratique et scientifique, en cherchant à en comprendre les mécanismes physiologiques et psychologiques au moyen d'hypothèses, de modèles et d'expérimentations.

Les fondements de la discipline ne datent cependant pas d'hier : de tout temps la multiplicité du langage humain a fasciné et intrigué. On en trouve par exemple des échos dans la Bible, avec le mythe de la Tour de Babel, qui décrit les affres d'une société dont les membres sont condamnés à ne plus pouvoir se comprendre entre eux suite à la fragmentation de leur langage, un châtiment infligé par Dieu afin de les punir de leur orgueil et de leur vanité. La question soulevée est profonde : comment la culture humaine peut-elle reposer sur la

base d'un paysage linguistique aussi varié, et surtout, est-ce qu'il existe une racine unique à l'origine de cette diversité ? Est-ce que l'idée d'un *protolangage* est raisonnable ?

La psycholinguistique s'articule principalement autour de quelques grands débats de ce genre. Est-ce que les facultés linguistiques humaines sont innées (génétiquement programmées, en quelque sorte) ou acquises (construites ou induites par l'environnement) ? Certaines observations semblent parler en faveur de la première hypothèse, popularisée par le célèbre linguiste américain (et aussi militant de gauche !) Noam Chomsky, dans son livre *Syntactic Structures* (CHOMSKY 1957). Il est ainsi remarquable que la quasi-totalité des enfants du monde apprendront correctement (et rapidement) une langue s'ils en ont une exposition suffisante (celle des parents par exemple). Ils manifestent ainsi une claire préférence pour l'aspect structural et organisé du langage (par opposition à l'incohérence des gargouillements ou des cris), ce qui n'est pas sans rappeler la préférence nettement marquée que nous entretenons naturellement à l'égard des sons mélodieux et musicaux, porteurs d'une certaine "signification" (par opposition au bruit). Des observations parlent également en faveur de certaines prédispositions physiologiques au langage : l'altération ou la destruction de certaines zones du cerveau (notamment les aires spécialisées de Broca et de Wernicke) peut anéantir les facultés langagières, sans pour autant diminuer les autres facultés cognitives, ce qui pousse à postuler une certaine "modularité" des facultés et des appareils linguistiques. Le phénomène rare et non-reproductible (pour des raisons évidemment éthiques) de l"'enfant-sauvage" est également un argument d'importance : au-delà de l'adolescence, il ne semble plus qu'il soit possible d'embrasser le langage dans toute sa richesse et sa complexité, l'appareil linguistique, souple et réceptif dans la petite enfance étant alors rigidifié en un état n'en permettant plus l'assimilation.

Mais bien entendu, comme pour de nombreux débats où sont polarisés des systèmes contradictoires, il est sage de tourner son regard en un point milieu où se rejoindront peut-être les points de vue. Il est ainsi raisonnable de considérer qu'un phénomène aussi complexe que le langage ne fait pas reposer

la totalité de son fonctionnement sur une simple "mécanique", mais qu'il est au contraire influencé et défini par une multitude de facteurs, d'une part internes et physiologiques, et d'autres part externes, comportementaux et culturels. La linguiste Jean Aitchison synthétise bien cette idée dans son ouvrage, *The Articulate Mammal : An Introduction to Psycholinguistics* : "Nature triggers off the behavior, and lays down the framework, but careful nurture is needed for it to reach its full potential. The dividing line between 'natural' and 'nurtured' behavior is by no means as clear cut as was once thought. In other words, language is 'natural' behavior–but it still has to be carefully 'nurtured' in order to reach its full potential. In modern terminology, the behavior is innately guided" (AITCHISON 1998).

1.4 Un peu de sorcellerie

En constatant les rapides progrès dans le domaine du traitement automatique du langage, il pourrait apparaître que ces préoccupations théoriques sont pour ainsi dire désuètes et futiles, alors que sont élaborées des manières concrètes de répondre à nos besoins (effectuer une traduction efficace, par exemple, ou bâtir un engin de recherche intelligent, capable de dégager les véritables besoins de l'usager). Mais il est cependant aisé de constater que les applications actuelles se heurtent à des barrières particulièrement résistantes et coriaces, qui ne pourront être franchies sans que ces questions plus profondes ne soient à tout le moins abordées.

Ne sommes-nous pas après tout en train de jouer aux apprenti-sorciers, en tentant d'insuffler ainsi à nos machines le don du langage, et par extension, de la pensée et de l'intelligence ? Et est-ce que nos espoirs d'y parvenir sont simplement fondés ? Les idées de certains penseurs montrent qu'il est légitime d'en douter. L'argument de la *Chambre Chinoise* (SEARLE 1980), imaginé par le philosophe américain John Searle est particulièrement frappant : on donne une série de règles permettant de manipuler les symboles de la langue chinoise à un individu ne sachant pas la parler, enfermé dans une chambre. Imaginons maintenant un individu *sinophone*, à l'extérieur de la chambre, à

qui on demanderait d'interagir avec l'individu enfermé au moyen de messages écrits, échangés par une fente. Étant donné son ignorance de la supercherie, il apparaîtra compréhensible que cet individu commette l'erreur de considérer cette manipulation "aveugle" des symboles en tant que compréhension *authentique* de la part de l'individu à l'intérieur de la chambre. Il suffit cependant de s'imaginer jouant le rôle de l'individu à l'intérieur de la chambre pour se rendre compte, par une sorte d'introspection, qu'une différence fondamentale et irréductible sépare les deux compréhensions. En remplaçant l'individu et ses règles de manipulation dans la chambre par un *programme*, et en proposant à un individu une interaction avec ce programme, au moyen d'entrées et de sorties, l'argument de Searle prend tout son sens : nos machines ne parviendront jamais qu'à *imiter* la cognition ou l'intelligence humaine, car leurs actions se réduiront toujours à l'application d'un ensemble de règles inertes, qu'ils ne pourront jamais véritablemenet intérioriser. Cet argument puissant et élégant (qui offre une sorte de contre-argument au fameux test proposé par Turing (TURING 1950)) a également introduit un débat important sur la nature du lien étroit qui unit les symboles du langage au monde ("Symbol Grounding Problem" (HARNAD 1990)).

1.5 Aperçu du mémoire

Le deuxième chapitre de ce mémoire consiste en une introduction à certaines notions théoriques et pratiques en traitement automatique du langage. Les notions fondamentales sur lesquelles reposent la présente étude y sont présentées, à savoir les modèles de langage statistiques et la désambiguïsation du sens. On y présente également la base de données lexicale WordNet, sur laquelle repose une part importante des idées que nous développerons.

Le troisième chapitre expose en détail la conception et l'entraînement (à l'aide de l'algorithme EM) d'un modèle de langage du second-ordre (HZSW), utilisant des variables cachées puisant leurs valeurs dans une ontologie particulière, extraite de WordNet. Une analyse approfondie du modèle à l'aide d'un univers de langage miniature permet au lecteur d'en comprendre les rouages et

les motivations. Ces explications sont accompagnées d'une évaluation comparative du modèle sur un corpus plus volumineux (Brown), ainsi que certaines extensions susceptibles de l'améliorer.

Le quatrième chapitre étend les idées du précédent en présentant un modèle de langage à l'architecture particulière (UVSW), le situant entre le deuxième et le troisième ordre, et mettant à profit de manière plus explicite un mécanisme de désambiguïsation du sens. Une évaluation comparative de ce modèle (à l'aide des corpus Brown et AP News) accompagne également sa présentation.

Le cinquième chapitre explore les causes possibles de la défaillance des modèles à l'étude, en élaborant notamment une critique de la notion même de "sens des mots", telle qu'elle est implicitement définie par les outils couramment utilisés en traitement automatique du langage.

Le sixième chapitre conclut ce mémoire en résumant les contributions théoriques et expérimentales qu'il aura permises.

Chapitre 2

Le traitement automatique du langage

Le traitement automatique du langage (ou plus communément NLP, "Natural Language Processing") rassemble une famille de théories et de techniques dont un des buts fondamentaux est d'améliorer la modélisation des compétences linguistiques humaines afin de résoudre d'importants problèmes pratiques (traduire automatiquement un document en une langue étrangère, ou comprendre une requête complexe, possiblement formulée de manière vocale, afin d'y répondre adéquatement, par exemple).

2.1 Une vieille rivalité

On retrouve au coeur de ce domaine un clivage faisant écho à celui qui sépare depuis des décennies en deux communautés plus ou moins rivales le super-domaine de l'intelligence artificielle. Cette dualité cantonne ainsi les approches *statistiques* et *symboliques* en des camps relativement autonomes et distincts, montrant parfois certaines difficultés à l'échange et la mise en commun.

Les partisans de l'approche statistique préconisent la construction de modèles d'apprentissage cherchant à dégager des régularités statistiques à l'aide d'une quantité finie de données d'entraînement souvent incomplètes et brui-

tées. L'effacité de ces mécanismes est vérifiée en testant leur capacité de *généralisation* : sur la base des exemples particuliers ayant participé à leur entraînement, ils devront avoir été en mesure d'inférer des principes plus généraux et robustes qu'ils pourront mettre à profit en présence de données inconnues. Ce principe est postulé en tant qu'un des fondements de l'intelligence et des capacités cognitives humaines et animales.

L'approche symbolique, bien qu'elle repose essentiellement sur les mêmes idéaux, a plutôt tendance à préconiser l'usage de règles et de postulats d'un niveau d'abstraction plus élevé, tentant de mettre à profit davantage de connaissances préalables ayant trait à la structure et à la nature du problème d'apprentissage à résoudre.

Il est intéressant de remarquer qu'on qualifie parfois de "sub-symbolique" la première école, afin de mettre en évidence le caractère plus "primitif" des algorithmes et des théories qui la composent. Probablement le représentant le plus fameux de cette famille, le *réseau de neurones* est un algorithme qui illustre bien cette idée : s'inspirant de certains aspects structuraux et procéduraux du cerveau, il tente d'en reproduire de manière plus ou moins fidèle les mécanismes d'apprentissage et de représentation. Il peut ainsi sembler que la véritable action d'un modèle de ce type se situe à un niveau antérieur à ceux où interviennent la formation et la manipulation réglée de symboles abstraits. Ces modèles peuvent ainsi sembler "paver la voie" aux mécanismes d'apprentissage symbolique qui pourront reposer sur leur base.

On retrouve une dichotomie apparentée en traitement automatique du langage. L'approche symbolique aura tendance, à l'instar de la super-école de laquelle elle hérite, à injecter une dose importante de connaissances linguistiques à priori dans les modèles. On abordera les problèmes en se basant tout d'abord sur des théories et des concepts linguistiques établis : on tentera par exemple d'analyser syntaxiquement une phrase en la décomposant à l'aide des règles d'une grammaire ayant la puissance générative nécessaire. On tentera également de modéliser et d'articuler la signification de cette phrase (tâche pour laquelle la syntaxe est largement insuffisante) en faisant usage de contraintes et de règles sémantiques explicites, souvent péniblement élaborées et codées

manuellement. On tentera également de modéliser le savoir et les connaissances, postulés en tant que fondement essentiel et préalable à toute pratique langagière, à l'aide de représentations sophistiquées (les *graphes conceptuels* (SOWA 1976) ou les *dépendances conceptuelles* (SCHANK 1972), par exemple). On essaiera aussi d'encapsuler le déroulement d'une situation donnée (par exemple un dîner au restaurant, ou une transaction dans un magasin) à l'aide de *scripts* (SCHANK et ABELSON 1977) précisant l'ordre, l'agencement ainsi que la signification des échanges verbaux ou écrits propres à cette situation [1]. Des entreprises encore plus ambitieuses postuleront finalement que l'aspect essentiel de tout mécanisme de traitement du langage est l'usage de la capacité vague et diffuse qu'est le "sens commun" (DOUGLAS B. LENAT 1986), constituée de la myriade de faits divers et de vérités banales qui composent la vie quotidienne des individus : le fait que les objets ont tendance à se diriger vers le sol si on les laisse tomber, le fait qu'il fait normalement plus chaud en automne qu'en hiver en Amérique du Nord, et ainsi de suite. Derrière l'évidence et la banalité de ces faits s'enchevêtrent un vaste réseau de faits intermédiaires rendant possibles des inférences complexes et nouvelles alimentant le langage et l'intelligence. La collection de ces faits, agissant comme une sorte de toile de fond au langage, est évidemment fonction de l'expérience des individus, de leur parcours particulier et accidentel. On ne peut donc pas espérer généraliser la liste exhaustive de ces faits, la "compresser" comme le ferait une théorie scientifique avec des observations et des faits épars. On n'apprend pas ces faits et ces vérités dans les livres, mais bien plutôt en expérimentant la vie normale et quotidienne d'un être humain. À cela tiennent donc les grandes difficultés inhérentes à la modélisation de ce savoir. Certains soutiennent néanmoins fermement que toute tentative sérieuse de compréhension ou d'émulation de l'intelligence doit nécessairement reposer sur un mécanisme s'y apparentant, en dépit des problèmes considérables devant être résolus.

L'approche statistique en traitement automatique du langage est celle dont il sera principalement question dans cette étude. Un modèle d'apprentissage statistique doit avoir la capacité de s'élaborer lui-même avec un minimum

[1] Ce qui évoque la notion de jeu de langage exposée à la section 1.2.2

d'aide extérieure, en construisant sa propre représentation, afin de solutionner de manière optimale le problème auquel il fait face. L'entraînement sur un grand nombre d'exemples (les mots d'un corpus) devrait ainsi lui permettre de dégager des régularités statistiques reflétant bien les mécanismes et régularités du "vrai" langage. Il est cependant clair qu'il serait absurde de négliger complètement l'apport de connaissances à priori pour la modélisation d'un phénomène aussi complexe que le langage (d'autant plus qu'il est de plus en plus facile de collecter et d'utiliser ces ressources). Un des buts de la présente étude est donc également d'étudier cette question particulière, ainsi que les problèmes qui s'y rattachent.

Entamons maintenant un examen sommaire de certaines techniques, en débutant par ce qui constitue sans doute le représentant le plus fameux et le plus étudié des algorithmes statistiques de traitement du langage.

2.2 Les modèles de langage statistiques

Un modèle de langage statistique est essentiellement fondé sur une idée très simple. Il s'agit de constuire un estimateur de la distribution empirique jointe de la série ordonnée des mots qui composent un texte ou un corpus de textes. Pour une série de variables aléatoires lexicales W_t, modélisant l'occurrence des mots d'un vocabulaire donné V à la position t dans un corpus $\mathcal{T}$ de T mots, considérons un modèle de la probabilité jointe :

$$P(W_1 = w_1, W_2 = w_2, \cdots, W_T = w_T) \equiv P(W_1^T = w_1^T) \equiv P(w_1^T) \equiv P(\mathcal{T}) \tag{2.1}$$

où les $w_t \in V$, et où des notations équivalentes et plus compactes pour les suites ordonnées de variables aléatoires sont introduites (qui seront en usage tout au long du mémoire). Une manière naïve et triviale de construire un tel modèle serait de postuler l'indépendance des variables qui le composent, ce qui permettrait de calculer la probabilité du "super-événement" que constitue le corpus dans sa totalité en tant que simple produit de la probabilité des

événements atomiques qui le composent, l'occurrence des mots :

$$P(w_1^T) = \prod_{t=1}^{T} P(w_t) \tag{2.2}$$

Mais la puissance des modèles de langage statistiques réside dans le fait qu'ils tentent plutôt de capturer la très forte dépendance qui relie les mots entre eux, et donc des variables aléatoires qui leur sont associées. Il semble tout à fait naturel et nécessaire en effet de prendre en considération l'influence évidente qu'exerce l'occurrence des mots qui accompagnent l'occurrence d'un mot quelconque. Bien que dans la plupart des cas, cette influence n'aille pas jusqu'à déterminer sans équivoque l'occurrence exacte d'un mot donné, elle n'en jouera pas moins un rôle important, en interdisant explicitement certaines possibilités (et par le fait même en en renforçant certaines autres). Ceci s'illustre en prenant une suite de mots composant un début de phrase, "*Le chien mange du...*", par exemple, et en considérant un vocabulaire V, duquel nous pourrions extraire des mots pour la compléter. En vertu de la structure syntaxique et du contenu sémantique de cette suite de mots, il est clair que l'ensemble des mots susceptibles de la poursuivre compose un sous-ensemble relativement petit de V. On peut remarquer par exemple que la structure syntaxique de ce début de phrase impose au mot suivant immédiatement la contrainte d'être un nom (ou peut-être un adjectif), tandis que son sens exige qu'il dénote quelque chose de comestible (ou que l'aspect qualificatif s'applique à quelque chose de comestible, s'il s'agit d'un adjectif), ce qui restreint de manière importante l'ensemble des possibilités. Le prédicteur qui modélise des contraintes conditionnelles de cette sorte, en s'appuyant sur l'information contextuelle dont il dispose, prendra nécessairement des décisions plus adaptées et performantes que celles d'un prédicteur aveugle, ne possédant ou n'exploitant pas cette information. Un tel modèle, prenant en considération l'influence du contexte lexical afin de conditionner les valeurs possibles des variables aléatoires, peut donc s'exprimer :

$$P(w_1^T) = \prod_{t=1}^{T} P(w_t | w_1^{t-1}) \tag{2.3}$$

Mais cette architecture plus adaptée entraîne un problème important, d'ordre combinatoire cette fois : le modèle s'articule maintenant autour d'un trop grand nombre de paramètres, qu'il n'est plus possible d'estimer et de gérer avec les ressources informatiques actuelles, pour un corpus d'une taille raisonnable (typiquement de l'ordre de quelques dizaines de millions de mots, pour un vocabulaire de quelques dizaines de milliers). Une heuristique solutionnant ce problème consiste à ne retenir qu'une fenêtre de taille fixe (typiquement de l'ordre de quelques mots seulement), précédant le mot à prédire. Une intuition simple justifie cette idée. Sans définir de manière formelle la nature exacte de l'influence que l'occurrence d'un mot peut exercer sur celle d'un autre mot, il semble raisonnable de postuler que cette influence s'atténuera, au-delà d'une certaine "distance". De même, il ne serait probablement pas souhaitable de prendre en considération la totalité des mots qui précèdent, car le bruit introduit par des mots lointains dont les rapports avec le mot à prédire sont pratiquement inexistants pourrait faire en sorte d'augmenter l'incertitude et la confusion. Les mots débutant une phrase devraient donc voir leur influence sur les mots la terminant diminuer, à mesure que la phrase s'allonge, ce qui mène directement à l'application de l'hypothèse *markovienne* selon laquelle il est raisonnable de poser qu'un système séquentiel ait une "mémoire" discrète et limitée, mais contenant à chacune de ses étapes un résumé ou l'essentiel de l'information nécessaire pour poursuivre. Un modèle ainsi réduit peut maintenant s'exprimer :

$$P(w_1^T) = \prod_{t=1}^{T} P(w_t | w_{t-n+1}^{t-1}) \tag{2.4}$$

où n est la taille de la fenêtre de contexte, un *hyper-paramètre* du modèle. On parle dans ce cas d'un modèle d'*ordre* n, ou encore d'un n-gramme [2]. Cette réduction, qui rend possible l'évaluation et l'optimisation du modèle (pour une valeur de n raisonnable), entraîne évidemment un certain nombre de problèmes. Pour une fenêtre de taille donnée n, on peut premièrement imaginer

[2] Il est à noter que nous userons du terme "n-gramme" de manière interchangeable, tantôt pour dénoter un modèle d'ordre n, tantôt pour dénoter une série de n mots consécutifs.

une multitude d'exemples montrant que l'occurrence d'un mot donné w, à la position t, dépend de celle d'un mot w' se trouvant à une distance plus grande que n, au-delà donc de la "sphère d'influence" rigide prise en considération par le modèle. L'occurrence du verbe "*manger*", dans la phrase "*Le chien mange de la bonne, tendre et appétissante nourriture*" est clairement liée à celle du nom "*nourriture*", bien que la distance qui les sépare soit plus grande que $n = 3$, une valeur typique et à peu près optimale en pratique (un modèle avec cette capacité particulière est communément appelé un *trigramme*). L'application stricte de cette réduction fait aussi en sorte de ne pas prendre en considération certaines "balises naturelles" qui jalonnent et structurent un texte : la séparation en phrases, en paragraphes ou en chapitres, par exemple. L'occurrence d'un mot d'une phrase courante risquant ainsi moins d'être influencée, en général, par l'occurrence d'un mot provenant d'une phrase précédante, il serait souhaitable d'introduire cette contrainte dans le modèle. Mais il est par contre très facile d'imaginer un exemple pathologique où un mot d'une phrase donnée exerce une influence déterminante sur un mot d'une phrase subséquente. On peut même imaginer un mot d'une importance particulièrement grande (dénotant le thème ou le sujet d'un texte par exemple) qui serait susceptible d'influencer, à lui seul, une grande partie des mots de l'ensemble du texte, en dépit de sa rareté.

Une autre point de vue par lequel on peut justifier l'usage de l'heuristique markovienne est son influence sur le nombre de *classes d'équivalence* avec lesquelles doit composer le modèle. Dans le contexte d'un problème de classification, où une décision discrète doit être produite, on doit s'en remettre à l'examen des *caractéristiques* qui décrivent ou accompagnent (en y étant par exemple fortement corrélées) l'objet à classifier. Le regroupement de ces caractéristiques en classes d'équivalence (où les ensembles de caractéristiques "similaires" se retrouveront dans une même classe d'équivalence) servira à déterminer et contraindre le domaine de la tâche de classification. Plus ces classes sont nombreuses, plus la granularité du mécanisme de classification est grande, ce qui permet davantage de précision. Mais plus les classes sont nombreuses, plus le nombre d'exemples requis pour construire un estimateur raisonnable

devient grand (nous examinerons plus en détail à la prochaine section ce problème fondamental et les techniques de lissage pour le contrer), ce qui rend le problème évidemment plus difficile. La composition des contextes de n mots accompagnant un mot donné constituent les classes d'équivalence d'un modèle de langage statistique. Plus n est grand, plus le nombre et la complexité des classes d'équivalence augmentent, améliorant la précision du mécanisme de décision, mais diminuant du même coup sa capacité à "couvrir" de nouveaux exemples, jamais rencontrés auparavant. L'optimisation de la taille et de la composition des classes d'équivalence [3] constitue certainement un des enjeux les plus importants en modélisation statistique du langage.

Une fois la taille des classes d'équivalence établie (à l'aide de l'hyperparamètre n, la taille de la fenêtre de contexte) il reste à construire les estimateurs qui serviront à quantifier les probabilités jointe et conditionnelle d'un événement aléatoire. On utilise pour ce faire une notion ayant une interprétation très intuitive, l'*estimation par maximum de vraisemblance* (MLE, "Maximum Likelihood Estimation"). Si on considère le problème de la découverte des paramètres du modèle en tant que problème d'optimisation, on peut montrer qu'une fonction de coût basée sur la vraisemblance est maximisée lorsque la probabilité jointe d'une séquence de mots est simplement estimée en "comptant" :

$$P_{MLE}(w_{t-n+1}^{t}) = \frac{C(w_{t-n+1}^{t})}{T} \tag{2.5}$$

où T est le nombre total d'exemples d'entraînement présentés au modèle, et $C(\cdot)$ est la fréquence (le compte) d'un événement. La probabilité conditionnelle d'un mot étant donné un contexte est estimée de manière similaire :

$$P_{MLE}(w_t|w_{t-n+1}^{t-1}) = \frac{C(w_{t-n+1}^{t})}{C(w_{t-n+1}^{t-1})} \tag{2.6}$$

Un petit examen nous convainc rapidement que cet estimateur est intuitivement satisfaisant : pour un contexte donné, plus le nombre de mots *pouvant* le compléter est grand, plus la probabilité de chacun sera petite, devant être

[3] Nous verrons plus loin qu'il existe d'autres façons de construire ces classes d'équivalence.

"partagée" en un plus grand nombre de parts. À l'inverse, s'il n'y a qu'un seul mot susceptible de compléter un groupe de mots donnés, la probabilité de son occurrence devient 1 (l'événement est certain).

Examinons quelque peu le concept d'influence dont nous avons fait usage, jusqu'ici, sans nous y attarder véritablement. Bien que l'intuition se satisfasse de l'idée selon laquelle le verbe "*manger*" joue un quelconque rôle (assurément sémantique) quant au choix du nom "*nourriture*" dans la phrase donnée précédemment en exemple, il reste qu'on gagnerait certainement quelque chose à pouvoir caractériser de manière plus précise la nature de cette influence (ou de cet agrégat d'influences). Comme nous l'avons fait précédemment, on peut dans un premier temps dégager l'influence syntaxique, dont les règles, de nature purement structurale, sont relativement faciles à extraire. Un "étiqueteur grammatical" (POS tagger) fonctionne ainsi sur la base de l'extraction de ce type de règles, afin de prédire efficacement , à l'aide d'un historique de taille étonnamment restreinte, le rôle grammatical qu'un mot donné joue à l'intérieur d'une phrase. Pour un modèle de langage statistique, visant à prédire un mot, et non un rôle grammatical, les contraintes syntaxiques sont pourtant clairement insuffisantes. Nous avons besoin d'une plus grande puissance d'expressivité. L'influence sémantique introduit ainsi une dimension d'analyse beaucoup plus vaste et difficile à caractériser. Il s'agit d'un espace dont les contours et les modes de représentation sont beaucoup plus nébuleux, faisant appel à la signification des mots et des phrases, problème pour lequel il a été démontré, au premier chapitre, que nous ne possédons pas de solution universelle. Pourtant, un modèle qui parviendrait à analyser les différents modes d'influence qui régissent la distribution des mots dans un texte (pour peut-être ensuite tenter d'en pondérer l'importance relative, par exemple) se montrerait probablement beaucoup plus souple et performant que les modèles actuels, encore incapables de ce genre de finesse. L'influence sémantique du verbe "*manger*", dans la phrase "*Le chien mange de la nourriture.*" est ainsi probablement plus significative que celle du verbe "*discuter*", dans la phrase "*Mes parents discutent de leur projet.*", parce qu'elle restreint de manière plus importante le domaine lexical duquel le mot agissant comme complément du

verbe sera extrait. Ainsi, bien que ce soit probablement discutable, on peut être porté à croire que le domaine "des choses que peut manger un chien" est plus restreint que le domaine des "choses dont mes parents peuvent discuter".

Une autre limite des modèles de langage statistiques, liée à la précédente, est dûe au fait qu'ils reposent en général sur la modélisation de l'occurrence des mots. Dans un contexte langagier, l'occurrence d'un mot, en elle-même, ne constitue pas toujours un phénomène d'une grande valeur informative. Le choix d'un mot particulier relève souvent, par exemple, de préférences stylistiques ou même carréments accidentelles de la part de l'auteur ou de l'orateur. Un modèle aurait donc avantage à ne pas se laisser trop facilement "distraire" par le bruit introduit par le choix accidentel de mots. Au lieu de s'attarder à ces phénomènes trop souvent superficiels, il devrait pouvoir s'attarder à la couche conceptuelle plus abstraite, véhiculant une information plus riche et robuste, se trouvant au-dessous.

2.2.1 Pourquoi veut-on modéliser le langage ?

Dans un contexte où le but ultime est d'arriver à construire un mécanisme capable de dériver (ou même de comprendre) le sens d'une phrase ou d'un texte (afin, par exemple, d'exécuter un ordre ou une requête), il peut sembler étrange et peu naturel de mettre ainsi l'accent sur l'aspect stochastique du langage, de son caractère accidentel et chaotique. Quels peuvent être les avantages d'un système qui aborde le langage en les termes d'une *distribution* de mots ? Ne serait-il pas possible d'aborder le problème de manière plus déterministe, sans le concours du hasard et des probabibilités ? Pour répondre à ces objections tout de même légitimes, il nous faut revenir à la théorie de l'apprentissage, dont les fondements sont résolument statistiques. La construction d'un modèle robuste, résistant aux variations et susceptible de bien généraliser en présence de données nouvelles implique nécessairement un "mécanisme d'exposition préalable à un échantillon de ces variations", lui permettant de formuler des hypothèses quant à la structure du "vrai générateur" (caché ou implicite), responsable des phénomènes de surface. La construction d'un modèle de langage statistique consiste donc en la recherche de ce "vrai générateur"

du langage, se cachant sous la surface des mots. On peut formuler de multiples hypothèses quant à la nature de ce générateur, et on peut même aller jusqu'à douter de son existence, comme nous l'avons vu. Mais quoi qu'il en soit, l'adhésion à ce paradigme probabiliste présente l'avantage indéniable d'un ancrage théorique robuste et bien établi dans la pratique.

Et c'est dans cette perspective que les modèles de langage statistiques sont rarement utilisés "en tant que tel", mais plutôt en tant que composantes dans des systèmes de traitement de plus haut niveau : ils peuvent par exemple servir à évaluer et ordonner des résultats intermédiaires (des réponses potentielles) dans des systèmes de **traduction automatique** ("machine translation") (BROWN, COCKE, PIETRA, PIETRA, JELINEK, LAFFERTY, MERCER et ROOSSIN 1990), de **reconnaissance de la parole** ("speech recognition") (JELINEK 1997) , de **recherche d'information** ("information retrieval") (PONTE et CROFT 1998), de **correction de l'orthographe** ("spelling correction") (KERNIGHAN, CHURCH et GALE 1990), d'**analyse grammaticale** ("parsing") (CHURCH 1988) et de **reconnaissance des caractères manuscrits** ("handwritten recognition") (SRIHARI et BALTUS 1993), entre autres.

2.2.2 Le lissage ("smoothing")

Un problème fondamental auquel doit faire face un modèle de langage statistique entraîné sur des données concrètes est la faible *densité* [4]. de ces données. Ce problème semble en soi quasiment insurmontable, quand on considère l'explosion exponentielle de combinaisons possibles, régie par la taille n de la fenêtre de contexte et la taille du vocabulaire, $|V|$. Pour une valeur de n relativement petite (typiquement 3) et un vocabulaire de taille raisonnable (autour de 25 000 mots) le nombre de combinaisons possibles atteint déjà une valeur astronomique : $|V|^n = 25\,000^3 \approx 15 \times 10^{12}$! Il n'est pas du tout envisageable de produire un échantillonnage satisfaisant dans cet espace avec un corpus de taille comparable à ceux qu'il nous est actuellement techniquement possible de construire. Le problème est d'autant plus difficile que les

[4] Une traduction approximative de "data sparseness"

événements rares (mais tout de même plausibles) formant la queue de la distribution sont particulièrement difficiles à capter, et une augmentation, même significative, de la taille du corpus ne parviendra pas à produire une augmentation significative de leur couverture. La situation s'aggrave évidemment à mesure qu'on tend asymptotiquement vers la couverture totale, comme nous laisse l'entendre la loi de Zipf [5](ZIPF 1949). La pratique a permis d'établir un constat clair : un modèle de langage statistique reposant uniquement sur l'estimation par maximum de vraisemblance ne parviendra jamais à produire autre chose qu'un estimateur fortement bruité, et ne couvrant pas adéquatement l'espace de probabilité (estimant donc nulle la probabilité de nombreux événements plausibles).

Pour obtenir un estimateur souple et robuste, capable de généraliser, il faut dès lors se tourner vers des solutions de rechange, évidemment heuristiques : les méthodes de lissage. L'idée de base est encore une fois très simple : il s'agit de modifier le modèle en y incorporant un mécanisme de redistribution de la masse de probabilité, des événements rencontrés dans le corpus d'entraînement (probablement surestimés [6]), vers les événements absents du corpus (possiblement sous-estimés). Ce mécanisme tendra donc à modifier la distribution, à en aplanir les brusques variations en restribuant de manière plus équitable la masse de probabilité. Il n'existe malheureusement pas une méthode remplissant parfaitement ce mandat, car le problème est en substance très difficile.

Pour s'en convaincre, considérons l'espace de toutes les combinaisons possibles de n mots, respectant la contrainte d'un vocabulaire donné. Comme nous l'avons vu, cet espace est gigantesque. En comparaison, il est clair que la taille du sous-espace des combinaisons de n mots "acceptables" est beaucoup moins importante. Mais qu'est-ce qu'une combinaison de n mots "accepta-

[5]Cette loi caractérise la relation entre la fréquence d'occurrence d'un mot et son *rang*, soit la position à laquelle on retrouve cette fréquence dans une liste ordonnée de manière décroissante : $f \propto \frac{1}{r}$.

[6]Un événement rencontré une seule fois se voit typiquement attribuer une importance beaucoup trop grande ; mais ce biais diminue à mesure que la fréquence empirique de l'événements augmente.

bles" ? Cette question renvoie encore une fois à une question centrale qui imprègne toute cette étude : est-ce qu'il existe un critère pouvant déterminer si une série de mots fait ou non partie du langage. Et cette question peut prendre de multiples formes, comme nous l'avons vu : est-ce que la série de mots a un sens, est-ce qu'elle respecte les règles de la syntaxe, est-ce qu'elle respecte les règles pragmatiques implicites qui régissent la situation de laquelle elle est tirée ? Aucune réponse potentielle à ces questions n'est susceptible de résoudre à elle seule le problème. Une fonction booléenne (ou graduée en valeurs réelles) `appartient_au_langage(proposition)`, dont la découverte réglerait pratiquement tous les problèmes théoriques et pratiques en traitement du langage, demeure une chimère inaccessible !

Plusieurs méthodes, fondées sur des principes et des heuristiques parfois complémentaires, offrent toutefois des solutions satisfaisantes. Il serait fastidieux d'exposer tous les minutieux détails qui les composent, aussi nous contenterons-nous d'un bref tour d'horizon présentant les idées de quelques méthodes parmi les plus populaires, et qui fonctionnent bien en pratique. Le lecteur désirant approfondir cette matière devrait se référer aux très complètes et exhaustives études de Stanley Chen et Joshua Goodman (CHEN et GOODMAN 1998; CHEN et GOODMAN 1999; GOODMAN 2001), desquelles nous nous sommes fortement inspiré.

Le lissage additif

Examinons tout d'abord le mécanisme de redistribution le plus simple qu'on puisse imaginer, le *lissage additif* (LIDSTONE 1920; JOHNSON 1932; JEFFREYS 1948) qui se contente d'ajouter à la compilation des fréquence $C(\cdot)$ une quantité δ (typiquement entre 0 et 1), et d'ajuster la normalisation en conséquence :

$$P_{additif}(w_t|w_{t-n+1}^{t-1}) = \frac{C(w_{t-n+1}^{t}) + \delta}{C(w_{t-n+1}^{t-1}) + \delta|V|} \tag{2.7}$$

Ceci aura notamment pour effet de donner une probabilité non-nulle à des événements jamais rencontrés lors de la phase d'entraînement du modèle, ce

qui revient à poser l'hypothèse raisonnable selon laquelle la probabilité à priori des événements (qu'ils aient été rencontrés ou non) suit une loi uniforme. Un problème fondamental avec cette approche met en lumière une des difficultés du lissage : une redistribution trop impartiale de la masse de probabilité va faire en sorte de favoriser un trop grand nombre d'événements aléatoires non-plausibles, aux dépens bien entendu des événements plausibles.

La formule de Good-Turing

La loi de Good-Turing (GOOD 1953) ne constitue pas à proprement parler une méthode de lissage. Il s'agit plutôt d'un estimateur qui "corrige" la fréquence r d'un événement :

$$r^* = (r+1)\frac{n_{r+1}}{n_r} \tag{2.8}$$

où n_r est le nombre d'événements apparaissant r fois dans le corpus d'entraînement. L'idée sur laquelle repose cette heuristique est de calculer $E(p_i|C(\lambda_i) = r)$, l'espérance de la "vraie" probabilité p_i d'un événement λ_i, pour lequel tout ce que nous savons est sa fréquence, r (on ne connaît pas l'*identité* de λ_i). Pour une dérivation complète de cet estimateur, on devra consulter (CHEN et GOODMAN 1998). Cette méthode de réestimation est souvent combinée en pratique à d'autres méthodes de lissage.

Le lissage de Katz

Un modèle d'ordre n lissé avec la méthode de Katz (KATZ 1987) peut s'en remettre au modèle sous-jacent d'orde $n-1$ si la fréquence du n-gramme qu'il tente d'estimer est nulle. Il se peut par exemple que le trigramme "*calcul rapidement exécuté*" ne soit pas observé dans un corpus d'entraînement donné. Un prédicteur d'ordre $n = 3$ se verrait donc dans l'obligation d'accorder une probabilité nulle à $P(exécuté|calcul, rapidement)$. Il est pourtant possible qu'*autre chose* ait été "*rapidement exécuté*" dans ce corpus (un programme, ou un ordre, par exemple) et qu'un prédicteur d'ordre inférieur $n = 2$ puisse par conséquent se prononcer de manière adéquate. Si ce n'est pas le cas,

on peut encore descendre d'un étage, et s'en remettre à l'unigramme (sur un très gros corpus, il est en effet très probable que quelque chose soit d'une manière ou d'une autre "*exécuté*").

Pour un trigramme lissé avec cette méthode, nous avons la formule récursive :

$$P_{katz}(w_t|w_{t-2},w_{t-1}) = \begin{cases} P_{MLE}(w_t|w_{t-2},w_{t-1}) & \text{si } r > k \\ d_r P_{MLE}(w_t|w_{t-2},w_{t-1}) & \text{si } 0 < r \leq k \\ \alpha(w_{t-2},w_{t-1}) P_{katz}(w_t|w_{t-1}) & \text{sinon} \end{cases} \quad (2.9)$$

où r est la fréquence du n-gramme estimé ($C(w_{t-2},w_{t-1},w_t)$ dans le cas du trigramme), k est un seuil au-delà duquel la réestimation ne prendra pas effet, d_r est une pondération pour l'estimateur MLE ("discount"), basée sur la loi de Good-Turing :

$$d_r = \frac{\frac{r^*}{r} - \frac{(k+1)n_{k+1}}{n_1}}{1 - \frac{(k+1)n_{k+1}}{n_1}} \quad (2.10)$$

et $\alpha(\cdot)$ est un facteur de renormalisation pour les modèles d'ordre inférieur. Dans le cas du trigramme, ce facteur se calcule :

$$\alpha(w_{t-2},w_{t-1}) = \frac{1 - \sum_{w':C(w_{t-2},w_{t-1},w')>0} P_{katz}(w'|w_{t-2},w_{t-1})}{1 - \sum_{w':C(w_{t-2},w_{t-1},w')>0} P_{katz}(w'|w_{t-1})} \quad (2.11)$$

ce qui permet de redistribuer au modèle d'ordre $n-1$ la part $(1-d_r)$ de masse de probabilité qui avait été retirée à l'estimateur MLE d'ordre n.

Le lissage de Jelinek-Mercer

Le lissage de Jelinek-Mercer fait appel à la notion générale de *mixture de modèles*. Pour l'estimation d'un événement aléatoire donné, les réponses de plusieurs modèles peuvent bien entendu différer. Mais il n'est par contre pas nécessairement le cas qu'un de ces modèles ait "davantage raison" que les autres. Il est possible que ces modèles s'attardent simplement à des aspects

différents de l'événement observé. Il est donc ainsi concevable qu'un *méta-modèle*, capable de combiner les forces respectives de ces différents modèles, soit en mesure de fournir une meilleure solution. L'application de cette idée à un modèle de langage statistique (JELINEK et MERCER 1980) lui procure une grande robustesse : pour un modèle d'ordre n, on peut consulter les modèles d'ordre inférieur sous-jacents $n-1, n-2, \cdots$, afin de combiner (ou interpoler) leur décision en une prédiction de plus grande qualité.

La version la plus simple de cette idée consiste seulement en une mixture de modèles d'ordre différent, pondérés avec des poids fixes. Dans le cas d'un trigramme, nous avons :

$$\begin{aligned} P_{interp}(w_t|w_{t-2}, w_{t-1}) &= \lambda_3 P_{MLE}(w_t|w_{t-2}, w_{t-1}) + \\ &\quad \lambda_2 P_{MLE}(w_t|w_{t-1}) + \\ &\quad \lambda_1 P_{MLE}(w_t) + \\ &\quad \lambda_0 \frac{1}{|V|} \end{aligned} \tag{2.12}$$

où la seule contrainte est $\sum_n \lambda_n = 1$. Cette mixture simple pourrait toutefois se révéler trop "rigide", étant donné qu'elle s'applique uniformément à toutes les prédictions. Afin de la rendre plus souple, on peut faire en sorte de la définir en fonction d'un aspect caractéristique de la prédiction courante. Un aspect possible est le contexte h, soit le bigramme $w_{t-2}w_{t-1}$ à partir duquel on tente de prédire le mot courant w_t. Les paramètres de la mixture forment ainsi une immense table de $|V|^2$ (le nombre de bigrammes possibles) par $n+1$ éléments. Mais étant donné que la plupart de ces contextes ne seront jamais rencontrés lors de l'entraînement, il nous faut encore une fois lisser la distribution ! Une manière simple de résoudre ce problème est de projeter l'espace à haute dimensionnalité des contextes vers un espace de plus faible dimensionnalité. On peut pour ce faire utiliser la fréquence d'occurrence $C(\cdot)$ d'un contexte h sur le corpus d'entraînement [7] pour composer une transformation qui aura l'effet

[7] Ce qui correspond simplement à la table de fréquence du bigramme non-normalisée, dans le cas d'un trigramme.

escompté :

$$h \rightarrow f(C(h)) = \lceil -\log \frac{C(h)+1}{T} \rceil = q(h) \tag{2.13}$$

Ceci réduit grandement le nombre de rangées de la table des paramètres de la mixture, qui passe de $|V|^2$ à un nombre compris entre 0 et $-\log \frac{1}{T}$ (qui varie donc en fonction de la taille du corpus et de la base du logarithme). La forme finale de notre trigramme interpolé devient donc :

$$\begin{aligned} P_{interp}(w_t|w_{t-2}, w_{t-1}) &= \lambda_3[q(h)]P_{MLE}(w_t|w_{t-2}, w_{t-1}) + \\ &\quad \lambda_2[q(h)]P_{MLE}P(w_t|w_{t-1}) + \\ &\quad \lambda_1[q(h)]P_{MLE}P(w_t) + \\ &\quad \lambda_0[q(h)]\frac{1}{|V|} \end{aligned} \tag{2.14}$$

Dans la version simple de la mixture (2.12) où les paramètres ne sont qu'un vecteur λ de $n+1$ éléments, il semblerait raisonnable d'établir manuellement la valeur des poids. On voudrait probablement accorder une plus grande part de confiance aux modèles plus informés, d'ordre supérieur, mais du reste, le choix de ces poids serait quelque peu arbitraire et probablement sub-optimal. Dans la version de la mixture où la paramétrisation est plus complexe (2.14) cette méthode manuelle conviendra encore moins : on préférerait un mécanisme automatisé, capable d'optimiser l'attribution des valeurs aux poids en fonction d'un critère. L'algorithme de Baum-Welch (BAUM 1972) (en fait une version particulière de l'algorithme EM) est une technique très utilisée en pratique pour résoudre ce problème d'optimisation. On commence tout d'abord par initialiser les paramètres de la mixture de manière uniforme, ce qui constitue l'à priori le plus raisonnable : $\lambda_i = \frac{1}{n+1}$. La première étape ("E-step") de l'algorithme consiste ensuite en le calcul de l'espérance conditionnelle du fait que *le modèle d'ordre i ait été le "bon" modèle* [8], étant donné les observations

[8] L'interprétation de λ_i peut porter ici à confusion : on peut l'interpréter en tant que la probabilité du fait que le modèle d'ordre i ait été le "bon", ou de manière équivalente en

disponibles :

$$E(\textit{modèle } i \textit{ est le bon}|\mathcal{T}) = \frac{\sum_t \lambda_i P_i(w_t)}{\sum_t \sum_j \lambda_i P_j(w_t)} \tag{2.15}$$

où P_i est le modèle d'ordre i non-pondéré. La deuxième étape ("M-step") réestime les paramètres de la mixture :

$$\lambda_i \leftarrow \frac{E(\textit{modèle } i \textit{ est le bon}|\mathcal{T})}{\sum_j E(\textit{modèle } j \textit{ est le bon}|\mathcal{T})} \tag{2.16}$$

En répétant cette procédure itérativement un certain nombre de fois, on converge vers un maximum local de la fonction de vraisemblance. On remarquera toutefois qu'il ne serait pas judicieux d'exécuter cette procédure sur les données de l'ensemble d'entraînement, étant donné que le modèle d'ordre le plus élevé (celui d'ordre n) dominerait systématiquement sur les autres, d'ordre inférieur [9]. Une manière simple de réduire ce biais indésirable consiste à exécuter la procédure d'optimisation de la mixture sur un jeu de données séparé, réservé pour cet usage, et avec lequel les différents modèles auront tous une chance égale de prouver leur valeur.

Quelques autres méthodes de lissage

Plusieurs autres techniques de lissage sont utilisées en pratique. Le lissage de Witten-Bell (WITTEN et BELL 1991), développé dans le contexte de la compression de texte, est une variante de Jelinek-Mercer dans laquelle les poids de la mixture sont estimés en tenant compte du nombre de mots uniques suivant un contexte. Le lissage "Absolute Discounting" (NEY, ESSEN et KNESER 1994) est une autre méthode d'interpolation qu'on complémente avec le retrait d'une quantité D fixe aux comptes positifs. Le lissage de Kneser-Ney (KNESER et NEY 1995) repose sur le constat que les modèles d'ordre inférieur

tant que la pondération du modèle d'ordre i, quantifiant son degré de participation dans la décision.

[9] Le modèle d'ordre le plus élevé, étant le plus informé des aléas de l'ensemble d'entraînement, offrirait donc les meilleures réponses, ce qui ferait en sorte de faire tendre la composante correspondant à ce modèle à 1 ($\lambda_n \rightarrow 1$), et les autres composantes à 0.

sont ceux sur lesquels l'attention doit être vraiment portée, étant donné qu'on peut habituellement se fier à la qualité de l'estimateur MLE d'ordre n. Cette méthode propose donc de pondérer l'importance des modèles d'ordre inférieur de manière plus précise, en utilisant le nombre de contextes différents dans lesquels l'événement intervient. Si on veut estimer la probabilité du bigramme "*New York*" par exemple, $P(York|New) = \alpha P(York|New) + \beta P(York)$, l'importance du modèle du premier ordre estimant la probabilité de "*York*" devrait être faible, étant donné que "*York*" apparaît rarement dans un autre contexte (il est raisonnable de croire que le modèle bigramme se tirera d'affaire seul). Un modèle interpolé avec Jelinek-Mercer aurait au contraire tendance à favoriser les deux modèles. (CHEN et GOODMAN 1998) proposent une version modifiée de Kneser-Ney qui donne d'excellents résultats en pratique.

Quelques autres idées

Il existe de nombreuses méthodes (autres que le lissage) pour tenter d'améliorer la performance d'un modèle de langage. L'idée d'augmenter l'ordre n d'un modèle n'offre pas généralement de gains assez importants pour justifier le coût additionel en mémoire et en temps de calcul. Les modèles de "skipping" (ROSENFELD 1994; HUANG, ALLEVA, HON, HWANG et ROSENFELD 1993; NEY, ESSEN et KNESER 1994; MARTIN, NEY et ZAPLO 1999; SIU et OSTENDORF 2000) proposent de relâcher quelque peu la contrainte régissant l'ordre des mots de contexte. Bien que "*La femme mange rapidement*" n'ait pas été rencontré lors de l'entraînement, par exemple, un modèle de skipping $P(w_t|w_{t-3}, w_{t-1})$ (qui laisse donc tomber le deuxième mot afin d'incorporer le premier au contexte) lui attribuerait tout de même une probabilité non-nulle, pour peu que "*La ___ mange rapidement*" ait été rencontré.

Une idée avec laquelle les modèles de cette étude présentent quelques ressemblances est le *regroupement en classes* [10] (BROWN, PIETRA, DESOUZA, LAI et MERCER 1992), avec laquelle on considère des classes de mots pour une caractérisation plus générale du contexte, ce qui permet de réduire de

[10] Une traduction très approximative du terme anglais "clustering", beaucoup plus explicite.

manière significative le nombre de classes d'équivalence. Au lieu de considérer par exemple le contexte lexical $w_{t-2}w_{t-1}$ pour la prédiction d'un mot w_t donné, on peut construire un estimateur moins rigide $P(w_t|Cl(w_{t-2}), Cl(w_{t-1}))$, se basant plutôt sur la classe syntaxique ou sémantique $Cl(\cdot)$ des mots qui le composent. Ceci présente l'avantage d'augmenter la densité des données d'entraînement et de permettre l'injection d'une information plus abstraite dans le modèle. On pourrait par exemple capter des schémas généraux du type "*le ANIMAL mange du NOURRITURE*". Il reste toutefois que le problème de la construction des classes est évidemment difficile.

La technique du "caching" (KUHN et DEMORI 1990) permet d'incorporer à un modèle statique et rigide (déjà entraîné) un modèle construit dynamiquement à l'aide de l'historique des données de test : $P(w|h) = \alpha P_{stat}(w|h) + \beta P_{dyn}(w|h)$.

De nombreuses autres méthodes sont décrites dans (GOODMAN 2001; ROSENFELD 2000), dont les modèles à *maximum d'entropie* (ROSENFELD 1994), qui visent à contrer le problème de la *fragmentation* (le fait que des modèles plus précis doivent être créés avec des données dont la densité est de plus en plus faible), et les modèles de mixture de phrases ("sentence mixture models") (IYER, OSTENDORF et ROHLICEK 1994) qui tentent de cloisonner différents types de phrase, afin de les assigner à des modèles qui y sont spécialement adaptés. (CHEN et GOODMAN 1998; GOODMAN 2001) étudient finalement de manière très systématique et probante la question complexe de la combinaison de toutes ces méthodes.

Un modèle de langage connexionniste : NNLM

Un autre modèle de langage statistique, NNLM [11] (BENGIO, DUCHARME, VINCENT et JAUVIN 2003; BENGIO 2002), met à profit un réseau de neurones à très forte capacité pour le calcul et l'optimisation de la fonction de vraisemblance servant de critère à son apprentissage. Ce modèle met en évidence le fait que la réalisation concrète d'un modèle stochastique n'a pas à être confinée à une seule famille d'algorithmes ou de modes de représentation. Bien que la

[11] Développé au laboratoire d'apprentissage LISA de l'Université de Montréal.

définition et la procédure d'entraînement des modèles de langage statistiques examinés jusqu'à présent aient pu fortement suggérer un mode de représentation particulier, basé sur l'utilisation de *tables* servant à la compilation des fréquences, il reste que d'autres avenues sont possibles.

La clé de voûte du modèle NNLM est la *représentation distribuée* des mots qui composent le vocabulaire : chaque mot y est associé à un vecteur dans un espace de dimension m, où $m \ll |V|$. Ces vecteurs constituent l'entrée du réseau de neurones, qui les combine au reste des paramètres à l'aide d'une topologie sans boucle de rétroaction ("feedforward"). Le réseau de neurones optimise la vraisemblance du corpus d'entraînement par *rétropropagation* [12].

La compression offerte par la représentation distribuée du vocabulaire présente l'avantage de permettre l'exploitation d'un contexte plus large que les modèles précédents, reposant sur une représentation moins compacte. Étant donné que les vecteurs d'entrée (les vecteurs de m éléments correspondant aux mots dans l'espace de la représentation) sont entraînés de manière identique aux autres paramètres du modèle (en subissant donc les mêmes modifications ayant pour but de minimiser la fonction d'erreur), une sorte d"'espace sémantique" devrait également être progressivement défini, au cours de l'entraînement. Deux mots sémantiquement "proches" devraient ainsi voir leurs représentants vectoriels dans l'espace de la représentation "proches" eux-aussi, au sens euclidien ou géodésique. À partir par exemple de la phrase :

(1a) Le chat marche dans la chambre.

on aimerait pouvoir généraliser à des phrases dont le sens est proche :

(1b) Le chat marche dans la pièce.
(1c) Le chien marche dans le salon.
(1d) Le chien court dans la chambre.
...

Un avantage important de ce modèle est un lissage implicite : étant donné que le réseau de neurones produit une fonction lisse et continue, une fine variation

[12] La fonction d'erreur est minimisée en entraînant les paramètres dans la bonne direction, indiquée par le gradient.

dans le vecteur d'entrée résultera en une variation proportionnelle de la probabilité de sortie. Des recherches sont effectuées (BENGIO et SENÉCAL 2003) afin d'améliorer les temps de calcul pour le moment quelque peu prohibitifs de cet algorithme.

2.3 La désambiguïsation du sens

Un autre problème fondamental en traitement automatique du langage est la *désambiguïsation du sens* (ou plus communément WSD, "Word Sense Disambiguation"). Dans les activités langagières de la vie quotidienne, il peut sembler que nous manipulions le sens des mots ou des expressions avec une aisance très grande. Comme cela a été évoqué dans l'introduction, il peut même sembler que nous ne fassions rien de spécial ou de particulier quand nous répondons par les gestes appropriés à l'ordre "*dépose le livre sur la table !*", témoignant de ce fait de notre compréhension. C'est ainsi que le problème ne se dévoile que lors d'une tentative d'émulation de cette compréhension par un quelconque procédé algorithmique. À moins d'évoluer dans un monde en vase clos, où l'ensemble des choses et des symboles s'y rapportant est réduit de manière importante, un système devant manipuler les symboles de cette phrase devra être en mesure de discriminer efficacement entre plusieurs interprétations possibles. Pratiquement tous les mots qui composent la phrase possèdent des modes d'usage pouvant différer, selon le contexte. Et l'ambiguïté n'est bien entendu pas confinée au seul aspect lexical du langage : on pourrait imaginer par exemple le cas d'une machine se trouvant en compagnie d'autres machines dans une pièce, et se demandant avec angoisse à *qui* (ou à *quoi* !) s'adresse réellement l'ordre...

L'étude de cet exemple particulier pourrait suggérer l'hypothèse selon laquelle nous comprenons cette phrase car nous la saisissons dans sa totalité, sans devoir recourir à une analyse de ses constituants individuels. L'image d'un "livre déposé sur une table" est ainsi tellement familière et fréquente que nous choisissons instantanément et correctement le sens approprié pour les mots polysémiques "*livre*" et "*table*".

On remarque que certains sens sont ainsi beaucoup plus courants que d'autres (les sens "*meuble*" de "*table*", et "*assemblage de pages*" de "*livre*" sont par exemple très usuels). Cette observation se traduit très bien en les termes d'un algorithme de désambiguïsation, probablement le plus simple qu'on puisse imaginer : si on possède un corpus étiqueté [13] pour lequel on a compilé la fréquence des événements "mot accompagné d'un sens", on peut établir que le sens d'un mot pris dans un *nouveau* contexte sera simplement le plus fréquent, selon notre compilation :

$$s^* = \arg\max_s C(w, s), \forall s \in S(w) \tag{2.17}$$

où $C(\cdot)$ correspond à notre compilation des fréquences des mots accompagnés de leur sens véritable et $S(w)$ est l'ensemble de tous les sens que peuvent dénoter un mot w particulier.

Cette manière de faire, pouvant sembler à prime abord trop simpliste pour présenter un réel intérêt, produit néanmoins des résultats étonnants : on peut désambïguiser correctement jusqu'à environ 70% des mots polysémiques qui composent un corpus donné. Des techniques apparentées à cette procédure simpliste servent en fait souvent de *base de comparaison* ("baseline"), une borne inférieure pour la performance que les algorithmes plus évolués (exploitant une information plus riche) devront s'assurer de franchir. Et la pratique montre bien que cette simple condition constitue à elle seule un défi de taille...

Le problème de la désambïguisation du sens est difficile en vertu des mêmes incertitudes théoriques formant l'entière toile de fond de cette étude : si des critères clairs et dépourvus d'ambiguïté se dégageaient des multiples formes d'analyse linguistique, il y a longtemps que les problèmes de classification et de traitement seraient chose du passé. Le problème se présente plutôt comme un embrouillamini opaque dans lequel toutes les apparences de régularité semblent se dissoudre.

Il est intéressant de remarquer qu'en dépit des apparences, le problème de la désambiguïsation du sens est très différent du problème de la désambi-

[13] Un corpus dont la construction implique qu'un linguiste ait minutieusement examiné le contexte de chaque mot afin d'en déterminer le sens exact.

guïsation des rôles syntaxiques ("POS tagging") (WILKS 1998). Il n'y a entre autres aucune "nouveauté" dans la distribution des rôles syntaxiques : elle est à peu près fixée une fois pour toutes dans les fondements les plus immuables du langage. Le sens des mots est une matière beaucoup plus friable et volatile, et de nouveaux usages sont sans cesse introduits, alors que d'autres tombent en disgrâce.

C'est donc ici qu'entrent en jeu une famille d'algorithmes tentant de fournir des réponses heuristiques au problèmes de la désambiguïsation du sens. Étant donné que ce sujet touche de très près les idées nouvelles que nous exposerons aux chapitres suivants, il convient ici de faire un rapide tour d'horizon de quelques-unes des méthodes parmi les plus représentatives. Pour une couverture plus exhaustive, on peut consulter (IDE et VÉRONIS 1998).

2.3.1 Quelques distinctions taxonomiques préalables

Il est nécessaire de distinguer préalablement quelques idées et concepts pouvant servir à catégoriser (ou du moins à caractériser) ces différents algorithmes et modèles. On peut d'emblée distinguer deux grandes familles de modèles : les modèles dits *supervisés*, dont l'entraînement a recours à une version étiquetée des exemples (où la "bonne réponse" est fournie, en d'autres termes) et les problèmes *non-supervisés*, où l'entraînement ne peut pas faire usage de cette information. On note que la question de la création de corpus étiquetés servant dans le contexte de la désambiguïsation du sens est particulièrement épineuse : il est très difficile et coûteux de produire de telles documents, et il n'en existe donc par conséquent que très peu. Une autre manière de distinguer les modèles est d'établir s'ils pourront bénéficier ou non d'un ensemble de sens prédéfinis pour les mots de leur vocabulaire. Il n'existe pas en effet une référence canonique établissant de manière absolue et immuable quels sont les différents sens d'un mot particulier. Même les différents dictionnaires ne s'entendent pas de manière parfaitement rigoureuse : un dictionnaire préférera mettre de l'emphase sur un aspect particulier de l'usage d'un mot, tandis qu'un autre préconisera un autre aspect. Les modèles choisissant de travailler avec un

ensemble prédéfini [14] devront donc composer avec cet aspect arbitraire de la définition du sens des mots. Les méthodes qui choisiront, au contraire, de procéder aveuglément, sans avoir recours à aucune prédétermination externe des contraintes du problème, devront bâtir elles-même ces catégories, en s'aidant de méthodes de regroupement en classes afin de tenter de distinguer les usages distincts des mots et d'en établir une catégorisation claire. Ceci implique évidemment d'importantes difficultés : comment déterminer le nombre de sens permis pour un mot, par exemple. Simplement *reconnaître* la polysémie d'un mot particulier ne règle évidemment pas ce problème. Une solution simple est de fixer ce nombre à une valeur constante K, qui sera la même pour tous les mots du vocabulaire. Un algorithme d'apprentissage pourrait également tenter apprendre la valeur optimale pour chacun des mots, en se basant sur un critère de vraisemblance. Une autre difficulté concerne l'*enchevêtrement* (ou le recoupement) des sens associés aux mots d'un vocabulaire : on peut trouver, dans un dictionnaire, plusieurs exemples de synonymes (des mots différents réunis par un sens unique). C'est le cas par exemple des mots "*automobile*" et "*voiture*", qui partagent le sens "*véhicule*". Les modèles de désambiguïsation ont avantage à exploiter cette caractéristique implicite du langage, qui permet la généralisation. Nous reviendrons également sur l'exploitation de cette caractéristique importante.

2.3.2 Méthodes de désambiguïsation non-statistiques

Une première famille d'algorithmes met à profit l'utilisation de dictionnaires ou de thésaurus. Ces techniques exploitent en général le fait que la définitions du sens des mots est elle-même composée de mots (et non d'un quelconque autre ensemble de symboles plus abstraits). Cette constatation suggère l'hypothèse selon laquelle le contenu lexical de la *définition d'un sens* sera en général corrélé au contenu lexical de la *définition des mots* se trouvant dans le voisinage d'un mot w dénotant ce sens. Un algorithme simple consiste à choisir ensuite le sens du mot w qui maximise l'intersection de l'ensemble

[14] Ce qui est évidemment beaucoup plus "rassurant", et présente des avantages dont nous aurons à discuter plus loin.

des mots de sa définition et l'ensemble des mots de la définition des mots de son voisinage (LESK 1986).

Une autre manière ingénieuse d'utiliser un dictionnaire pour déterminer le sens des mots consiste en l'exploitation des différences quant à la polysémie de deux langages (l'anglais et le français par exemple) (DAGAN et ITAI 1994). Les deux sens les plus communs du mot polysémique anglais "*drug*" sont ainsi traduits en français par deux termes bien distincts : "*médicament*" et "*drogue*". Un algorithme capable d'utiliser des caractéristiques étymologiques de ce genre dispose d'un moyen puissant et simple de lever l'ambiguïté de plusieurs mots. Une difficulté notable de cette méthode est le problème de l'*alignement*. On doit aligner dans un premier temps les phrases d'un corpus bilingue, afin de les mettre en correspondance, et aligner ensuite les mots à l'intérieur de ces phrases (ce qui constitue un problème beaucoup plus difficile). Il est possible d'automatiser ces procédures à l'aide de modèles statististiques, mais une précision satisfaisante est très difficile à atteindre (car il n'est pas toujours le cas que de tels alignement existent).

2.3.3 Méthodes de désambiguïsation statistiques

Parmi les méthodes de désambiguïsation se basant sur des algorithmes statistiques, on retrouve celles dont l'entraînement repose sur une classification préalable des exemples d'entraînement. Un représentant simple et intuitif de cette famille d'algorithmes supervisés repose sur le principe de *classification bayesienne* (GALE, CHURCH et YAROWSKY 1992) : pour un mot w_t à la position t, accompagné d'un ensemble de mots C_t (si ces mots sont le contexte de w_t, une possibilité serait par exemple $C_t = \{w_{t-n}, \cdots, w_{t-1}, w_t, w_{t+1}, \cdots, w_{t+n}\}$), on applique la *décision de Bayes* :

$$s^* = \arg\max_s P(s|w_t, C_t), \forall s \in S(w_t) \tag{2.18}$$

où $S(w_t)$ est l'ensemble des sens possibles du mot w_t. En appliquant la règle de Bayes, nous avons :

$$P(s|w_t, C_t) = \frac{P(C_t|s, w_t) \cdot P(s, w_t)}{P(w_t, C_t)} \tag{2.19}$$

où on peut laisser tomber le dénominateur $P(w, C_t)$, qui n'influencera pas le résultat de la classification, et où $P(s)$ est la probabilité d'occurrence à priori du sens s, indépendante de toute information contextuelle. Le calcul de la probabilité conditionnelle d'un contexte C_t est simplifié en posant l'hypothèse fortement réductrice selon laquelle les mots qui le composent ne sont pas corrélés entre eux [15] :

$$P(C_t|s, w_t) = \prod_{w' \in C_t} P(w'|s, w_t) \tag{2.20}$$

où les différents w' correspondent aux mots qui composent le contexte C_t. La probabilité d'un mot de contexte w' particulier est estimée par MLE à l'aide des données d'entraînement :

$$P(w'|s, w_t) = \frac{C(w', s, w_t)}{C(s, w_t)} \tag{2.21}$$

de même que la probabilité jointe d'un sens et d'un mot :

$$P(s, w_t) = \frac{C(s, w_t)}{T} \tag{2.22}$$

Il est possible d'adapter cet algorithme de manière à permettre son usage dans un contexte d'entraînement non-supervisé. Bien qu'il ne soit plus possible alors de déterminer quel est le sens exact d'un mot (en donnant son identité précise tirée d'une table de référence, par exemple), il est néanmoins possible de regrouper de manière optimale les contextes C_t accompagnant les mots ("clustering"). On peut ensuite poser l'hypothèse raisonnable selon laquelle à chacune des classes de contexte correspond un sens différent.

[15] L'hypothèse naïve de Bayes ("Naive Bayes assumption").

2.3.4 Un modèle de désambiguïsation connexionniste : NNWSD

NNWSD est un modèle de désambiguïsation basé sur un réseau de neurones à grande capacité [16]. Ce modèle tente d'exploiter la puissance expressive des réseaux de neurones, en s'aidant encore une fois d'une représentation distribuée, ainsi que d'une fonction objective hybride permettant l'entraînement à l'aide d'exemples supervisés et non-supervisés, ainsi que des exemples *semi*-supervisés (pour lesquels un "degré de confiance" probabiliste est préalablement fourni).

Une incertitude fondamentale en modélisation du langage a trait à la nature de l'"espace sémantique" dans lequel le sens des mots doit être représenté. Une première intuition suggère de construire cet espace à l'aide de la simple occurrence des mots. Un contexte pourrait ainsi être représenté par un vecteur de valeurs booléennes de la taille du vocabulaire, avec des valeurs *vraies* aux positions correspondant aux mots du contexte, et des valeurs *fausses* pour les autres mots. Ce genre de représentation est utilisée entre autres en *recherche d'information* (SALTON et MCGILL 1983) où elle sert à l'encodage et à la comparaison de documents [17]. Un problème avec un tel espace est sa dimensionnalité nécessairement très élevée : il sera difficile de le peupler d'un nombre d'exemples suffisant pour éviter la sous-représentation. Ce problème bien connu est la *malédiction de la dimensionnalité* : plus l'espace de représentation d'un problème de classification est complexe, plus il existe de manières de le résoudre.

Pour aggraver la situation, on peut facilement imaginer des espaces plus complexes dans lesquels la *position* des mots dans la fenêtre de contexte est significative [18] ou encore des espaces générés par les caractéristiques syntaxiques et grammaticales des mots de contexte. Bien qu'elles en augmentent la com-

[16] Ce modèle est également en cours de développement au laboratoire LISA de l'Université de Montréal.

[17] On peut aisément comparer des documents encodés de cette manière à l'aide de métriques euclidiennes, par exemple.

[18] Par opposition à la représentation booléenne précédence, dans laquelle on utilisait un *sac de mots* abolissant toute notion de position.

plexité, ces caractéristiques présentent l'avantage d'injecter dans la représentation un plus grand contenu informatif.

Le modèle connexionniste NNWSD tente de résoudre ce problème en s'aidant d'une représentation distribuée permettant de projeter les structures habitant ces espaces complexes à un sous-espace réduit avec lequel il est plus aisé de construire un mécanisme de discrimination. Cette réduction de dimensionnalité devrait également pouvoir permettre une capacité de généralisation accrue.

2.3.5 Pourquoi faire de la désambiguïsation ?

Le problème de la désambiguïsation du sens n'est en somme pas du tout trivial : il s'agit plutôt d'un problème fondamental, dont la résolution pourrait entraîner une cascade d'avancements dans l'univers du traitement du langage. Les trois principaux champs d'investigation susceptibles de bénéficier le plus de tels avancements sont la **reconnaissance de la parole** ("speech recognition"), la **recherche d'information** ("information retrieval") et la **traduction automatique**. Ces trois problèmes difficiles partagent une même caractéristique, à savoir une grande sensibilité aux variations sémantiques. Pour qu'un système de recherche d'information puisse ainsi extraire un ensemble de documents répondant bien aux besoins de l'usager, il va de soi qu'il doit être en mesure de "comprendre" la requête, ce qui implique nécessairement une composante du mécanisme ayant la capacité de lever les ambiguïtés. De même, un moteur de traduction doit comprendre le texte source à traduire, afin de pouvoir résoudre les incertitudes susceptibles d'influencer grandement la génération, la structure et la signification du texte résultant.

2.4 WordNet, une base de données lexicale

WordNet (FELLBAUM 1998).est une base de données lexicale développée par des linguistes du laboratoire des sciences cognitives de l'Université Princeton. Son but est de répertorier, classifier et relier de diverses manières le

contenu sémantique et lexical de la langue anglaise [19]. Le système se présente sous la forme d'une base de données électronique qu'on peut télécharger [20] sur un système local et y accéder à partir d'un programme à l'aide d'interfaces disponibles pour de nombreux langages de programmation.

La composante atomique sur laquelle repose le système entier est le *synset* ("synonym set"), qui constitue en fait une solution raisonnable au problème de la représentation du sens. Un synset est un groupe de mots interchangeables, dénotant un sens ou un usage particulier. La version 1.7 de WordNet définit ainsi le nom commun anglais "*car*" à l'aide de cinq synsets :

1. `car, auto, automobile, machine, motorcar -- (4-wheeled motor vehicle; usually propelled by an internal combustion engine; "he needs a car to get to work")`
2. `car, railcar, railway car, railroad car -- (a wheeled vehicle adapted to the rails of railroad; "three cars had jumped the rails")`
3. `car, gondola -- (car suspended from an airship and carrying personnel and cargo and power plant)`
4. `car, elevator car -- (where passengers ride up and down; "the car was on the top floor")`
5. `cable car, car -- (a conveyance for passengers or freight on a cable railway; "they took a cable car to the top of the mountain")`

dénotant chacun un sens différent, décrit par une courte définition. Une occurrence particulière du mot "*car*" dénotant par exemple le premier sens, dans le contexte d'une phrase ou d'un énoncé, serait ainsi caractérisée par le fait qu'on pourrait remplacer le mot polysémique par l'un ou l'autre des mots du synset associé **sans altérer la signification de l'ensemble**.

À l'instar d'un dictionnaire traditionnel, WordNet offre ainsi, pour chaque mot, une liste de synsets correspondant à tous ses sens répertoriés. Mais les synsets ont également d'autres usages : ils peuvent représenter des concepts plus abstraits, de plus haut niveau que les mots et leurs sens, qu'on peut organiser sous forme d'*ontologies*. Une ontologie est un système de catégories per-

[19] Des versions de WordNet pour d'autres langues existent, mais la version anglaise est cependant la plus complète à ce jour.

[20] WordNet est distribué avec une licence spéciale très libérale, permettant de l'utiliser commercialement ou à des fins de recherche.

mettant de classifier les éléments d'un univers. Les systèmes de catégorisation qui nous intéressent correspondent aux différentes relations sémantiques avec lesquelles il est possible de regrouper de manière cohérente les composantes d'un univers linguistique (les mots, les sens et les concepts par exemple). La relation sémantique servant de critère pour l'agrégation d'un groupe de concepts définira le *type* de l'ontologie. WordNet définit et répertorie ainsi une grande variété de relations sémantiques permettant d'organiser le sens des mots (et donc par extension les mots eux-mêmes) en des systèmes de catégories qu'on peut consulter de manière cohérente et uniforme. On pourra ainsi interroger le système quant aux *hypernymes* d'un mot particulier. La relation d'hypernymie, communément nommée "relation EST-UN" ("IS-A") se rapporte à la *généralisation* des concepts. À partir par exemple du sens le plus commun du nom "*car*" (le sens "*car, auto...*") la relation d'hypernymie définit un "arbre" [21] de concepts de plus en plus généraux :

```
1. car, auto, automobile, machine, motorcar
    => motor vehicle, automotive vehicle
        => vehicle
            => conveyance, transport
                => instrumentality, instrumentation
                    => artifact, artefact
                        => object, physical object
                            => entity, something
```

Il est clair que le dernier concept, "*entity, something*", est le plus général, le plus abstrait (il pourrait ainsi être le super-concept d'une multitude de concepts plus spécialisés). La figure 2.1 schématise de manière simplifiée l'hypernymie du mot polysémique "*bank*" [22].

On peut également interroger le système quant à la relation inverse de l'hypernymie, l'*hyponymie*, soit la *spécialisation* des concepts. WordNet offre en fait une multitude d'autres d'ontologies, faisant usage de relations sémantiques plus spécialisées et restrictives. On peut ainsi interroger le système quant aux

[21] Il s'agit en fait d'un "quasi-arbre" : certains concepts peuvent hériter de *plusieurs* concepts plus généraux, ce qui fait de la structure hiérarchique un graphe acyclique (DAG).

[22] Fort probablement l'exemple de mot ambigu le plus couramment cité...

méronymes d'un mot ou d'un concept, les parties constitutives d'un objet ("HAS-PART"). Les méronymes associés au sens "*car, auto...*" du mot "*car*" sont :

```
1. car, auto, automobile, machine, motorcar
     HAS PART: accelerator, accelerator pedal, gas pedal, gas,
               throttle, gun
     HAS PART: air bag
     HAS PART: auto accessory
     HAS PART: automobile engine
     HAS PART: automobile horn, car horn, motor horn, horn
     (...)
```

On peut aussi consulter le système quant à la relation inverse, l'*holonymie*, ou encore pour les relations de *synonymie* et d'*antonymie*.

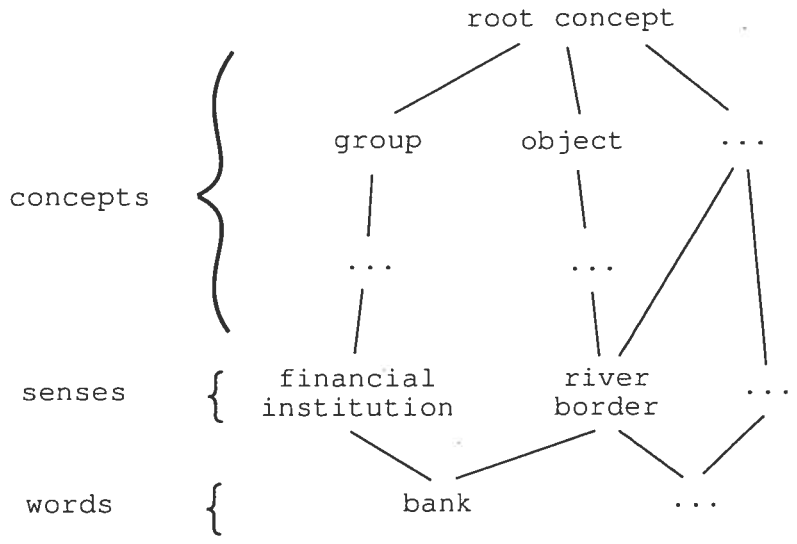


Figure 2.1 – Un exemple simplifié illustrant la relation "EST-UN" de WordNet à l'aide du mot polysémique anglais "*bank*".

Bien que WordNet soit un système d'une étonnante ampleur (la version la plus récente, 2.0, répertorie environ 150 000 mots de *classe ouverte* [23] ainsi qu'environ 200 000 sens) son statut de projet "en développement" implique que

[23] Une classe ouverte est une classe de mots pour laquelle l'ajout de nouveaux éléments

certaines de ses composantes sont incomplètes. À chaque nouvelle version, le lexique s'enrichit de nouveaux mots, et des configurations sémantiques sont ajoutées, modifiées, ou encore rendues désuètes. Si on examine par exemple l'ontologie générée par la relation d'hypernymie (celle dont nous ferons le plus usage dans la présente étude), il est notable qu'elle est la plus complète dans son embranchement nominal [24]. Les noms sont ainsi classés en un système de catégories complet et précis comprenant plusieurs niveaux d'imbrication [25]. On retrouve en revanche un système de classification beaucoup moins élaboré pour les verbes, qui sont organisés en un système hiérarchique beaucoup plus "plat" (moins de niveaux d'imbrication), où on passe très rapidement d'un concept spécialisé (le sens "*operate, run*" du verbe "*running*", par exemple) à un concept général ("*control, command*"). Et il n'y a finalement à ce jour aucune catégorisation hiérarchique définie pour les embranchements des adjectifs et des adverbes. Ce déséquilibre potentiellement problématique se retrouve à l'intérieur même des super-catégories, où il est évidemment beaucoup plus apparent dans la branche nominale : certains mots sont ainsi liés à une grande chaîne de concepts finement graduée, tandis que d'autres sont très proche des concepts les plus généraux.

2.4.1 Pourquoi utiliser WordNet ?

WordNet jouit d'une énorme et grandissante popularité au sein de la communauté scientifique, et joue également un rôle important dans plusieurs projets commerciaux. Sa richesse et sa précision en font un outil de choix, susceptible d'être mis à profit par une multitude de techniques et de théories diverses. Son utilisation fait en sorte de procurer aux algorithmes et applications une importante plate-forme de connaissances à priori du langage et

au lexique est relativement fréquente. On remarque que la classes nominale est clairement ouverte, tandis ques les classes des verbes et des adjectifs le sont moins. Celle des pronoms est par contre complètement fermée.

[24] Le lexique de WordNet est séparé en quatre grandes super-catégories lexicales : les noms, les verbes, les adjectifs et les adverbes.

[25] On retrouve notamment certaines sections de cette ontologie où la profondeur dépasse 10 niveaux.

du monde dans lequel il s'articule. Un exemple particulièrement représentatif et ingénieux de son utilisation est donné par les métriques heuristiques de *distance sémantique* (BUDANITSKY 2000) entre les concepts d'une ontologie particulière, basées sur la distance à parcourir dans le graphe. Cette distance peut permettre de quantifier par exemple la *similarité* de deux concepts. Elle peut également servir à faire de la désambiguïsation (AGIRRE et RIGAU 1996; DIAB et RESNIK 2002).

Chapitre 3

Des modèles du second ordre lissés avec WordNet

Ce chapitre et le suivant seront consacrés à la discusion de certaines idées offrant de nouvelles perspectives en modélisation du langage. Ces idées ont été développées et testées au laboratoire LISA de l'Université de Montréal, et les résultats de certaines expériences s'y rapportant seront également exposés.

3.1 Quelques mots sur les modèles graphiques

Le trait commun à ces nouvelles méthodes est l'utilisation de *modèles graphiques* (JORDAN 1998) et de *variables cachées* pour la modélisation de phénomènes pour lesquels il n'est pas possible (ou du moins très difficile) d'obtenir ou de fabriquer des ensembles d'entraînement supervisés. Les modèles graphiques sont un formalisme puissant et général basé sur un mariage entre la théorie des probabilités et la théorie des graphes. Ils permettent entre autres la représentation de systèmes complexes de manière modulaire et intuitive, en exposant clairement leur dynamique interne. Ils permettent également de faire le pont entre divers formalismes apparentés [1] qui en constituent des cas spéciaux.

[1] Citons l'exemple des HMM ("Hidden Markov Models").

Les noeuds d'un modèle graphique représentent les variables aléatoires décrivant les états possibles du système, tandis que les arcs qui les relient représentent les dépendances conditionnelles qui régissent leur distribution [2].

La modélisation graphique d'un problème fait en sorte d'attirer l'attention sur la structure de dépendance conditionnelle des variables qui le composent, ce qui est susceptible d'offrir un avantage important : la représentation de la distribution jointe de la totalité ou d'un sous-ensemble des variables aléatoires d'un système peut s'exprimer de manière plus compacte en exploitant la réduction offerte par l'indépendance conditionnelle explicite de certaines relations.

Les modèles graphiques simples dont nous faisons usage dans un contexte de modélisation du langage ne comportent que des arcs dirigés (les modèles présentant cette caractéristique sont aussi appelés *réseaux bayésiens*) ainsi que des variables aléatoires discrètes (représentant l'occurrence des concepts, des sens ou des mots, des phénomènes discrets par nature).

Avant de pouvoir faire de l'*inférence* à l'aide d'un modèle graphique (répondre à une question, étant donné une observation, par exemple) on doit tout d'abord s'attaquer à deux grands problèmes qui les concernent : le calcul des paramètres qui régissent les distributions conditionnelles des différentes variables aléatoires et la détermination de la structure du modèle, soit l'architecture qui définit les dépendances conditionnelles entre les variables aléatoires. Le premier problème est un problème d'optimisation, et on peut donc le résoudre à l'aide d'algorithmes se basant sur le principe de maximum de vraisemblance, en cherchant la paramétrisation du modèle qui "explique" le mieux un ensemble d'observations.

Dans le cadre de cette étude, nous n'aurons pas à nous attaquer au problème plus difficile de la détermination et de l'optimisation de l'architecture du modèle, qui implique souvent une recherche coûteuse dans l'espace des configurations possibles. À l'instar de nombreuses autres méthodes en traitement du langage, nous tenterons plutôt de mettre à profit des hypothèses structu-

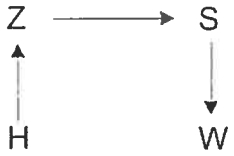
[2] Ceci laisse naturellement entendre que l'*absence* d'arc reliant deux variables implique l'indépendance conditionnelle de ces dernières.

rales découlant d'informations et d'intuitions à priori propres aux problèmes particuliers qne nous tentons de résoudre.

3.2 Un modèle bigramme lissé avec le sens des mots

Étant donné que nous nous proposons d'en étudier des variantes, faisons tout d'abord remarquer que la représentation d'un modèle bigramme à l'aide d'un modèle graphique est naturelle :

où l'unique arc représente la dépendance conditionnelle régissant la distribution de la variable aléatoire W, associée au mot courant, qui dépend de la valeur de la variable aléatoire H, représentant le mot précédent. Cette représentation graphique, bien que triviale, a tout de même le mérite de suggérer aisément un modèle légèrement plus complexe :



où les variables aléatoires S et Z peuvent prendre les valeurs des sens associés à des mots w et h. Ces variables de sens sont dites *cachées* car il ne sera pas possible de faire des observations permettant d'estimer directement les distributions les impliquant. Ceci constitue la caractéristique fondamentale de ce modèle, que nous nous proposons maintenant d'étudier en détail.

En général, l'ajout de variables cachées dans un modèle probabiliste peut s'interpréter de deux manières différentes. Elle peut premièrement servir à postuler l'existence de phénomènes réels, ayant une influence concrète, mais pour lesquels il n'est pas possible d'effectuer des observations. Elle peut également servir à ajouter une articulation supplémentaire au modèle, sans pour

autant dénoter l'existence d'un phénomène concret se rapportant à la réalité ou à la structure du système que l'on tente de modéliser. Le modèle que nous étudions se situerait ainsi probablement à mi-chemin entre ces deux conceptions. La discussion du premier chapitre a en effet proposé des idées selon lesquelles le sens ne se rapporterait pas nécessairement à une entité tangible, "accompagnant" les mots et le langage. En ajoutant ainsi des variables dont le but est de représenter des entitées discrètes accompagnant l'occurrence des mots (ce que l'architecture du modèle suggère clairement), il n'est donc pas du tout assuré que l'on capte "quelque chose" se trouvant dans la réalité, mais se dérobant en quelque sorte à notre obervation. L'espoir est plutôt que cette articulation supplémentaire permette d'enrichir l'expressivité du modèle, qu'elle lui permette de capter certaines régularités sémantiques plus profondes. Notons que ces considérations abstraites ne se rapportent qu'à l'*interprétation* du modèle, et qu'elles ne concernent donc nullement ses aspects techniques et mécaniques.

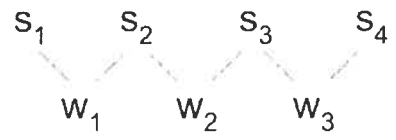
Afin d'étudier cette idée en détail, définissons tout d'abord un univers linguistique restreint et artificiel dans lequel nous ferons évoluer un modèle-jouet [3] qui servira à illustrer clairement les concepts et les calculs en jeu. Soit tout d'abord le vocabulaire :

$$V = \{w_1, w_2, w_3\} \tag{3.1}$$

et soit ensuite un ensemble de sens :

$$S = \{s_1, s_2, s_3, s_4\} \tag{3.2}$$

Définissons ensuite une ontologie simple, associant à chaque mot du vocabulaire un ensemble de sens pouvant leur être attribués :



[3] Ce modèle-jouet a été développé dans l'environnement de calcul scientifique Matlab.

Et définissons finalement un minuscule corpus d'entraînement de $T = 10$ mots :

$$\mathcal{T} = \{w_1, w_2, w_1, w_3, w_2, w_3, w_3, w_1, w_2, w_1\} \tag{3.3}$$

Pour des fins de comparaison, considérons tout d'abord la table de fréquence empirique du bigramme, uniquement basée sur les événements observés dans le corpus d'entraînement :

$C_{bigramme}(W, H)$	w_1	w_2	w_3
w_1	0	2	1
w_2	2	0	1
w_3	1	1	1

Tableau 3.1 – Fréquence des événements sur le corpus $\mathcal{T}$.

Nous pouvons ensuite en tirer une table de probabilité conditionnelle (TPC), en la normalisant de manière à ce que la somme des éléments de chaque colonne soit 1 :

$P_{bigramme}(W\|H)$	w_1	w_2	w_3
w_1	0	$\frac{2}{3}$	$\frac{1}{3}$
w_2	$\frac{2}{3}$	0	$\frac{1}{3}$
w_3	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$

Tableau 3.2 – Probabilité conditionnelle des événements sur le corpus $\mathcal{T}$.

Les cellules de ces tables correspondent à des transitions possibles dans l'espace du bigramme. On remarque que certaines de ces transitions ont une probabilité nulle. C'est le cas notamment de l'événement $\{H = w_1, W = w_1\}$ (deux occurrence consécutives du mot w_1) qu'il n'est pas possible d'observer dans $\mathcal{T}$. Il est important de distinguer les deux significations possibles de cette "absence" : elle peut signifier que l'événement n'est pas plausible (il n'est pas grammaticalement ou sémantiquement possible, par exemple) ou encore que l'événement n'a tout simplement pas été rencontré, étant donné la taille possiblement trop restreinte du corpus d'entraînement (ce qui est certainement le cas ici !). Nous avons vu au chapitre précédent comment les techniques de

lissage pouvaient résoudre en partie ce problème, à l'aide par exemple d'une simple mixture de modèles d'ordre inférieur. L'extension au modèle bigramme que nous proposons offre en contrepartie un lissage implicite, de par son architecture même : les variables aléatoires modélisant le sens vont faire en sorte de *répartir* la masse de probabilité sur certains événements n'ayant pas été rencontrés lors de l'entraînement, en exploitant le fait qu'à **certains mots sont associés plus d'un sens, et vice-versa**.

3.2.1 L'évaluation du modèle (à l'aide des valeurs courantes des paramètres)

La probabilité d'occurrence d'un mot w_t étant donné l'occurrence d'un mot précédent w_{t-1} est obtenue en considérant tous les "chemins" possibles reliant ces deux mots dans l'ontologie (on nomme cette phase du calcul la *propagation avant*) :

$$P_{HZSW}(w_t|w_{t-1}) = \sum_{s,z} P(w_t|s) \cdot P(s|z) \cdot P(z|w_{t-1}) \tag{3.4}$$

où on doit sommer sur toutes les valeurs possibles de S et Z car on ne dispose d'aucune valeur observée pour ces variables. Cette première équation expose clairement les différents paramètres du modèle, qu'on représentera encore une fois sous forme de tables : $P(W|S)$, la probabilité qu'un mot ait été "généré" [4] par un sens, $P(S|Z)$, la probabilité de transition d'un sens à un autre, et $P(Z|H)$, la probabilité qu'un mot "dénote" un sens [5]. Bien que ce ne soit pas très apparent ici, en raison de la petite taille du problème, remarquons tout d'abord que la densité des tables $P(W|S)$ et $P(Z|H)$ est par nature très faible. L'utilisation d'une ontologie (celle offerte par WordNet par exemple) déterminant à quels sens sont associés les mots fait en sorte de contenir de

[4] On utilisera une convention selon laquelle un sens *génère* un mot, et un mot *dénote* un sens.

[5] Il est à noter que $P(W|S)$ et $P(Z|H)$, étant simplement les inverses l'un de l'autre, pourraient ne constituer qu'une seule et même table paramètre, que l'on pourrait transformer à l'aide de la loi de Bayes. Pour des raisons de clarté, nous allons cependant les découpler de manière explicite tout au long de l'exposé.

manière importante l'explosion combinatoire qui résulterait de l'absence d'une telle contrainte. Toutefois, il est à noter que la table $P(S|Z)$ est beaucoup plus dense qu'une table de transition pour les mots.

Nous sommes maintenant en mesure de calculer la probabilité des événements qui composent notre petit corpus d'entraînement. Mais un premier calcul nécessitant bien entendu des valeurs préalables "raisonnables" pour les paramètres, nous devrons tout d'abord les initialiser. Étant donné qu'une partie des événements qui nous intéressent ne sont pas observés, nous allons devoir nous contenter d'un processus d'initialisation heuristique, permettant de diriger le modèle sur une voie de départ raisonnable. Cette procédure consiste en un parcours préalable du corpus d'entraînement, afin de compiler la fréquence des différents événements qu'on y retrouve. Lorsqu'aucune observation n'est disponible pour un événement particulier (ou encore lorsqu'une observation est incomplète), on se contentera d'accumuler l'ensemble des valeurs possibles. Étant donné que certains événements seront représentés plus fréquemment que d'autres dans le corpus d'entraînement, une distribution approximative mais raisonnable devrait se dégager à l'issue de cette première étape. Les tables suivantes montrent le résultat de l'initialisation des trois tables de paramètres du modèle (la fréquence et la probabilité conditionnelle des événements sont rapportées à l'aide de la notation "$c : p$") :

$C(W,S) : P(W\|S)$	s_1	s_2	s_3	s_4
w_1	3 : 1	3 : $\frac{1}{2}$	0 : 0	0 : 0
w_2	0 : 0	3 : $\frac{1}{2}$	3 : $\frac{1}{2}$	0 : 0
w_3	0 : 0	0 : 0	3 : $\frac{1}{2}$	3 : 1

Tableau 3.3 – Fréquence et probabilité d'un sens générant un mot sur le corpus $\mathcal{T}$.

$C(S,Z) : P(S\|Z)$	s_1	s_2	s_3	s_4
s_1	$0:0$	$2:\frac{1}{6}$	$3:\frac{1}{4}$	$1:\frac{1}{6}$
s_2	$2:\frac{1}{3}$	$4:\frac{1}{3}$	$4:\frac{1}{3}$	$2:\frac{1}{3}$
s_3	$3:\frac{1}{2}$	$4:\frac{1}{3}$	$3:\frac{1}{4}$	$2:\frac{1}{3}$
s_4	$1:\frac{1}{6}$	$2:\frac{1}{6}$	$2:\frac{1}{6}$	$1:\frac{1}{6}$

Tableau 3.4 – Fréquence et probabilité d'un sens suivi d'un autre sens sur le corpus $\mathcal{T}$.

$C(Z,H) : P(Z\|H)$	w_1	w_2	w_3
s_1	$3:\frac{1}{2}$	$0:0$	$0:0$
s_2	$3:\frac{1}{2}$	$3:\frac{1}{2}$	$0:0$
s_3	$0:0$	$3:\frac{1}{2}$	$3:\frac{1}{2}$
s_4	$0:0$	$0:0$	$3:\frac{1}{2}$

Tableau 3.5 – Fréquence et probabilité d'un mot dénotant un sens sur le corpus $\mathcal{T}$.

Ces paramètres nous permettent de calculer une table de toutes les transitions possibles (qui comprend entre autres la probabilité d'événements ne faisant pas partie du corpus d'entraînement) :

$P_{HZSW}(W\|H)$	w_1	w_2	w_3
w_1	0.25	0.375	0.375
w_2	0.375	0.3125	0.3125
w_3	0.375	0.3125	0.3125

Tableau 3.6 – Probabilité de transition d'un mot à un autre sur le corpus $\mathcal{T}$, selon le modèle $HZSW$.

qu'on peut comparer à la table du bigramme (tableau 3.2) afin de constater le lissage inhérent qu'offre notre modèle : alors que la table du bigramme comporte des "trous" (des cellule associées à des événements dont la probabilité est nulle), la table de notre modèle n'en comporte aucun, toutes les possibilités étant couvertes. L'ajout des variables de sens crée donc des chemins supplémentaires entre les mots, en relâchant la contrainte selon laquelle un événement particulier doit avoir été observé tel quel pour se voir attribuer une probabilité non-nulle. La probabilité de la séquence non-observée $w_1 w_1$

est ainsi non-nulle, car la séquence de sens $s_1 s_2$, susceptible de la générer (ou de l'"expliquer"), s'est vue attribuer une probabilité non-nulle par l'observation d'autres événements (w_1 suivi de w_2, par exemple). Notons au passage que ce n'est pas le cas de la séquence de sens $s_1 s_1$, susceptible elle-aussi de générer $w_1 w_1$, mais dont l'estimation de la probabilité n'a pas pu être dérivée par aucune observation.

Bien que ce constat préliminaire puisse paraître encourageant, il reste cependant à vérifier la capacité du modèle à "expliquer" le corpus d'entraînement dans son ensemble, et non pas seulement ses composantes individuelles. Pour ce faire, considérons une fonction objective simple, la probabilité jointe des événements qui composent le corpus, que l'on voudra maximiser :

$$P(\mathcal{T}) = P(w_1^T) = \prod_t P(w_t|w_{t-1}) \tag{3.5}$$

et évaluons-la afin de comparer les performances du bigramme et de notre modèle :

$$\begin{aligned} P(\mathcal{T})_{bigramme} &= P_{bigramme}(w_1^T) \approx 0.0008 \\ P(\mathcal{T})_{HZSW} &= P_{HZSW}(w_1^T) \approx 0.00008 \end{aligned}$$

Étant donné que notre modèle répartit la masse de probabilité de manière plus uniforme et lisse, ce résultat n'est guère surprenant : la masse y est "diluée" en un plus grand nombre d'événements, faisant en sorte évidemment de diminuer la qualité de l'explication de la séquence précise du corpus d'entraînement. Le modèle bigramme est ainsi plus "proche" des données d'entraînement que l'est notre modèle (la séquence complète $\mathcal{T}$ est 10 fois plus probable). On doit toutefois prendre en considération le fait qu'il s'agit d'une métrique rapportant la performance à l'*entraînement*. Étant donné que notre problème en est un d'*apprentissage*, et non de pure optimisation, la métrique qui nous intéresse vraiment est la performance sur un corpus de test avec lequel le modèle n'aura jamais été en contact, et qui mesurera donc sa capacité à généraliser. Une performance sub-optimale sur l'ensemble d'entraînement est en fait souvent

garante d'une performance adéquate sur un ensemble de test.

Mais il reste toutefois à ajuster les paramètres du modèle, car la paramétrisation initiale avec laquelle nous l'avons évalué n'est probablement pas optimale.

3.2.2 Réestimation des paramètres à l'aide de l'algorithme EM

Un modèle de langage sans variable cachée optimise la fonction de vraisemblance de manière déterministe : étant donné que toutes les observations sont disponibles, une fois le corpus parcouru, toutes les variables aléatoires du modèle sont rigidifiées aux valeurs particulières des événements rencontrés. On peut ainsi montrer que la fonction de vraisemblance est maximisée avec l'estimateur obtenu en résolvant :

$$\frac{\partial E}{\partial P(y|x)} = 0 \longrightarrow P_{MLE}(y|x) = \frac{C(x,y)}{C(x)} \tag{3.6}$$

où E est la fonction de coût, x et y sont des observations, et $P(x|y)$ est le paramètre qui nous intéresse, et qui correspond bel et bien à l'estimateur MLE suggéré par l'intuition.

Ce paradigme d'optimisation simple n'est cependant pas applicable au problème qui nous intéresse, car il ne sera pas possible de compiler les valeurs des variables cachées de notre modèle. Une manière simple de comprendre comment l'algorithme EM (Dempster, Laird et Rubin 1977; Bilmes 1997) solutionne ce problème est de considérer qu'il permet le remplacement de ces compilations ($C(\cdot)$) par des estimés raisonnables ($E[C(\cdot)]$), qu'on améliorera ensuite de manière itérative.

De manière plus générale, l'algorithme EM cherche à optimiser la vraisemblance d'un modèle dans lequel certaines variables sont observées et certaines autres sont cachées. Les variables cachées représentent souvent des phénomènes dont on pose l'hypothèse qu'ils sont la *cause* des phénomènes de surface associés aux variables observées. Dans la première des deux étapes de l'algorithme ("E-step"), on calcule l'espérance du compte des événements aléatoires

à partir de l'estimation courante des paramètres du modèle. Pour l'événement d'un sens s générant un mot w, nous avons :

$$\begin{aligned} E[\#\ de\ fois\ où\ s\ génère\ w|\mathcal{T}] &= E[C(w,s)|\mathcal{T}] \\ &= \sum_t P(w_t = w, s_t = s|w_t, h_t) \\ &= \sum_t \mathcal{I}_{\{w_t=w\}} \cdot P(s_t = s|w_t, h_t) \quad (3.7) \end{aligned}$$

Cette espérance conditionnelle est donc une simple sommation de la probabilité postérieure de l'événement, étant donné les différentes observations effectuées à la position t dans le corpus [6]. Étant donné que W n'est pas une variable cachée, on remarque que sa probabilité postérieure est remplacée à la troisième ligne par une fonction indicatrice (car $P(X = x|X = x) = 1$). La sommation des valeurs de la fonction indicatrice joue donc le rôle d'un simple compteur dont on pondérera la valeur par la probabilité postérieure de $S = s$, qui s'obtient aisément par la loi de Bayes :

$$\begin{aligned} P(s|w,h) &= \frac{P(w|s,h) \cdot P(s|h)}{P(w|h)} \\ &= \frac{P(w|s) \cdot P(s|h)}{P(w|h)} \end{aligned} \quad (3.8)$$

où la simplification de la deuxième étape ($P(w|s,h)$ qui devient $P(w|s)$) est rendue possible par la structure du modèle, qui assure l'indépendance conditionnelle des variables W et H (explicitement représentée par le fait qu'il n'y a pas d'arc reliant les deux noeuds du graphe correspondant à ces variables). On remarquera également qu'on peut réutiliser les calculs de la phase de propagation avant pour le calcul de la probabilité postérieure. On évalue de manière similaire l'espérance des comptes des autres événements :

[6] Il est à noter que $h_t = w_{t-1}$, de même que $z_t = s_{t-1}$.

$$
\begin{aligned}
E[\#\ de\ fois\ où\ s\ suit\ z|\mathcal{T}] &= E[C(s,z)|\mathcal{T}] \\
&= \sum_t P(s_t = s, z_t = z|w_t, h_t) \\
E[\#\ de\ fois\ où\ h\ dénote\ z|\mathcal{T}] &= E[C(z,h)|\mathcal{T}] \\
&= \sum_t P(z_t = z, h_t = h|w_t, h_t) \\
&= \sum_t P(z_t = z|w_t, h_t) \cdot \mathcal{I}_{\{h_t=h\}} \qquad (3.9)
\end{aligned}
$$

où les probabilités postérieures s'obtiennent par quelques manipulations simples, en mettant encore une fois à profit les réductions rendues possibles par l'indépendance conditionnelle de certains couples de variables :

$$
\begin{aligned}
P(s,z|w,h) &= P(z|s,w,h) \cdot P(s|w,h) \\
&= \frac{P(s|z) \cdot P(z|h)}{P(s|h)} \cdot P(s|w,h) \qquad (3.10)
\end{aligned}
$$

où $P(s|w,h)$ a été précédemment calculé avec (3.8), et finalement :

$$
P(z|w,h) = \sum_s P(s,z|w,h) \qquad (3.11)
$$

La deuxième étape de l'algorithme ("M-step") maximise la vraisemblance des données observées en dérivant de nouveaux estimateurs pour les paramètres, obtenus à partir des espérances calculées à la première étape :

$$
\begin{aligned}
P(w|s) &\leftarrow \frac{E[\#\ de\ fois\ où\ s\ génère\ w|\mathcal{T}]}{\sum_{w'} E[\#\ de\ fois\ où\ s\ génère\ w'|\mathcal{T}]} \\
&\leftarrow \frac{\sum_t \mathcal{I}_{\{w_t=w\}} \cdot P(s_t = s|w_t, h_t)}{\sum_t \sum_{w'} \mathcal{I}_{\{w_t=w'\}} \cdot P(s_t = s|w_t, h_t)} \qquad (3.12) \\
P(s|z) &\leftarrow \frac{E[\#\ de\ fois\ où\ s\ suit\ z|\mathcal{T}]}{\sum_{s'} E[\#\ de\ fois\ où\ s'\ suit\ z|\mathcal{T}]} \\
&\leftarrow \frac{\sum_t P(s_t = s, z_t = z|w_t, h_t)}{\sum_t \sum_{s'} P(s_t = s', z_t = z|w_t, h_t)} \qquad (3.13) \\
P(z|h) &\leftarrow \frac{E[\#\ de\ fois\ où\ h\ dénote\ z|\mathcal{T}]}{\sum_{z'} E[\#\ de\ fois\ où\ h\ dénote\ z'|\mathcal{T}]} \\
&\leftarrow \frac{\sum_t P(z_t = z|w_t, h_t) \cdot \mathcal{I}_{\{h_t=h\}}}{\sum_t \sum_{z'} P(z_t = z'|w_t, h_t) \cdot \mathcal{I}_{\{h_t=h\}}} \qquad (3.14)
\end{aligned}
$$

On peut montrer que l'alternance itérative des deux étapes fera converger la vraisemblance des données observées du modèle, ce qui constitue l'objectif de l'optimisation d'un modèle de langage statistique, qu'il contienne des variables cachées ou non.

3.2.3 Analyse détaillée de l'entraînement

Examinons maintenant plus attentivement ce qu'il advient des paramètres de notre modèle-jouet à l'étude, lorsque l'on tourne la manivelle de cette procédure. Après la première itération, les trois TPC deviennent :

$P(W\|S)$	s_1	s_2	s_3	s_4
w_1	1 : –	0.4839 :↓	0 : –	0 : –
w_1	0 : –	0.5161 :↑	0.5145 :↑	0 : –
w_1	0 : –	0 : –	0.4855 :↓	1 : –

Tableau 3.7 – Probabilité réestimée qu'un sens génère un mot sur le corpus $\mathcal{T}$.

$P(S\|Z)$	s_1	s_2	s_3	s_4
s_1	$0:-$	$0.1613:\downarrow$	$0.3261:\uparrow$	$0.1471:\downarrow$
s_2	$0.2667:\downarrow$	$0.3226:\downarrow$	$0.3043:\downarrow$	$0.3235:\downarrow$
s_3	$0.6:\uparrow$	$0.3387:\downarrow$	$0.1957:\downarrow$	$0.3529:\uparrow$
s_4	$0.1333:\downarrow$	$0.1774:\uparrow$	$0.1739:\uparrow$	$0.1765:\uparrow$

Tableau 3.8 – Probabilité réestimée de la transition d'un sens à un autre sur le corpus $\mathcal{T}$.

$P(Z\|H)$	w_1	w_2	w_3
s_1	$0.5556:\uparrow$	$0:-$	$0:-$
s_2	$0.4444:\downarrow$	$0.4741:\downarrow$	$0:-$
s_3	$0:-$	$0.5259:\uparrow$	$0.4963:\downarrow$
s_4	$0:-$	$0:-$	$0.5037:\uparrow$

Tableau 3.9 – Probabilité réestimée qu'un mot dénote un sens sur le corpus $\mathcal{T}$.

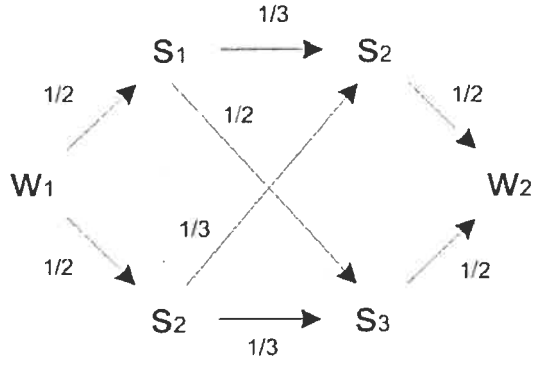
où les flèches indiquent l'augmentation ou la diminution de la masse de probabilité attribuée à un événement particulier, en se référant à l'état initial des paramètres (tableaux 3.3, 3.4 et 3.5). On constate que l'entraînement a modifié les paramètres du modèle de manière significative, mais les raisons expliquant ces changements peuvent apparaître obscures, à première vue. Examinons donc de plus près la dynamique de l'une de ces modifications, pour tenter d'y voir plus clair. Si on considère la table $P(W|S)$ originale (tableau 3.3, dont les valeurs ont été calculées lors de l'initialisation) les deux mots que peut générer le sens s_2 sont équiprobables : $P(w_1|s_2) = \frac{1}{2}$ et $P(w_2|s_2) = \frac{1}{2}$. Sans à priori, une distribution uniforme des sens est en effet une hypothèse raisonnable. Pourtant, si on examine les valeurs de la table après la première itération, on constate que la probabilité du deuxième mot est maintenant légèrement plus élevée que celle du premier : $P(w_2|s_2) \approx 0.52$ tandis que $P(w_1|s_2) \approx 0.48$. Comment le modèle peut-il se prononcer de manière aussi décisive sur la probabilité d'événements faisant intervenir le sens, alors qu'il n'a pu observer que l'occurrence seule des mots ? La réponse a trait à la maximisation de la *cohérence* des chaînes d'événements qui composent le modèle : les paramètres que nous examinons ne sont pas une composante isolée du système, n'ayant

aucune influence sur le reste. Il s'agit plutôt d'un maillon imbriqué dans une chaîne, et dont la dynamique influence la totalité du système.

Voyons maintenant ce qu'il advient des paramètres qui nous intéressent pendant cette première itération d'entraînement. Étant donné qu'on examine la probabilité du fait que le sens s_2 ait généré un mot quelconque, il est nécessaire d'examiner seulement quatre événements distincts du corpus d'entraînement : w_1w_2 (à la position $t = 1$ et $t = 8$), w_3w_2 ($t = 4$), w_2w_1 ($t = 1$ et $t = 9$) et w_3w_1 ($t = 7$), qui partagent tous la même caractéristique : le dernier maillon de leur chaîne est $s_2 \rightarrow \{w_1, w_2\}$ [7]. La formule de réestimation de $P(w|s_2)$, pour $w \in \{w_1, w_2\}$, est :

$$P(w|s_2) \leftarrow \frac{\sum_t \mathcal{I}_{\{w_t=w\}} \cdot P(s_t = s_2|w_t, h_t)}{\sum_t \sum_{w'} \mathcal{I}_{\{w_t=w'\}} \cdot P(s_t = s_2|w_t, h_t)} \tag{3.15}$$

Nous devons maintenant calculer la probabilité postérieure des quatre événements (six en comptant les répétitions) qu'implique cette formule (ce qui correspond au "E-step"). Voici tout d'abord une représentation graphique du premier événement, montrant tous les chemins possibles le composant :



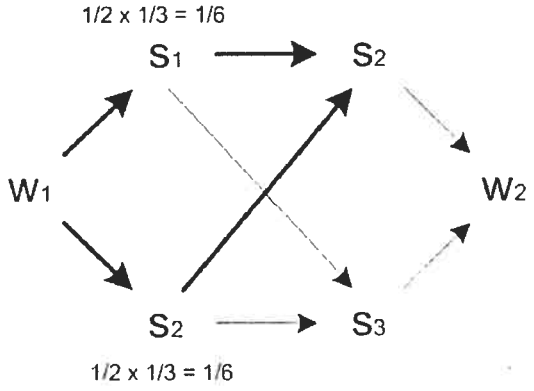
On utilise (3.8) pour calculer la probabilité postérieure qui nous intéresse :

$$P(s_2|w_1, w_2) = \frac{P(w_2|s_2) \cdot P(s_2|w_1)}{P(w_2|w_1)} = \frac{0.5 \times 0.\bar{3}}{0.375} = 0.\bar{4}$$

où $P(w_2|s_2)$ provient directement des paramètres courants, $P(s_2|w_1)$ est la

[7] L'ontologie de notre modèle ne permet en effet que les transitions $s_2 \rightarrow w_1$ et $s_2 \rightarrow w_2$.

sommation de la probabilité de tous les chemins menant du mot w_1 au sens s_2 :



et $P(w_2|w_1)$ est la sommation de la probabilité de tous les chemins menant du mot w_1 au mot w_2. On répète le calcul pour les trois autres probabilités postérieures qui nous intéressent :

$$\begin{aligned} P(s_2|w_3,w_2) &= \frac{P(w_2|s_2)\cdot P(s_2|w_3)}{P(w_2|w_3)} = \frac{0.5\times 0.\bar{3}}{0.3125} = 0.5\bar{3} \\ P(s_2|w_2,w_1) &= \frac{P(w_1|s_2)\cdot P(s_2|w_2)}{P(w_1|w_2)} = \frac{0.5\times 0.\bar{3}}{0.375} = 0.\bar{4} \\ P(s_2|w_3,w_1) &= \frac{P(w_1|s_2)\cdot P(s_2|w_3)}{P(w_1|w_3)} = \frac{0.5\times 0.\bar{3}}{0.375} = 0.\bar{4} \end{aligned}$$

et nous avons maintenant tout ce qu'il nous faut pour réestimer la distribution des mots pouvant être générés par le sens s_2, en évaluant (3.15) pour $w \in \{w_1, w_2\}$ (ce qui correspond au "M-step") :

$$\begin{aligned} P(w_1|s_2) &\leftarrow \frac{3\times 0.\bar{4}}{(5\times 0.\bar{4})+0.5\bar{3}} \approx 0.48 \\ P(w_2|s_2) &\leftarrow \frac{(2\times 0.\bar{4})+0.5\bar{3}}{(5\times 0.\bar{4})+0.5\bar{3}} \approx 0.52 \end{aligned}$$

où la normalisation des deux sommes implique clairement la probabilité des six événements étudiés. Dans ce cas particulier, la légère perturbation que

subit la distribution (rompant son uniformité) est entièrement attribuable à la seule irrégularité que constitue la probabilité de passage du mot w_3 au mot w_2, lègèrement plus petite (0.3125, par opposition à 0.375 pour les autres transitions). Ce léger déséquilibre donne une légère avance à la probabilité postérieure $P(s_2|w_3, w_2)$, ce qui explique la modification de la distribution que nous observons.

Bien qu'il serait possible de faire une micro-analyse détaillée expliquant tous les changements survenus dans la première itération d'entraînement, il suffit de savoir que le principe est rigoureusement identique pour la réestimation de tous les paramètres, la distribution conditionnelle des événements étant modifiée selon les mêmes règles. Une analyse de cette sorte deviendrait en outre rapidement fastidieuse, étant donné la dépendance récursive des calculs d'une couche de paramètres sur les couches précédentes : la réestimation de la couche la plus profonde $P(Z|H)$ dépend ainsi de la réestimation de la couche intermédiaire $P(S|Z)$, qui dépend elle-même de la couche dont nous venons d'étudier en détail la dynamique d'apprentissage, $P(W|S)$.

Il est par contre facile de vérifier que l'entraînement augmente bel et bien la performance du modèle : examinons ce qu'il advient de la table des transitions d'un mot à un autre, si on tourne quelques fois encore la manivelle de notre procédure. Au bout de cinq itérations d'entraînement, cette table devient :

$P_{HZSW}(W\|H)$	w_1	w_2	w_3
w_1	0.1022 :↓	0.5046 :↑	0.3932 :↑
w_2	0.5046 :↑	0.2014 :↓	0.2940 :↓
w_3	0.3932 :↑	0.294 :↓	0.3128 :↑

Tableau 3.10 – Probabilité réestimée de transition d'un mot à un autre sur le corpus $\mathcal{T}$, selon le modèle $HZSW$.

où les flèches indiquent encore une fois l'augmentation ou la diminution de la probabilité des événements, par rapport à la table originale 3.6. On peut réévaluer à l'aide de certaines de ces valeurs la vraisemblance du corpus $\mathcal{T}$ (3.3) :

$$P_{HZSW}(\mathcal{T}) = P_{HZSW}(w_1^T) \approx 0.00027$$

ce qui constitue clairement une amélioration (rappelons que la vraisemblance calculée à l'aide des paramètres initiaux, avant l'entraînement, était ≈ 0.00008) mais est encore inférieure à la vraisemblance du modèle bigramme (avec lequel la vraisemblance était ≈ 0.0008).

Algorithme 1 Une époque d'entraînement du modèle de langage lissé avec les sens.

(1) pour chaque paire $\{w_t, h_t\}$ du corpus d'entraînement :
  (2) propagation avant : calculer (dans cet ordre) les probabilités de transition $P(Z|h_t)$, $P(S|h_t)$ et $P(w_t|h_t)$
  (3) propagation arrière : calculer (dans cet ordre) les probabilités postérieures $P(w_t, S|w_t, h_t)$, $P(S, Z|w_t, h_t)$, $P(Z, h_t|w_t, h_t)$, et les accumuler
(4) mettre à jour les paramètres $P(Z|H)$, $P(S|Z)$ et $P(W|S)$ à l'aide des espérances calculées en (3)

3.2.4 Une évaluation du modèle HZSW

Des expériences ont été menées afin d'évaluer la performance du modèle HZSW sur le corpus Brown [8].

Il doit être clair d'emblée que nous ne nous soucierons pas d'établir une véritable étude comparative avec les modèles à l'état de l'art issus de la recherche de pointe en traitement du langage naturel. Plusieurs raisons motivent cette attitude. Les difficultés techniques résultant de telles comparaisons dépasseraient premièrement le cadre de cette étude, dont le but premier est exploratoire. Les modèles de langage de qualité "industrielle" sont devenus un art très précis et délicat pour lequel une grande maîtrise technique est exigée (un bon exemple de cette complexité est donné par les études de Chen et Goodman (CHEN et GOODMAN 1998; CHEN et GOODMAN 1999; GOODMAN 2001)). Nous ne prétendrons pas à une telle maîtrise ici. Le trop grand nombre d'options et de choix architecturaux que nous imposeraient une comparaison

[8] Tous les programmes qui ont été utilisés pour produire les résultats de cette recherche ont été écrits en C++, et s'inscrivent dans le prolongement de la librairie PLearn, développée au laboratoire LISA, dans un environnement de type Unix (GNU/Linux).

exhaustive avec de "vrais" modèles ferait en sorte de brouiller le comportement véritable et fondamental de nos modèles, que nous cherchons au contraire à dégager avec la plus grande clarté. Voilà pourquoi nous nous contenterons d'une méthodologie claire et sans ambiguïté, qui nous permettra de déterminer si l'ajout de variables cachées augmente ou non la qualité et la capacité de généralisation d'un modèle de langage statistique.

Le corpus Brown

Le corpus Brown est composé d'une suite de 1 181 041 mots provenant de textes et articles aux sujets divers, rédigés en anglais. Le corpus comporte des balises qui indiquent où commence et se termine un texte particulier. Nous avons prétraité le corpus Brown de manière simple en nous contentant de remplacer la ponctuation (ou ce qui semblait être de la ponctuation) par un symbole unique (`<punctuation>`) et les symboles à caractère numérique (détectés à l'aide d'une heuristique simple) par un autre symbole unique (`<numeric>`). Un vocabulaire des 15 000 mots les plus fréquents a ensuite été extrait sur le corpus filtré. Ce vocabulaire forme la base d'une ontologie de 34 339 synsets (dont 31 889 sens et 2 450 concepts de plus haut niveau) avec laquelle nous avons entraîné nos modèles. Sur ces 15 000 mots, 13 448 sont "connus" de WordNet (ils sont répertoriés et attachés à l'ontologie), et 1 512 sont des mots "inconnus" (qui peuvent être par exemple des pronoms, des déterminants ou des noms propres n'apparaissant pas dans la base de données). Ces mots inconnus se voient néanmoins accorder un rôle dans l'ontologie : à chacun est attribué un sens unique et relié à un concept de très haut niveau (du même niveau que les super-concepts lexicaux). Les mots ne faisant pas partie du vocabulaire (parce que leur fréquence était trop faible) sont quant à eux rassemblés en une seule classe (`<oov>`, "out of vocabulary"). La totalité du corpus est finalement divisée en trois ensembles distincts : l'ensemble réservé à l'entraînement des paramètres est composé des 900 000 premiers mots, l'ensemble de validation est composé des 100 000 mots suivants, et l'ensemble de test des 181 041 mots restants. Le tableau 3.11 récapitule le prétraitement du corpus Brown.

corpus Brown	1 181 041 mots
vocabulaire	15 000 mots (13 448 connus, 1 512 inconnus)
ontologie	31 889 sens, 2 450 concepts
entraînement	900 000 mots (76%)
validation	100 000 mots (8%)
test	181 041 mots (16%)

Tableau 3.11 – Le corpus Brown.

Il est à noter que l'utilisation d'une classe unique pour les mots inconnus fait en sorte d'améliorer significativement la performance des modèles, étant donné que la prédiction d'un mot à partir de cette classe (ou de cette classe à partir d'un mot) encapsule un grand nombre de décisions plus fines. Tous les modèles évalués dans le contexte de cette étude sont cependant à égalité car ils bénéficient tous de l'usage de cette classe.

Mesurer la qualité d'un modèle

Il est courant en pratique de mesurer la qualité d'un modèle de langage statistique en rapportant sa *perplexité* sur les données d'entraînement et de test :

$$PP(w_1^T) = e^{\frac{1}{T}\log P(w_1^T)} \tag{3.16}$$

Cette métrique trouve une justification précise avec la théorie de l'information [9], mais il est possible de l'interpréter de manière intuitive en tant que nombre moyen de mots sur lesquels le modèle "hésite", au moment d'une prédiction. Une perplexité de 200 (typique en pratique) signifie donc que le modèle aurait à choisir en moyenne entre 200 mots différents, tandis qu'un modèle aveugle, qui assignerait l'équiprobabilité aux $|V|$ mots de son vocabulaire aurait quant à lui une perplexité de $|V|$. .

Bien qu'il existe d'autres métriques d'évaluation pour les modèles de lan-

[9] Où elle dérive en fait de l'*entropie*, le degré moyen d'incertitude (ou de surprise).

gage (CHEN, BEEFERMAN et ROSENFELD 1998), nous nous restreindrons à la perplexité dans le cadre de cette étude.

Les modèles de base

(CHEN et GOODMAN 1998) rapportent que la performance des techniques de lissage de Katz et de Jelinek-Mercer est en fait dépendante de la taille du corpus d'entraînement : la méthode de Katz fonctionne mieux avec de gros corpus tandis que la méthode de Jelinek-Mercer fonctionne mieux avec des corpus plus modestes. Nous avons donc décidé de nous restreindre à l'utilisation de la méthode de Jelinek-Mercer, qui offre l'avantage d'être facilement applicable à nos modèles [10].

Nous avons donc comparé le modèle HZSW à un modèle bigramme interpolé avec la procédure décrite à la section 2.2.2, et entraîné sur le corpus Brown. Au bout de 10 itérations, la perplexité de l'ensemble de validation est à peu près stabilisée :

entraînement	validation	test
103	244	237

Tableau 3.12 – Perplexité du bigramme sur le corpus Brown.

En examinant ces résultats, on remarque tout d'abord que la perplexité sur l'ensemble d'entraînement est très faible (en comparaison avec la perplexité sur les ensembles de validation et de test). Ceci est dû au fait qu'un modèle d'apprentissage devient une sorte de "réflexion" des données ayant participé à son entraînement. Sa performance sur cette section du corpus n'est donc pas véritablement informative ou significative. Elle constitue néanmoins un critère fiable quant à la progression de l'entraînement du modèle (elle devrait en général diminuer à chaque nouvelle itération).

Voici la forme interpolée du modèle HZSW :

[10] Définir la fonction de redistribution de la masse de probabilité au modèle d'ordre inférieur serait beaucoup plus complexe avec la méthode de Katz.

$$\begin{aligned} P_{HZSW}(w|h) &= \lambda_2[q(h)]\hat{P}_{HZSW}P(w|h) + \\ &= \lambda_1[q(h)]P_{MLE}P(w) + \\ &= \lambda_0[q(h)]\frac{1}{|V|} \end{aligned} \tag{3.17}$$

où $\hat{P}_{HZSW}(w|h)$ est la version non-lissée du modèle, à ne pas confondre avec $P_{HZSW}(w|h)$, et où les $\lambda_i[q(h)]$ sont optimisés sur le corpus de validation.

Il est à noter que l'entraînement de ce modèle est passablement coûteuse [11]. Ceci est en majeure partie dû à la propagation du flot de probabilité à travers la table $P(S|Z)$, régissant la transition d'un sens à un autre, par nature beaucoup plus dense qu'une table de transition d'un mot à un autre. En dépit des "trucs" d'optimisation dont nous avons fait usage [12], l'exécution de cet algorithme, de par sa structure, est beaucoup plus coûteux qu'un bigramme. Voici les résultats au bout de 10 itérations :

entraînement	validation	test
103	242	236

Tableau 3.13 – Perplexité du modèle HZSW sur le corpus Brown.

On constate que ces résultats sont pratiquement identiques à ceux obtenus avec le modèle bigramme.

[11] Une itération complète demande près de neuf heures de calcul sur une machine munie d'un Pentium 4 et de 2GB de mémoire RAM.

[12] Le résultat du calcul de $P(w|h)$ pour un w et un h particuliers peut par exemple être stocké dans une table, afin de ne pas le refaire inutilement.

3.3 Une extension avec des concepts de plus haut niveau

La stratégie du modèle précédent repose en fait principalement sur une caractéristique fondamentale des données avec lesquelles on l'entraîne : l'enchevêtrement des sens. Cette notion s'illustre très clairement avec l'exemple de l'ontologie de notre modèle-jouet : les sens s_1 et s_4 ne peuvent générer respectivement qu'un seul mot, tandis que les sens s_2 et s_3 sont attachés à trois mots différents. Sans cette configuration cruciale, les données d'entraînement ne consitueraient qu'un ensemble de chemins indépendants et uniques reliant un mot à un autre. La simple connaissance de l'*existance* de ces chemins alternatifs ne serait d'aucune utilité au modèle, qui devrait se contenter de sommer leur probabilité, ce qui est équivalent au fait d'estimer directement la probabilité de transition de leur point de départ à leur point d'arrivée (d'un mot à un autre donc, comme dans un bigramme). L'enchevêtrement des sens qu'offre l'ontologie prévient ceci en permettant aux variables de sens de créer un réseau complexe de chemins composés d'embranchements permettant parfois d'atteindre plus d'un mot à partir d'un sens, et vice-versa. La figure 3.1 illustre schématiquement cette idée : certains embranchements permettent au flot de probabilité de dériver du chemin principal (représenté par les flèches pleines, reliant la paire de mots w et w', rencontrée lors de l'entraînement) et d'emprunter des chemins alternatifs (représentés par des flèches pointillées), formant des combinaisons de mots n'ayant pas été rencontrées lors de l'entraînement (ww et ww'').

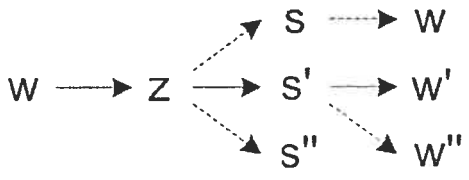


Figure 3.1 – L'enchevêtrement des sens.

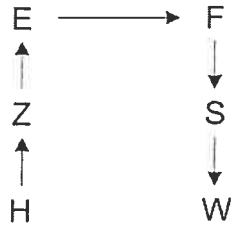
L'enchevêtrement des sens est donc une caractéristique essentielle de notre

ontologie, qu'il est potentiellement souhaitable de contrôler afin de l'optimiser. Ce problème est toutefois difficile, car il n'existe évidemment pas une procédure permettant de calibrer le "degré" de l'enchevêtrement, fixé antérieurement par la structure de l'ontologie. Même s'il était possible de paramétriser cet aspect structurel du modèle, le problème resterait complet, car trouver le degré d'enchevêtrement optimal impliquerait des solutions à des problèmes beaucoup plus complexes. Le "bon" degré se situerait ainsi entre deux bornes extrêmes : le degré 0 (aucun enchevêtrement, chaque sens étant confiné dans l'univers d'un seul mot), et le degré infini (un seul sens général de très haut niveau peut générer l'ensemble des mots du vocabulaire), réduisant le modèle à un prédicteur uniforme (zérogramme). Le degré d'enchevêtrement optimal permettrait donc de générer toutes les combinaisons de mots faisant partie du langage, tout en rejetant celles n'en faisant pas partie, ce qui devrait encore une fois faire appel à la fonction discriminante `appartient_au_langage(proposition)` à laquelle nous n'avons évidemment pas accès. On pourrait par contre argumenter que les concepteurs de l'ontologie particulière dont nous faisons usage ont déjà résolu le problème en quelque sorte, en posant de manière explicite (à l'aide de leur jugement) les sens que peut dénoter un mot. Mais cette source d'information à priori (bien qu'elle soit utile et qu'elle ait été minutieusement conçue) n'a cependant pas à elle seule toute la souplesse requise : nous avons besoin d'un mécanisme capable d'apprendre à calibrer sa représentation de façon autonome et dynamique.

La possibilité d'un tel mécanisme nous est offerte par WordNet. Nous avons vu à la section 2.4 que WordNet organise l'information lexicale en une série d'ontologies basées sur des relations sémantiques particulières. Une de ces relations, l'hypernymie, consiste en l'agrégation de concepts de différents niveaux d'abstraction, formant un "quasi-arbre" dont la racine est un concept universel, le plus général qui soit, et les feuilles des concepts très spécialisés correspondant aux sens des mots. Il est possible d'exploiter cette structure en introduisant dans notre modèle des variables cachées supplémentaires, correspondant à ces concepts de plus niveau niveau dans la hiérarchie. Ceci devrait permettre d'augmenter sa capacité de représentation et de généralisation.

Afin de lever toute ambiguïté, fixons tout d'abord notre terminologie, qui devient plus complexe avec l'introduction de ces nouvelles idées : la **hiérarchie** se rapporte à la structure quasi-arborescente que nous extrayons de WordNet, et qui contient les hypernymes dont nous voulons faire usage. Nous utiliserons une convention selon laquelle la position de l'arbre est inversée (voir la figure 2.1) : ses feuilles, le **niveau des mots**, en constitue donc le "bas", et la racine (le concept le plus général, que nous nommerons le **concept universel**, ou c_0), en constitue le "haut". On note également que le **niveau des sens** se trouve juste au-dessus du niveau des mots [13]. Un **concept** est un noeud intermédiaire à un niveau quelconque de la hiérarchie (hormis les niveaux des mots et des sens) qui représente une idée plus générale que les sens. Les **ancêtres** sont les concepts qu'il est possible d'atteindre en *remontant* dans la hiérarchie à partir d'un mot, d'un sens ou d'un concept (c_0 est donc l'ancêtre de tous les éléments de l'ontologie). À l'inverse, les **descendants** sont les sens ou les mots qu'il est possible d'atteindre à partir d'un concept, en se dirigeant vers le bas.

Comme nous l'avons fait pour le précédent modèle, formalisons la nouvelle architecture à l'aide d'un modèle graphique simple :



où les nouvelles variables E et F représentent des concepts de plus haut niveau que les sens. Le domaine de ces variables est différent de celui des variables

[13] On remarque que la notion de niveau dans la hiérarchie peut s'avérer quelque peu confuse car WordNet ne définit pas la relation d'hypernymie de manière uniforme pour les différentes catégories lexicales, comme nous l'avons vu à la section 2.4. Si on numérote les niveaux à partir du bas, c_0 pourra ainsi se retrouver à plusieurs niveaux différents, selon la profondeur de l'arbre au noeud de départ. Notre usage de cette notion se contentera donc de rester délibérément vague, et servira seulement à établir une distinction entre les concepts de "plus haut" et de "plus bas" niveau.

de sens S et Z. Alors que les valeurs que peuvent prendre ces dernières sont confinées à un niveau précis et invariable, les variables de concept puisent leurs valeurs dans l'ensemble des concepts se trouvant au-dessus des sens, à un niveau quelconque de la hiérarchie. Tout comme pour les variables de sens, il est nécessaire d'exploiter la structure de l'ontologie afin de contenir l'explosion combinatoire qui résulterait de l'absence de contrainte sur le domaine des variables de concept. Étant donné un mot particulier w, les valeurs permises sont donc confinées dans l'ensemble des ancêtres de ce mot, soit les concepts qu'il est possible de rejoindre en remontant vers c_0 à partir des sens de w, comme l'illustre ce schéma :

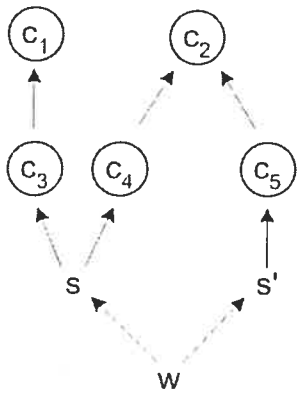


Figure 3.2 – Les ancêtres d'un mot.

On remarque que le choix d'une valeur pour les variables de concept est plus complexe que le choix d'une valeur pour les variables de sens dans le modèle précédent. Alors qu'on se contentait de choisir un sens parmi ceux permis pour un mot donné, le choix d'un concept implique maintenant également la détermination de son niveau d'abstraction.

Imaginons une situation mettant en évidence les avantages escomptés d'un tel ajout à la structure du modèle. La premier schéma de la figure 3.3 illustre une situation pathologique où les deux chemins qui relient les mots w et w' sont composés de sens appartenant *uniquement* à ces mots. La décomposition de l'événement en chemins indépendants et uniques ne permet rien de plus qu'un chemin direct reliant les deux mots ne le permettrait : elle ne fait qu'alourdir

le modèle en y introduisant des degrés de liberté superflus. Notre premier modèle reposait en fait sur l'espoir que de tels événements ne seraient pas trop fréquents. Le deuxième schéma de la figure illustre une situation où notre deuxième modèle transforme, à l'aide de l'ontologie, un chemin simple en un réseau comprenant possiblement plusieurs bifurcations entrantes et sortantes, ce qui devrait favoriser la capacité de généralisation du modèle en permettant de nouvelles combinaisons de mots reliés par des concepts de plus haut niveau que les sens.

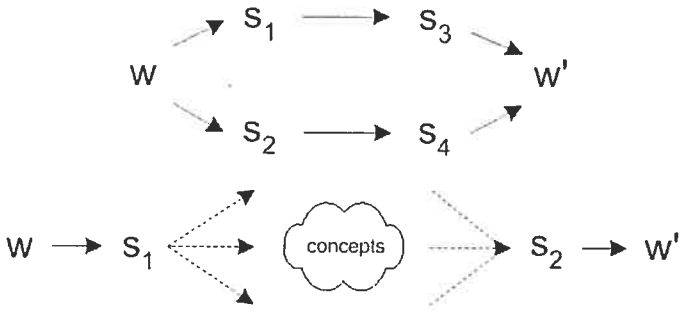
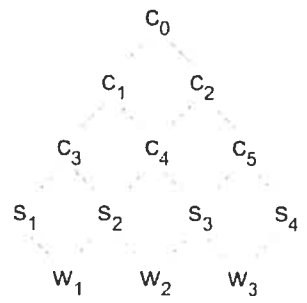


Figure 3.3 – Deux types de chemin reliant le mot w au mot w' : des chemins composés de sens uniquement (haut), et des chemins composés de sens et de concepts (bas).

Reprenons l'étude de notre modèle-jouet en définissant tout d'abord un ensemble de concepts de plus haut niveau que les sens :

$$C = \{c_0, c_1, c_2, c_3, c_4, c_5\} \tag{3.18}$$

et étendons ensuite son ontologie, afin de les y incorporer :



On retrouve à la tête de cette ontologie le concept universel c_0, duquel héritent tous les autres. Le vocabulaire V (3.1) et le corpus d'entraînement $\mathcal{T}$ (3.3) restent inchangés.

De nouvelles tables de paramètres devront être ajoutées au modèle : $P(E|Z)$ (la probabilité qu'un concept ait été généré par un sens), $P(F|E)$ (la probabilité de transition d'un concept à un autre, jouant un rôle similaire à celui que jouait la table de transition d'un sens à un autre, dans le précédent modèle) et $P(S|F)$ (la proabilité qu'un sens descende d'un concept). On inititialise les paramètres selon un procédé similaire au premier modèle, en posant l'hypothèse que la distribution des concepts est uniforme. La propagation avant du flot de probabilité devra passer par les nouveaux chemins que nous avons introduits :

$$P_{HZEFSW}(w_t|w_{t-1}) = \sum_{z,e,f,s} P(w_t|s) \cdot P(s|f) \cdot P(f|e) \cdot P(e|z) \cdot P(z|w_{t-1}) \quad (3.19)$$

La procédure d'entraînement est également similaire à la précédente, avec l'ajout de trois formules de réestimation pour les nouveaux paramètres :

$$P(e|z) \leftarrow \frac{\sum_t P(e_t = e, z_t = z|w_t, h_t)}{\sum_t \sum_{e'} P(e_t = e', z_t = z|w_t, h_t)} \quad (3.20)$$

$$P(f|e) \leftarrow \frac{\sum_t P(f_t = f, e_t = e|w_t, h_t)}{\sum_t \sum_{f'} P(f_t = f', e_t = e|w_t, h_t)} \quad (3.21)$$

$$P(s|f) \leftarrow \frac{\sum_t P(s_t = s, f_t = f|w_t, h_t)}{\sum_t \sum_{s'} P(s_t = s', f_t = f|w_t, h_t)} \quad (3.22)$$

Démarrons la procédure, et dirigeons notre attention sur l'état des paramètres de la table des probabilités de transition d'un sens à un concept, $P(E|Z)$, après 50 itérations d'entraînement :

$P(E\|Z)$	s_1	s_2	s_3	s_4
c_0	0.0559	0	0	0.7467
c_1	0.0982	0	0	0
c_2	0	0.5546	0.1766	0.1432
c_3	0.8459	0	0	0
c_4	0	0.4454	0.4628	0
c_5	0	0	0.3606	0.1101

Tableau 3.14 – La probabilité réestimée d'un sens générant un concept sur le corpus $\mathcal{T}$.

L'examen de cette table particulière nous permet de nous rendre compte de l'"usage" que fait le modèle de l'ontologie. À priori, tout peut sembler normal (la vraisemblance des données d'entraînement a bel et bien augmenté, de manière continue, pendant les 50 itérations) mais si on examine de plus près les valeurs de cette table, on remarque un phénomène singulier : la masse de probabilité associée aux transitions de plus bas niveau (les transitions d'un sens à une catégorie directement parente : $s_1 \rightarrow c_3$, $s_2 \rightarrow c_3$, $s_2 \rightarrow c_4$, $s_3 \rightarrow c_4$, $s_3 \rightarrow c_5$, $s_4 \rightarrow c_5$), alors qu'elle était évidemment proportionelle à l'initialisation, est maintenant devenue dominante (elle est passée de 36% à 55%). Cette tendance, encore plus évidente sur d'autres ensembles d'entraînement avec lesquels nous avons expérimenté, est due au fait que le modèle n'a pas "intérêt" à s'intéresser aux catégories des niveaux supérieurs, étant donné la perte d'information que leur utilisation implique. En effet, plus on monte dans la hiérarchie, plus l'information contenue dans l'ensemble d'entraînement est diluée en des concepts de plus en plus généraux. Bien que l'usage d'une ontologie présente l'avantage d'améliorer possiblement la performance du modèle sur un ensemble de données inconnu, ce mécanisme est en contradiction avec les objectifs immédiats de la fonction de coût que nous cherchons à optimiser, obstinément préoccupée par la vraisemblance des événements qui composent l'ensemble d'entraînement. En d'autres termes, si on donne au modèle la possibilité de choisir lui-même le niveau d'abstraction de sa représentation, il choisira l'option la moins lisse possible [14].

[14] On remarque qu'on ne donnait *pas* ce choix au modèle précédent.

Il est possible de modifier légèrement la structure du modèle afin de se convaincre que ceci n'est pas seulement l'effet d'un aléa quelconque du processus d'optimisation. L'architecture actuelle découple de manière explicite les niveaux des sens et des concepts de plus haut niveau, et force le flot de probabilité à passer d'un niveau à l'autre. En brouillant la distinction entre les deux ensembles (en en faisant l'union) on relâche cette contrainte en permettant au flot de demeurer au niveau des sens [15]. Examinons donc la table $P(E|Z)$ d'un modèle ainsi modifié, après 50 itérations d'entraînement :

$P(E\|Z)$	s_1	s_2	s_3	s_4
s_1	0.7069	0	0	0
s_2	0	0.7918	0	0
s_3	0	0	0.5402	0
s_4	0	0	0	0.4565
c_0	0.0061	0	0	0.3514
c_1	0.0299	0.1904	0.0086	0
c_2	0	0.0096	0.0556	0.1036
c_3	0.2572	0	0	0
c_4	0	0.0082	0.2343	0
c_5	0	0	0.1613	0.0885

Tableau 3.15 – La probabilité réestimée d'un sens générant un sens ou un concept sur le corpus $\mathcal{T}$.

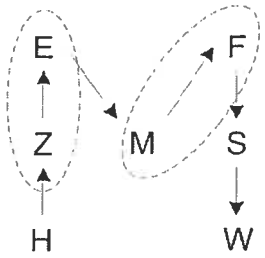
Les quatre premières rangées correspondent aux nouvelles transitions "stationnaires" qu'il est maintenant possible pour le flot d'emprunter et où la masse de probabilité est clairement cantonnée (la probabilité de chaque transition constitue le maximum d'une colonne). En poussant l'investigation un peu plus loin, on se rend compte que la table des probabilités de transition d'un concept à un autre, $P(F|E)$ (qui inclut maintenant des transitions impliquant les sens), reflète également cette tendance : les transitions les plus probables sont exactement celles que favorisait le modèle HZSW, qui n'avait que les sens à sa disposition. Ceci confirme donc notre soupçon : l'usage de l'ontologie complète est "rejeté" par le modèle, qui se contente d'un sous-ensemble équivalent

[15] La table $P(E|Z)$ admet maintenant par exemple la transition $s \rightarrow s$, auparavant interdite par la structure du modèle.

à l'ontologie de sens du modèle précédent.

Comment inciter le modèle à utiliser l'ontologie que nous lui offrons ? Une solution consiste à découpler la réestimation des paramètres du modèle sur des ensembles de données disjoints (certains paramètres entraînés avec l'ensemble d'entraînement, d'autres avec l'ensemble de validation, par exemple), afin de "tromper" la fonction d'optimisation en ne lui permettant pas de "coller" de trop près aux données d'une seule source. Ceci devrait forcer le choix de concepts de plus haut niveau et ainsi favoriser le lissage.

Cette nouvelle procédure d'entraînement nécessite cependant qu'on modifie quelque peu l'architecture du modèle, en introduisant une variable de sens additionnelle, M, ayant pour effet de découpler la table de transition d'un concept à un autre, $P(F|E)$, en deux tables distinctes :



Cette idée repose en fait sur un mécanisme du même genre que celui de la mixture de modèles, exposé à la section 2.2.2. Alors que les transitions d'un sens à un concept de bas niveau [16] devraient donner de bons résultats sur le premier ensemble de données, servant à l'entraînement de la plupart des paramètres, il n'en sera pas nécessairement de même sur le deuxième ensemble, servant à l'entraînement des *paramètres de lissage* $P(E|Z)$ et $P(F|M)$ (entourés de pointillés sur la figure et qui régissent justement les transitions d'un sens à un concept). Pour rétablir la qualité délibérément diminuée des prédictions sur ce deuxième ensemble, le modèle devra entraîner les paramètres de lissage de manière à favoriser des transitions vers des concepts de plus haut niveau, susceptibles de couvrir (expliquer) un plus grand nombre d'événements.

[16] $z \rightarrow e$ ou $m \rightarrow f$ avec e et f des concepts assez spécialisés, demeurant au bas de l'ontologie, par exemple.

Il est possible de contraindre la capacité du modèle en imposant une limite au niveau d'abstraction des concepts à l'aide d'un critère simple : la vérification du fait que le nombre de mots *descendants* d'un concept donné ne dépasse pas un certain seuil. Les concepts vagues et généraux couvrant un trop grand nombre de mots seront ainsi rejetés, ce qui réduira le nombre de chemins possibles reliant deux mots. En particulier, les transitions impliquant le concept c_0 (couvrant l'ensemble du vocabulaire) ne seront pas considérées si le seuil est inférieur à $|V|$.

Algorithme 2 Une époque d'entraînement du modèle de langage lissé avec les concepts.

(1) pour chaque paire $\{w_t, h_t\}$ du corpus d'entraînement :
  (2) propagation avant : calculer (dans cet ordre) les probabilités de transition $P(Z|h_t)$, $P(E|h_t)$, $P(M|h_t)$, $P(F|h_t)$, $P(S|h_t)$, $P(w_t|h_t)$
  (3) propagation arrière : calculer (dans cet ordre) les probabilités postérieures $P(w_t, S|w_t, h_t)$, $P(S, F|w_t, h_t)$, $P(F, M|w_t, h_t)$, $P(M, E|w_t, h_t)$, $P(E, Z|w_t, h_t)$, $P(Z, h_t|w_t, h_t)$, et les accumuler
(4) mettre à jour les paramètres $P(Z|H)$, $P(M|E)$, $P(S|F)$ et $P(W|S)$ à l'aide des espérances calculées en (3).
(5) répéter (1) à (3) avec chaque paire $\{w_t, h_t\}$ du corpus de validation.
(6) mettre à jour les paramètres $P(E|Z)$, $P(F|M)$ à l'aide des espérances calculées en (5).

Chapitre 4

Un modèle mettant à profit la désambiguïsation du sens

En dépit des apparences, l'usage des paramètres impliquant le sens des mots dans les modèles précédents ($P(S|W)$ et $P(W|S)$, par exemple) ne constituait pas un véritable mécanisme de désambiguïsation. Considérons pour s'en convaincre deux sens courants du mot ambigu "*calcul*" : "*procédure effectuée sur des symboles*" (s) et "*agglomérat de sels minéraux ou de matières organiques*" (s') [1]. Sans information à priori, une hypothèse raisonnable est l'uniformité de leur distribution :

$$\begin{aligned} P(s|calcul) &= \frac{1}{2} \\ P(s'|calcul) &= \frac{1}{2} \end{aligned}$$

L'entraînement d'un modèle contenant ces paramètres ferait probablement en sorte de les modifier, comme nous l'avons vu : un des sens de "*calcul*" pourrait voir sa probabilité augmenter, par exemple, et l'autre diminuer. Mais est-ce que cette modification constituerait un jugement valable, susceptible de diminuer l'incertitude quant au véritable sens du mot dans un contexte

[1] Selon le dictionnaire Petit Robert, le mot "*calcul*" possède plus de deux sens, mais ceux que nous avons choisis représentent bien des classes d'usage clairement disjointes.

précis ? Examinons tout d'abord une situation où le mot ambigu joue le rôle de contexte. Supposons que le mot "*calcul*" soit seulement suivi des mots "*biliaire*" (une seule fois), et "*rapide*" (une seule fois), dans l'ensemble d'entraînement :

$$P(biliaire|calcul) = \frac{1}{2} \tag{4.1}$$

$$P(rapide|calcul) = \frac{1}{2} \tag{4.2}$$

où (4.1) se rapporte clairement au sens s', et (4.2) au sens s. Dans ces deux exemples, le seul indice susceptible d'aider le modèle à choisir le bon sens de "*calcul*" est le mot suivant, qu'on tente justement de prédire. Les autres mots du voisinage n'étant pas pris en considération, le modèle est aussi peu informé quant au véritable sens du mot "*calcul*" que le serait l'algorithme naïf décrit à la section 2.3, basé uniquement sur la fréquence. En d'autres termes, la connaissance qu'a le modèle de la probabilité d'un mot dénotant un sens est à peu près équivalente à sa connaissance de la probabilité d'un mot suivant un autre mot : elle n'est pas d'un ordre plus élevé. Ceci n'implique pas par contre que cette capacité supplémentaire soit inutile : le modèle pourrait ainsi découvrir qu'un autre mot partageant le sens s ("*procédure*" par exemple) est souvent suivi du mot "*rapide*", ce qui aurait une influence sur la prédiction $P(rapide|calcul)$, par exemple.

Examinons ensuite le cas inverse, où le mot ambigu est celui que nous tentons de prédire :

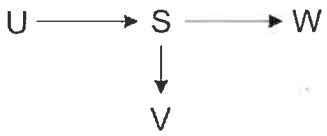
$$P(calcul|douloureux) \tag{4.3}$$

$$P(calcul|savant) \tag{4.4}$$

Ces exemples bien choisis montrent qu'un seul mot de contexte est parfois suffisant pour désambiguïser le sens d'un mot. Le problème toutefois est que cette désambiguïsation est inutile dans un tel contexte, étant donné qu'elle intervient seulement au moment où le mot ambigu a *déjà* été prédit : le mécanisme

tourne pour ainsi dire à vide...

On peut conclure à partir de ces exemples que la désambiguïsation ne joue pas un rôle effectif dans le contexte de modèles du second ordre tels que ceux que nous avons précédemment étudiés. L'examen des exemples (4.3) et (4.4) peut cependant nous suggérer un nouveau modèle dans lequel le résultat de la désambiguïsation du mot w_t est propagé à la prédiction de mot suivant w_{t+1}. Ce modèle impliquant maintenant *deux* mots de contexte est représenté graphiquement par :



où U, V et W sont des variables lexicales représentant trois mots consécutifs, et S une variable qui correspond aux sens possibles du deuxième mot (V). Une fois levée l'ambiguïté d'un mot v (grâce au mot précédent u), la prédiction du mot w devrait être améliorée si on pose l'hypothèse raisonnable que la prédiction d'un mot à partir d'un mot *dénotant un sens particulier* est en général plus facile que la prédiction d'un mot à partir d'un mot *dénotant un sens quelconque.*

Examinons par exemple $P(biliaire|douloureux, calcul)$, la prédiction du mot "*biliaire*" à partir du bigramme de contexte "*douloureux calcul*". Si le modèle parvient à désambiguïser correctement le sens du mot "*calcul*" à l'aide du mot "*douloureux*", la prédiction se voit réduite à $P(biliaire|s')$, qui devrait être plus facile que la prédiction $P(biliaire|calcul)$.

La propagation avant du flot de probabilité pour ce modèle s'effectue d'une manière similaire à celles des modèles précédents :

$$P_{UVSW}(w|u,v) = \sum_s P(w|s) \cdot P(s|u,v) \tag{4.5}$$

où

$$P(s|u,v) = \frac{P(v|s) \cdot P(s|u)}{P(v|u)} \tag{4.6}$$

en vertu de la loi de Bayes, et où finalement

$$P(v|u) = \sum_s P(v|s) \cdot P(s|u) \tag{4.7}$$

Pour la propagation arrière, les formules de réestimation des trois tables de paramètres qui régissent le modèle sont :

$$P(w|s) \leftarrow \frac{\sum_t P(w_t = w, s_t = s|u_t, v_t, w_t)}{\sum_t \sum_{w'} P(w_t = w', s_t = s|u_t, v_t, w_t)} \tag{4.8}$$

$$P(v|s) \leftarrow \frac{\sum_t P(v_t = v, s_t = s|u_t, v_t, w_t)}{\sum_t \sum_{v'} P(v_t = v', s_t = s|u_t, v_t, w_t)} \tag{4.9}$$

$$P(s|u) \leftarrow \frac{\sum_t P(s_t = s, u_t = u|u_t, v_t, w_t)}{\sum_t \sum_{s'} P(s_t = s', u_t = u|u_t, v_t, w_t)} \tag{4.10}$$

Algorithme 3 Une époque d'entraînement du modèle de langage UVSW, mettant à profit la désambiguïsation.

(1) pour chaque triplet $\{u_t, v_t, w_t\}$ du corpus d'entraînement :
  (2) passe avant (fprop) : calculer (dans cet ordre) les probabilités de transition $P(v_t|u_t)$, $P(S|u_t, v_t)$ et $P(w_t|u_t, v_t)$
  (3) passe arrière (bprop) : calculer (dans cet ordre) les probabilités postérieures $P(w_t, S|u_t, v_t, w_t)$, $P(v_t, S|u_t, v_t, w_t)$, $P(S, u_t|u_t, v_t, w_t)$, et les accumuler
(4) mettre à jour les paramètres $P(W|S)$, $P(V|S)$ et $P(S|U)$ à l'aide des espérances calculées en (3).

4.1 Une évaluation du modèle UVSW

Étant donné la nature moins prohibitive des calculs du modèle UVSW (composé seulement de tables de transition des mots aux sens, ou des sens aux mots, dont la densité est beaucoup plus faible), il a été possible de l'entraîner et de le comparer à un corpus additionnel, beaucoup plus volumineux que Brown, soit le corpus AP News, composé d'articles de journaux de l'an-

née 1996, provenant de l'agence de presse AP. Nous lui avons fait subir un prétraitement similaire à celui du corpus Brown :

corpus AP News	14 535 816 mots
vocabulaire	15 000 mots (12 483 connus, 2 517 inconnus)
ontologie	30 596 sens, 2 598 concepts
entraînement	11 000 000 mots (76%)
validation	1 500 000 mots (10%)
test	2 035 816 mots (14%)

Tableau 4.1 – Le corpus AP News.

La question de la capacité du modèle UVSW est ambigüe : il peut sembler d'une part qu'elle soit de l'ordre d'un trigramme, étant donné l'utilisation d'un contexte de deux mots, mais d'autre part on remarque que la paramétrisation implique exclusivement des événements du second ordre. Étant donné cette incertitude, il semble raisonnable de le comparer à un trigramme (en plus du bigramme interpolé ayant servi à la comparaison du modèle HZSW) lissé avec la méthode Jelinek-Mercer :

$$\begin{aligned} P(w|u,v) &= \lambda_3[q(u,v)]P_{MLE}(w|u,v) + \\ &= \lambda_2[q(u,v)]P_{MLE}(w|v) + \\ &= \lambda_1[q(u,v)]P_{MLE}(w) + \\ &= \lambda_0[q(u,v)]\frac{1}{|V|} \end{aligned} \tag{4.11}$$

où la dimensionnalité de la matrice de paramètres de la mixture λ est maintenant fonction des deux mots de contexte.

Les tables suivantes présentes les résultats des modèles de base, obtenus au bout de 10 itérations, pour les deux corpus :

	entraînement	validation	test
bigramme	103	244	237
trigramme	34	243	235

Tableau 4.2 – Perplexité des modèles de base sur le corpus Brown.

	entraînement	validation	test
bigramme	103	145	146
trigramme	29	102	104

Tableau 4.3 – Perplexité des modèles de base sur le corpus AP News.

On constate qu'il semble beaucoup plus "facile" de modéliser le corpus AP News : la perplexité des modèles en test y est significativement plus basse que celle sur le corpus Brown. Cette différence s'explique probablement par le fait que le style journalistique a tendance à utiliser un vocabulaire et des formules relativement uniformes avec lesquels il est plus facile de faire des prédictions [2]. Le fait que les articles proviennent tous de la même année est susceptible de favoriser également ce biais. On remarque en outre que l'écart entre les performances du bigramme et du trigramme est beaucoup plus important sur le corpus AP News. La capacité supplémentaire du trigramme n'est pas mise à profit sur le corpus Brown, en raison de sa petite taille, ce qui corrobore un résultat fondamental de la théorie de l'apprentissage statistique.

Voici la forme finale de la version interpolée du modèle UVSW :

$$\begin{aligned} P_{UVSW}(w|u,v) &= \lambda_3[q(u,v)]P_{UVSW}(w|u,v) + \\ &= \lambda_2[q(u,v)]P_{MLE}(w|v) + \\ &= \lambda_1[q(u,v)]P_{MLE}(w) + \\ &= \lambda_0[q(u,v)]\frac{1}{|V|} \end{aligned} \tag{4.12}$$

[2] Rappelons que les textes du corpus Brown ne sont pas orientés autour d'une thématique particulière

Voici les résultats du modèle UVSW, obtenus au bout de 10 itérations, pour les deux corpus :

entraînement	validation	test
89	253	245

Tableau 4.4 – Perplexité du modèle UVSW sur le corpus Brown.

entraînement	validation	test
96	141	143

Tableau 4.5 – Perplexité du modèle UVSW sur le corpus AP News.

Chapitre 5

Analyse et critique

Un constat négatif clair se dégage de l'ensemble de cette étude, à la lumière des résultats obtenus : il ne semble pas que l'ajout de variables aléatoires cachées à un modèle de langage statistique fasse en sorte d'en améliorer la performance. Le but visé par l'utilisation de ces variables était la capture d'un aspect du langage plus profond que l'occurrence des phénomènes lexicaux de surface, à l'aide d'une combinaison de connaissances à priori (provenant de WordNet) et d'un mécanisme d'apprentissage statistique. Il y a évidemment lieu de questionner cet assemblage, un point de vue que nous développerons quelque peu dans ce chapitre.

5.1 Une granularité trop fine

Si on examine les ontologies extraites de nos corpus, on constate un détail particulièrement troublant : pour plusieurs mots, les sens qui l'accompagnent sont à peine distinguables les uns des autres. Considérons par exemple les dix sens du verbe "*read*" :

1. `read -- (interpret something that is written or printed; "read the advertisement"; "Have you read Salman Rushdie?")`
2. `read, say -- (have or contain a certain wording or form; "The passage reads as follows"; "What does the law say?")`
3. `read -- (look at, interpreted, and say out loud something that is written or printed; "The proclamation will be read")`

```
4. read, scan -- (obtain data from magnetic tapes; "This dictionary can
   be read by the computer")
5. read -- (interpret the significance of, as of palms, tea leaves,
   intestines, the sky, etc.; also of human behavior; "She read the sky
   and predicted rain"; "I can't read his strange behavior")
6. take, read -- (interpret something in a certain way; convey a
   particular meaning or impression; "I read this address as a satire";
   "How should I take this message?"; You can't take credit for this!")
7. learn, study, read, take -- (be a student of a certain subject; "She
   is reading for the bar exam")
8. read, register, show, record -- (indicate a certain reading; of gauges
   and instruments; "The thermometer showed thirteen degrees below zero";
   "The gauge read 'empty'")
9. read -- (to hear and understand; "I read you loud and clear!")
10. understand, read, interpret, translate -- (make sense of a language;
    "She understands French"; "Can you read Greek?")
```

Même en prêtant une attention particulière à l'exercice, il serait très difficile, pour un lecteur moyen, de faire une véritable distinction entre la plupart de ces usages du verbe, qui semblent se rapporter tous plus ou moins à un concept unique. On pourrait certes soutenir qu'il est fondamental, dans certains contextes, de savoir distinguer le quatrième sens (l'acte de lecture d'un *mécanisme*) des sens plus communs du verbe (tels que les sens 1 et 7, se rapportant à une activité de type manifestement plus *humaine*). Mais il reste qu'il y a lieu de s'interroger sur les effets de l'introduction de distinctions sémantiques aussi subtiles et nombreuses au sein de modèles probabilistes entraînés avec un nombre relativement restreint d'exemples.

Une solution simple et élégante au problème de la granularité des sens nous est pourtant fournie par l'ontologie elle-même. Examinons les hypernymes des dix sens du verbe "*read*" :

```
1. read
    => interpret, construe
        => understand
2. read, say
    => have, feature
3. read
```

```
    => talk, speak, utter, mouth, verbalize
        => communicate, intercommunicate
            => interact
                => act, move
4. read, scan
    => interpret, construe
        => understand
5. read
    => predict, foretell, prognosticate, call, forebode, anticipate, promise
        => guess, venture, hazard
            => speculate
                => reason
                    => think, cogitate, cerebrate
6. take, read
    => interpret, construe
        => understand
7. learn, study, read, take
8. read, register, show, record
    => indicate
        => inform
            => communicate, intercommunicate
                => interact
                    => act, move
9. read
    => understand
10. understand, read, interpret, translate
     => understand
```

On remarque que cinq de ces sens (1, 4, 6, 9, 10) partagent un ancêtre de haut niveau, le concept "*understand*". Ceci peut suggérer une méthode simple de réduction de l'univers des sens (illustrée avec la figure 5.1) : s'élever de k niveaux au-dessus du niveau des sens, et agglutiner tous les sens pour lesquels on y découvre un ancêtre commun. Chaque classe constitue ainsi un nouveau sens jouant le rôle d'un nombre variable de sens semblables.

Il est également possible de partir du "haut" de l'ontologie. L'ensemble des concepts se trouvant à k niveaux *sous* le concept universel est nécessairement de cardinalité plus faible que l'ensemble des sens véritables (se trouvant juste au-dessus des mots), étant donné leur degré d'abstraction plus élevé (plusieurs

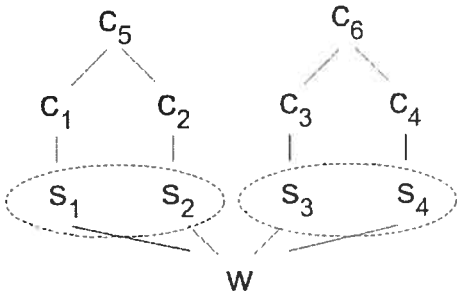


Figure 5.1 – Une première méthode de réduction de la polysémie dans WordNet (en partant du "bas", et en remontant de 2 niveaux dans la hiérarchie).

sens sont encapsulés par un seul concept plus général). Il est facile d'utiliser ces concepts en tant qu'ensemble restreint de sens, en extrayant simplement pour chacun l'ensemble de ses *mots descendants* (pour chaque mot w descendant du concept c, on définit c en tant que sens de w) :

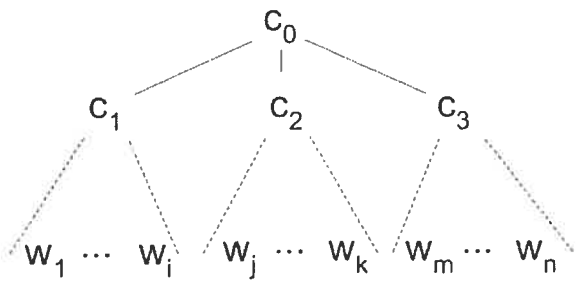


Figure 5.2 – Une deuxième méthode de réduction de la polysémie dans WordNet (en partant du "haut").

En choisissant le niveau d'arrêt k, on peut ainsi régler la cardinalité de l'ensemble des sens, ce qui aura en outre une influence directe sur le degré d'enchevêtrement : plus on se trouve près du concept universel, moins l'univers des sens est vaste, mais plus l'enchevêtrement est en revanche élevé, car plusieurs mots partagent le même sens. On note que ces méthodes ne s'appliquent en fait qu'aux branches nominales et verbales de l'ontologie, les seules pour

lesquelles l'hypernymie est définie.

5.2 L'utilisation de données supervisées

Le corpus SemCor est une ressource très prisée dans le domaine du traitement du langage (et plus particulièrement en désambiguïsation). Provenant du même contexte de recherche que WordNet, il est composé d'un sous-ensemble du corpus Brown (approximativement le quart) pour lequel la majorité des mots polysémiques ont été identifiés et désambiguïsés par des spécialistes en lexicographie (des spécialistes *humains*, il va sans dire !). Les sens possibles d'un mot sont fonction de la version particulière de WordNet qui a été utilisée [1]. Voici par exemple la version annotée de la phrase "*A similar resolution passed in the Senate by a vote of 29-5.*", tirée du premier fichier de SemCor :

```
<wf cmd=ignore pos=DT>A</wf>
<wf cmd=done pos=JJ lemma=similar wnsn=1 lexsn=3:00:00::>similar</wf>
<wf cmd=done pos=NN lemma=resolution wnsn=1 lexsn=1:10:01::>resolution</wf>
<wf cmd=done pos=VB lemma=pass wnsn=9 lexsn=2:41:04::>passed</wf>
<wf cmd=ignore pos=IN>in</wf>
<wf cmd=ignore pos=DT>the</wf>
<wf cmd=done pos=NN lemma=senate wnsn=1 lexsn=1:14:00::>Senate</wf>
<wf cmd=ignore pos=IN>by</wf>
<wf cmd=ignore pos=DT>a</wf>
<wf cmd=done pos=NN lemma=vote wnsn=1 lexsn=1:04:00::>vote</wf>
<wf cmd=ignore pos=IN>of</wf>
<wf cmd=done pos=NN lemma=29 wnsn=1 lexsn=1:23:00::>29</wf>
<punc>-</punc>
<wf cmd=done pos=NN lemma=5 wnsn=1 lexsn=1:23:00::>5</wf>
<punc>.</punc>
```

On peut aisément vérifier que le "bon" sens de "*Senate*" (faisant ici clairement référence, dans ce contexte, au Sénat des États-Unis, et non au concept général de sénat) est bel et bien celui choisi par l'annotateur, en faisant une recherche dans la base de données [2].

[1] Les sens de la version originale de SemCor sont définis par la version 1.6.

[2] En utilisant une clé composée du lemme et du lexsn.

Le corpus annoté SemCor trouve de nombreux usages, dont les plus évidents sont sans contredit pour la désambiguïsation : on peut s'en servir par exemple pour entraîner et évaluer des sytèmes d'apprentissage supervisé. Mais l'information contenue dans ce corpus pourrait également servir de complément supervisé à l'apprentissage essentiellement non-supervisé des modèles que nous avons développés dans le cadre de cette étude. À l'aide de cette supervision, il serait possible de rendre déterministe l'optimisation de nos modèles (à l'instar des modèles de langage sans variable cachée, pour lesquels toutes les observations sont disponibles). Pour le modèle HZSW, ceci se traduirait par le remplacement des formules de réestimation (3.13), (3.14) et (3.14) par ces simples estimateurs MLE :

$$P_{MLE}(w|s) = \frac{C(w,s)}{C(s)} \tag{5.1}$$

$$P_{MLE}(s|z) = \frac{C(s,z)}{C(z)} \tag{5.2}$$

$$P_{MLE}(z|h) = \frac{C(z,h)}{C(h)} \tag{5.3}$$

Si on la complète avec des données non-supervisées, cette injection de connaissances à priori dans le modèle devrait ainsi faire en sorte de le pousser délicatement dans la "bonne direction", tel un enfant à qui on apprendrait à marcher en le guidant de la main (pour éviter qu'il ne tombe), mais en le laissant la plupart du temps libre d'effectuer l'apprentissage par lui-même...

Il serait également possible de remplacer l'initialisation aléatoire des paramètres par une initialisation faisant usage de ces données supervisées.

5.3 Un outil puissant mais monolithique

Quand on pense aux efforts importants qu'a probablement dû coûter l'élaboration de documents tels que SemCor et WordNet, on en vient à se demander comment un tel *consensus* a pu être simplement possible quant à une question aussi fine, celle de la signification des mots. Cette question légitime évoque

encore une fois l'arme à double tranchant que constituent la richesse et la complexité de WordNet.

(VÉRONIS 2001) donne forme à cette critique à l'aide d'une étude montrant qu'un consensus entre différents juges est un événement rare dans le contexte de deux tâches bien précises ayant trait à la désambiguïsation : déterminer premièrement si les mots d'un ensemble donné sont polysémiques ou non, et déterminer ensuite le sens correct des mots jugés hautement polysémiques par la première expérience. En effectuant des corrections statistiques pour éliminer le biais inhérent à ce type de mesure [3], on trouve qu'un consensus en matière de désambiguïsation est un événement assez peu probable (sous la barre du 50% pour les deux tâches). Il est à noter que les juges étaient des étudiants en linguistique (sans formation particulière en lexicographie) et qu'ils étaient *payés* pour accomplir les tâches...

Il est important de bien saisir le véritable fer de lance de cette étude : elle n'a pas pour but de montrer que le problème de la désambiguïsation du sens est en soi difficile ou insurmontable, elle montre plutôt que nous ne parvenons pas en général à nous entendre sur ce qui constitue sa solution.

Véronis fait également une critique de la composition des outils dont les systèmes actuels font usage. Les dictionnaires traditionnels (de même que WordNet, une ressource dont le design et l'usage sont clairement plus orientés vers le traitement informatique) ne fournissent en fait que des *définitions* du sens des mots, au lieu de fournir les *indices distributionnels* ("distributional clues") plus concrets dont un système automatisé pourrait peut-être faire un meilleur usage. De fait, il s'avère dans bien des cas beaucoup plus facile d'attribuer un sens particulier à un mot en appliquant de simples critères basés sur des observations syntaxiques ou structurales de surface [4]. Par opposition,

[3] Un niveau de bonnes réponses obtenues par "chance" est à prendre à considération, ainsi que le nombre de juges, qui fait évidemment en sorte de faire tendre la probabilité d'un consensus vers zéro, à mesure qu'il croît.

[4] On donne ainsi l'exemple d'une distinction subtile quant à l'usage du mot " *degré*", qui pourrait être tranché à l'aide de la simple détection de mots à caractère ordinal ou cardinal (*premier* degré, *deuxième* degré...) dénotant un premier sens (qu'on pourrait qualifier de "discret") ou encore de mots marquant l'intensité, dénotant un deuxième sens (qu'on pourrait qualifier de "continu" : un *faible* degré, degré *élevé*...). Un autre exemple intéressant concerne

il semble beaucoup plus difficile de déterminer le sens d'un mot en procédant par *introspection*, en cherchant à rapprocher une définition abstraite du dictionnaire au phénomène bruité et flou que constitue son occurrence dans un "vrai" contexte.

Ces réflexions sur le concept de sens ne sont pas sans rappeler le slogan wittgensteinien, "la signification c'est l'usage", dont les fondements ont été expliqués au premier chapitre. Au lieu de forcer nos systèmes à s'exécuter dans le cadre rigide et dogmatique de nos ontologies, laissons-les plutôt déterminer eux-mêmes ce qui constitue les frontières des classes d'usage des mots et des concepts.

5.4 Les dés sont jetés dès l'initialisation...

Remarquons en dernier lieu un autre fait troublant par rapport à l'entraînement de nos modèles. Il semble que le mécanisme d'apprentissage ne soit guère utile, car la plus grande part de la diminution de la perplexité est attribuable à l'optimisation de la mixture [5]. La conclusion qu'il est possible de tirer de ceci est que l'initialisation fait le gros du travail, et qu'il est par conséquent très difficile d'améliorer la distribution initiale, pratiquement optimale, et à peu près équivalente à celle d'un modèle de langage sans variable cachée. Ceci est évidemment gênant, et fait planer un doute légitime sur l'ensemble de la procédure. Les modèle ont-ils la capacité nécessaire ainsi que l'accès à des données suffisament informatives pour leur permettre d'augmenter itérativement la qualité de leur représentation ? Ou ne font-ils au contraire que tourner à vide, en tentant désespérément de capter ce qui est hors de leur portée ?

le sens le plus commun du mot "*barrage*", qu'on peut déterminer dans un nombre surprenant de cas à l'aide de la simple détection du verbe "construire" dans son voisinage !

[5] Ceci est aisément vérifiable en bloquant simplement la mise à jour itérative des poids de la mixture, et en constatant que l'apprentissage ralentit ainsi considérablement.

Chapitre 6

Conclusion

(GOODMAN 2001) conclut son étude sur une note presque humoristique ("All hope abandon, ye who enter here"), en faisant remarquer que la modélisation du langage est un domaine de recherche particulièrement coriace ("not for the faint of heart or easily depressed"), avec lequel il est de plus en plus difficile de faire des percées significatives. Une conception couramment admise est qu'il est très difficile de battre un modèle trigramme à l'état de l'art actuel. Cette croyance n'est certainement pas sans fondement, car bien qu'il ne soit pas possible de parvenir à faire autre chose que la réfuter (car on ne pourrait certainement pas prétendre à l'exhaustion de tous les modèles possibles !), elle repose néanmoins sur de puissants arguments expérimentaux. Et si quelques études parviennent à montrer qu'il est possible de le faire, il reste que c'est souvent au prix de coûteux compromis en espace ou en temps de calcul, pour des gains modestes. Les conclusion de cette étude ne peuvent malheureusement qu'abonder en ce même sens, étant donné que nous avons échoué dans notre tentative de montrer qu'un modèle de langage dont la capacité de représentation est augmentée à l'aide des mécanismes proposés pouvait battre de simples modèles interpolés. En dépit de ces résultats négatifs, nous croyons néanmoins qu'un de ses modestes mérites aura été d'avoir quelque peu aidé à défricher un terrain relativement nouveau et difficile, à la frontière de plusieurs domaines tout de même prometteurs.

6.1 Contributions expérimentales et pratiques

Bien que l'accent ait été moindre sur cet aspect, il reste qu'il subsistera de cette étude d'importantes contributions techniques l'ayant soutenue et rendue possible. Un système complexe, composé de nombreux programmes de prétraitement et de construction de la représentation, des interfaces conviviales simplifiant l'accès à des systèmes externes figurent ainsi au nombre des objets informatiques pour lesquels de nombreux mois de développement auront été nécessaires, et qui pourront peut-êtte se voir réutilisés, et servir ainsi dans des contextes de recherche différents.

6.2 Contributions théoriques

Cette étude aura en outre permis l'exploration d'un cadre moins rigide pour la modélisation du langage statistique, en détournant l'attention des phénomènes de surface (l'occurrences des mots) pour plonger au coeur d'un aspect plus profond du langage, la distribution des sens et des concepts. Ce degré de liberté supplémentaire offrait l'espoir d'une robustesse et d'une puissance de généralisation accrues, et devait faire en sorte de lisser la distribution des mots de manière plus adaptée et informée. Le fait que les phénomènes sur lesquels nous voulions porter notre attention ne soient pas observables nous a poussé a développer une méthodologie d'optimisation stochastique fortement inspirée de recherches actuelles, s'articulant dans d'autres secteurs de l'apprentissage statistique. Bien que ces techniques d'aprentissage non-supervisé n'aient pas produit les résultats escomptés dans le présent contexte, elles pourraient néanmoins paver la voie à des idées similaires futures. Finalement, nous avons montré comment il était possible d'harnacher une source de connaissances à priori telle que WordNet à des modèles statistiques foncièrement "ignorants", n'ayant d'autres ressources à leur disposition que les phénomènes de surface très bruités qui composent un corpus d'entraînement. Les quelques critiques et observations que nous nous sommes permis d'émettre sur cet outil devraient en outre permettre de prévoir ou de prévenir certains pièges qui guettent ses utilisateurs.

Par l'inclusion et la discussion de certaines idées à caractère plus philosophique au sein de ce mémoire, nous espérons finalement avoir suggéré l'importance d'une ouverture à l'égard de toutes les disciplines qui tentent de comprendre et d'expliquer le fonctionnement du langage et de l'intelligence, souvent avec des approches et des idées très différentes. Le problème est difficile et profond, et l'attaque sur un front unique est peut-être irrémédiablement vouée à l'échec.

Références

AGIRRE, E. et G. RIGAU (1996), « Word sense disambiguation using conceptual density », *Proceedings of COLING'96*, p. 16–22.

AITCHISON, J. (1998), *The Articulate Mammal : An Introduction to Psycholinguistics* (Fourth ed.)., London : Routledge.

BAUM, L. E. (1972), « An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process », *Inequalities 3*, p. 1–8.

BENGIO, Y. (2002), « New Distributed Probabilistic Language Models », Rapport technique 1215, Dept. IRO, Université de Montréal.

BENGIO, Y., R. DUCHARME, P. VINCENT et C. JAUVIN (2003), « A Neural Probabilistic Language Model », *Journal of Machine Learning Research 3*, p. 1137–1155.

BENGIO, Y. et J.-S. SENÉCAL (2003), « Quick Training of Probabilistic Neural Nets by Importance Sampling », *AISTATS*,

BILMES, J. (1997), « A Gentle Tutorial on the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models », Rapport technique ICSI-TR-97-021, University of Berkeley.

BROWN, P. F., J. COCKE, S. D. PIETRA, V. J. D. PIETRA, F. JELINEK, J. D. LAFFERTY, R. L. MERCER et P. S. ROOSSIN (1990), « A Statistical Approach to Machine Translation », *Computational Linguistics 16*(2), p. 79–85.

BROWN, P. F., V. J. D. PIETRA, P. V. DESOUZA, J. C. LAI et R. L. MERCER (1992), « Class-Based n-gram Models of Natural Language »,

Computational Linguistics 18(4), p. 467–479.

BUDANITSKY, A. (2000), « Semantic Distance in WordNet : An Experimental, Application-oriented Evaluation of Five Measures ».

CHEN, S., D. BEEFERMAN et R. ROSENFELD (1998), « Evaluation metrics for language models », *DARPA Broadcast News Transcription and Understanding Workshop*,

CHEN, S. F. et J. T. GOODMAN (1998), « An Empirical Study of Smoothing Techniques for Language Modeling », Rapport technique TR-10-98, Computer Science Group, Harvard University.

CHEN, S. F. et J. T. GOODMAN (1999), « An Empirical Study of Smoothing Techniques for Language Modeling », *Computer, Speech and Language 13*(4), p. 359–393.

CHOMSKY, N. (1957), *Syntactic Structures*, The Hague : Mouton.

CHURCH, K. (1988), « A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text », *Proceedings of the Second Conference on Applied Natural Language Processing*, p. 136–143.

DAGAN, I. et A. ITAI (1994), « Word Sense Disambiguation Using a Second Language Monolingual Corpus », *Computational Linguistics 20*(4), p. 563–596.

DEMPSTER, A. P., N. M. LAIRD et D. B. RUBIN (1977), « Maximum-likelihood from incomplete data via the EM algorithm », *Journal of Royal Statistical Society B 39*, p. 1–38.

DIAB, M. et P. RESNIK (2002), « An unsupervised method for word sense tagging using parallel corpora », *40th Annual Meeting of the ACL*,

DOUGLAS B. LENAT, MAYANK PRAKASH, M. S. (1986), « CYC : Using common sense knowledge to overcome brittleness and knowledge acquisition bottlenecks », *AI Magazine 6*(4), p. 65–85.

FELLBAUM, C. (1998), *WordNet : An Electronic Lexical Database*. MIT Press.

GALE, W., K. CHURCH et D. YAROWSKY (1992), « A Method for Disambiguating Word Senses in a Large Corpus », *Computers and the Humanities 26*, p. 415–439.

GOOD, I. (1953), « The population frequencies of species and the estimation of population parameters », *Biometrika 40*(3 and 4), p. 129–264.

GOODMAN, J. (2001), « A Bit of Progress in Language Modeling, Extended Version », Rapport technique, Machine Learning and Applied Statistics Group, Microsoft Research, Redmond, WA.

HARNAD, S. (1990), « The Symbol Grounding Problem », *Physica 42*, p. 335–346.

HUANG, X., F. ALLEVA, H.-W. HON, M.-Y. HWANG et R. ROSENFELD (1993), « The SPHINX-II speech recognition system : an overview », *Computer Speech and Language* 7(2), p. 137–148.

IDE, N. et J. VÉRONIS (1998), « Introduction to the special issue on word sense disambiguation : the state of the art », *Computational Linguistics 24*, p. 1–40.

IYER, R., M. OSTENDORF et J. ROHLICEK (1994), « Language modeling with sentence-level mixtures ».

JEFFREYS, H. (1948), *Theory of Probability* (Second ed.)., Oxford : Clarendon Press.

JELINEK, F. (1997), *Statistical Methods for Speech Recognition*. MIT Press.

JELINEK, F. et R. MERCER (1980), « Interpolated estimation of Markov source parameters from sparse data », *Pattern Recognition in Practice*. North-Holland, Amsterdam,

JOHNSON, W. (1932), « Probability : deductive and inductive problems », *Mind 41*, p. 421–423.

JORDAN, M. (1998), *Learning in Graphical Models*, Dordrecht, Netherlands : Kluwer.

KATZ, S. M. (1987, March), « Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer », *IEEE Transactions on Acoustics, Speech, and Signal Processing ASSP-35* (3), p. 400–401.

KERNIGHAN, M., K. CHURCH et W. GALE (1990), « A Spelling Correction Program Based on a Noisy Channel Model », *Proceedings of the Thir-*

teenth International Conference on Computational Linguistics, p. 205–210.

KNESER, R. et H. NEY (1995), « Improved backing-off for n-gram language modelling », *Proceedings of IEEE ICASP'95*, Detroit, p. 49–52.

KUHN, R. et R. DEMORI (1990), « A cache-based natural language model for speech recognition », *IEEE Transactions on Pattern Analysis and Machine Intelligence 12*(6), p. 570–583.

LESK, M. E. (1986), « Automatic Sense Disambiguation Using Machine Readable Dictionaries : How to Tell a Pine Cone from an Ice Cream Cone », *Proceedings of the 1986 SIGDOC Conference*, p. 24–26.

LIDSTONE, G. (1920), « Note on the general case of the Bayes-Laplace formula for inductive or a posteriori probabilities », *Transactions of the Faculty of Actuaries 8*, p. 182–192.

MARTIN, S., H. NEY et J. ZAPLO (1999), « Smoothing Methods in Maximum Entropy Language Modeling ».

NEY, H., U. ESSEN et R. KNESER (1994), « On structuring probabilistic dependences in stochastic language modelling », *Computer Speech and Language 8*, p. 1–38.

PONTE, J. M. et W. B. CROFT (1998), « A Language Modeling Approach to Information Retrieval », *Research and Development in Information Retrieval*, p. 275–281.

ROSENFELD, R. (1994), *Adaptive Statistical Language Modeling : A Maximum Entropy Approach*, Ph. D. thesis, Carnegie-Mellon University, Aarhus, Denmark.

ROSENFELD, R. (2000), « Two decades of statistical language modeling : Where do we go from here », *Proceedings of the IEEE*,

SALTON, G. et M. MCGILL (1983), *Introduction to Modern Information Retrieval*, New York : McGraw-Hill.

SCHANK, R. C. (1972), « Conceptual Dependency : A Theory of Natural Language Understanding », *Cognitive Psychology 3*(4), p. 532–631.

SCHANK, R. C. et R. P. ABELSON (1977), *Scripts, Plans, Goals and Understanding : an Inquiry into Human Knowledge Structures*, Hillsdale,

NJ : L. Erlbaum.

SEARLE, J. (1980), « Minds, Brains, and Programs », *Behavioral and Brain Sciences 3*, p. 417–424.

SIU, M. et M. OSTENDORF (2000), « "Variable N-gram Language Modeling and Extensions for Conversational Speech », *IEEE transaction on Speech and Audio Processing 8*(2), p. 63–75.

SOWA, J. F. (1976), « Conceptual graphs for a database interface », *IBM Journal of Research and Development 20*(4), p. 336–357.

SRIHARI, R. K. et C. M. BALTUS (1993), « Incorporating Syntactic Constraints in Recognizing Handwritten Sentences », *Proceedings of the 13th IJCAI*, Chambery, France, p. 1262–1268.

TURING, A. M. (1950), « Computing Machinery and Intelligence », *Mind 59*(236), p. 433–460.

VÉRONIS, J. (2001), « Sense tagging : does it make sense ? ».

WHITEHEAD, A. N. et B. RUSSELL (1962), *Principia Mathematica*, New York : Cambdridge University Press.

WILKS, Y. (1998), « Is Word-sense disambiguation just one more NLP task ? », *Proceedings of SENSEVAL Conference*,

WITTEN, I. et T. BELL (1991), « The zero-frequency problem : Estimating the probabilities of novel events in adaptive text compression », *IEEE Transactions on Information Theory 37*(4), p. 1085–1094.

WITTGENSTEIN, L. (1922), *Tractatus Logico-Philosophicus*, London : Routledge & Kegan Paul Ltd.

WITTGENSTEIN, L. (1958), *Philosophical Investigations* (Second ed.)., Oxford : Basil Blackwell.

ZIPF, G. K. (1949), *Human Behaviour and the Principle of Least-Effort*, Cambridge, MA : Addison-Wesley.