Modèles neuronaux pour la recherche d'information : approches dirigées par les ressources sémantiques

Gia-Hung Nguyen

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *18/12/2018* **par :**

GIA-HUNG NGUYEN

Modèles neuronaux pour la recherche d'information : approches dirigées par les ressources sémantiques

---

JURY

BRIGITTE GRAU	Professeur, Ecole nationale supérieure d'Informatique pour l'industrie et l'entreprise	Rapportrice
MASSIH-REZA AMINI	Professeur, Université Grenoble Alpes	Rapporteur
ANTOINE DOUCET	Professeur, Université de La Rochelle	Examinateur
MOHAND BOUGHANEM	Professeur, Université Toulouse 3	Invité
LYNDA TAMINE	Professeur, Université Toulouse 3	Directrice de thèse
NATHALIE SOUF	MCF, Université Toulouse 3	Co-directrice
LAURE SOULIER	MCF, Sorbonne Université	Encadrante

École doctorale et spécialité :
MITT : Informatique et Télécommunications
Unité de Recherche :
Institut de Recherche en Informatique de Toulouse (UMR 5505)
Directeur(s) de Thèse :
Lynda TAMINE et *Nathalie SOUF*
Rapporteurs :
Brigitte GRAU et *Massih-Reza AMINI*

Modèles neuronaux pour la recherche d'information

Approches dirigées par les ressources sémantiques

Gia-Hung Nguyen

Novembre 2018

REMERCIEMENTS

Je souhaite avant tout exprimer mes sincères remerciements à *Lynda Tamine-Lechani*, ma directrice de thèse, pour sa confiance et ses nombreux encouragements. Son soutien permanent, ses conseils avisés m'ont été d'une grande aide tout au long de la thèse. Sans son encadrement bienveillant et ses idées, ce travail n'aurait pu voir le jour. Je souhaite exprimer ma sincère reconnaissance et mes remerciements à ma co-encadrante *Nathalie Souf* pour sa présence, ses relectures minutieuses et ses conseils qu'elle m'a prodigués tout au long de ces années de thèse. Je tiens à exprimer ma gratitude à ma deuxième co-encadrante *Laure Soulier* qui est aussi ma tutrice depuis les premiers jours à l'IRIT. Sa bienveillance persistante malgré la distance et ses nombreux conseils m'ont permis de franchir tous les challenges dans la thèse. Je la remercie également pour son écoute et ses encouragements qui m'ont aidé à passer les moments les plus difficiles.

Je remercie très sincèrement Professeur *Brigitte Grau* et Professeur *Massih-Reza Amini* pour avoir accepté d'être rapporteurs de ma thèse. Je tiens également à remercier les examinateurs, Professeur *Mohand Boughanem* et Professeur *Antoine Doucet* d'avoir accepté de participer à mon jury de thèse.

Je souhaite exprimer ma gratitude à M. *Gilles Hubert*, pour m'avoir accueilli au sein de l'équipe IRIS et pour avoir créé une telle ambiance sympathique dans nos bureaux. Je tiens à remercier l'ensemble des membres de l'équipe IRIS pour les échanges, les discussions ainsi que leur gentillesse. Merci *José* pour tes conseils professionnels. Merci également à l'ensemble de mes amis au bureau, *Amjed, Bilel, Hamid, Ameni, Meriam, Mr.KandyCrush, Thiziri, Rafik, ...*, à tous ceux qui ont contribué à récréer ma vie de thésard.
Paul, vu tout ce que l'on a eu ensemble au bureau et ailleurs, est-ce que je peux nous appeler "Partenaire Particulier" ?

Je tiens à remercier également tous mes amis à Toulouse avec qui nous avons passé de bons moments ensemble : *Chi Ngoc* et sa petite famille, *Chi Diep*, *Anh Hoa*, mes *camarades* à INSA de Toulouse, ... Merci *Bac Thoai* de m'avoir accueilli et de m'avoir donné une deuxième famille en France.

Enfin, mes remerciements sont dédiés à mon père, ma mère et mon frère pour leur soutien, leur confiance et leur patience qui m'ont donné l'assurance d'aller jusqu'ici et encore plus loin...

RÉSUMÉ

Les modèles de recherche d'information (RI) classiques se basent sur la théorie probabiliste en supposant l'indépendance des termes lors de la représentation et l'appariement des documents et des requêtes. Cependant, cette hypothèse ne tient pas dans la réalité du langage naturel où les termes sont liés par des relations sémantiques. En conséquence, cette limite entraîne un écart entre la représentation des documents et le langage de la requête exprimant le besoin de l'utilisateur. Ce problème est connu en RI sous le nom de "fossé sémantique", qui est l'une des principales raisons de la dégradation des performances d'un SRI (Crestani, 2000). De nombreux travaux en RI ont montré que l'utilisation des sources d'évidence provenant des ressources sémantiques externes pourrait améliorer la performance de l'appariement. Par ailleurs, les approches neuronales sont devenues récemment des modèles de référence qui permettent de capturer à partir des corpus, la sémantique latente des mots qui peut être injectée dans les modèles RI.

Nos travaux de thèse s'inscrivent dans ce contexte en visant à réduire le fossé sémantique dans les représentations et l'appariement des documents et des requêtes. Nous proposons de combiner la sémantique relationnelle issue des ressources sémantiques et la sémantique distribuée dans le corpus apprise par les modèles neuronaux. Nos contributions consistent en deux volets principaux : 1) amélioration des représentations de textes aux fins des tâches de RI ; et 2) exploitation des réseaux de neurones pour un appariement sémantique de documents-requêtes.

Concernant l'amélioration des représentations de textes, nous proposons deux approches qui intègrent la sémantique relationnelle dans l'apprentissage de représentations distribuées : a) un modèle hors ligne qui combine deux types de représentations pré-entraînées pour obtenir une représentation finale du document ; b) un modèle en ligne qui apprend conjointement les représentations de textes à plusieurs niveaux (mots, concepts, documents). Pour mieux capturer la sémantique relationnelle dans les représentations de textes, nous proposons d'intégrer les contraintes de relations entre les mots/concepts dans ces deux méthodes d'apprentissage. Deux approches sont utilisées pour injecter ces contraintes relationnelles, une basée sur la régularisation de la fonction objectif, une basée sur les instances dans le texte d'entraînement.

Concernant l'utilisation des réseaux de neurones pour la RI, nous proposons un modèle neuronal pour l'appariement des paires de document-requête en tenant compte des représentations sémantiques en entrée. Ce modèle consiste en un réseau de neurones siamois qui apprend une fonction d'appariement des documents en utilisant des vecteurs d'entrée combinant à la fois les représentations distribuées et les représentations basées sur les ressources sémantiques externes. Dans ce modèle, nous proposons également une méthode pour construire une représentation symbolique de texte, qui s'appuie sur les concepts et leurs relations dans une ressource externe. Cette représentation vise à capturer la sémantique relationnelle du document en projetant son contenu conceptuel dans la hiérarchie de la ressource externe.

L'ensemble de nos contributions, en termes de représentations et appariement des documents/requête ont fait l'objet d'évaluation expérimentale sur des tâches dédiées à évaluer à la fois la qualité des représentations de textes à plusieurs niveaux, et l'efficacité de leur application en RI. Les expérimentations sont menées sur les jeux de données TREC du domaine générique comme Robusto4, GOV, ainsi que du domaine médical comme PubMed, OHSUMED, TRED Med.

ABSTRACT

Tackling the vocabulary mismatch has been a long-standing and major goal in information retrieval (IR). To infer and match discrete word senses within the context of documents and queries being matched, one line of work makes use of hand-labeled external knowledge resources such as linguistic resources and knowledge graphs. Such resources allow exploiting the objects and their relations (e.g., synonymy, hyponymy) within, e.g., query or document expansion to lower the vocabulary mismatch between queries and documents ; this is referred to as the *relational semantics*. Another line of work attempts to automatically infer hidden word senses from corpora using word collocations by performing dimensionality reduction techniques, such as distributed representation learning, leading to *distributional semantics*.

In this thesis, we focus on bridging the semantic gap between the documents and queries representations, hence improve the matching performance. We propose to combine relational semantics from knowledge resources and distributed semantics of the corpus inferred by neural models. Our contributions consist of two main aspects :

— *Improving distributed representations of text for IR tasks*. We propose two models that integrate relational semantics into the distributed representations : a) an offline model that combines two types of pre-trained representations to obtain a hybrid representation of the document ; b) an online model that jointly learns distributed representations of documents, concepts and words. To better integrate relational semantics from knowledge resources, we propose two approaches to inject these relational constraints, one based on the regularization of the objective function, the other based on instances in the training text.

— *Exploiting neural networks for semantic matching of documents*. We propose a neural model for document-query matching. Our neural model relies on : 1) a representation of raw-data that models the relational semantics of text by jointly considering objects and relations expressed in a knowledge resource, and 2) an end-to-end neural architecture that learns the query-document relevance by leveraging the distributional and relational semantics of documents and queries.

PUBLICATIONS

Articles publiés dans des conférences et workshops internationaux

1. **Gia-Hung Nguyen**, Lynda Tamine, Laure Soulier, Nathalie Souf. *A Tri-Partite Neural Document Language Model for Semantic Information Retrieval*. Dans : European Semantic Web Conference (ESWC 2018), juin 2018, p. 445-461.
2. **Gia-Hung Nguyen**, Laure Soulier, Lynda Tamine, Nathalie Souf. *DSRIM : A Deep Neural Information Retrieval Model Enhanced by a Knowledge Resource Driven Representation of Documents*. Dans : International Conference on the Theory of Information Retrieval (ICTIR 2017), octobre 2017, p. 19-26.
3. **Gia-Hung Nguyen**, Lynda Tamine, Laure Soulier, Nathalie Souf. *Learning Concept-Driven Document Embeddings for Medical Information Search*. Dans : Conference on Artificial Intelligence in Medicine (AIME 2017), juin 2017, p. 160-170.
4. **Gia-Hung Nguyen**, Lynda Tamine, Laure Soulier, Nathalie Souf. *Toward a Deep Neural Approach for Knowledge-Based IR* (poster). Dans : Workshop on Neural Information Retrieval, in conjunction with the ACM SIGIR Conference (NEU-IR 2016), juillet 2016.

Articles publiés dans des conférences nationales

1. **Gia-Hung Nguyen**, Lynda Tamine, Laure Soulier, Nathalie Souf. *Modèle neuronal tripartite pour la représentation de documents*. Dans : Conférence francophone en Recherche d'Information et Applications (CORIA 2018), mai 2018.
2. **Gia-Hung Nguyen**, Lynda Tamine, Laure Soulier, Nathalie Souf. *Apprentissage de représentation des documents médicaux guidé par les concepts pour la recherche d'information*. Dans : Symposium sur l'Ingénierie de l'Information Médicale (SIIM 2018), novembre 2017.
3. **Gia-Hung Nguyen**, Laure Soulier, Lynda Tamine, Nathalie Souf. *Modèle Neuronal de Recherche d'Information Augmenté par une Ressource Sémantique*. Dans : Conférence francophone en Recherche d'Information et Applications (CORIA 2017), mars 2017, p. 265-283.

Participation à des campagnes d'évaluation internationales

1. **Gia-Hung Nguyen**, Laure Soulier, Lynda Tamine, Nathalie Souf. *IRIT @ TREC 2016 Clinical Decision Support Track*. Dans : Working notes for TREC CDS 2016, novembre 2016.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

1

CONTEXTE ET CONTRIBUTION DE LA THÈSE

1 Contexte et problématique

1.1 *Contexte de la thèse*

La recherche d'information (RI) est un domaine de l'informatique qui permet de sélectionner à partir d'une collection de documents ceux qui sont susceptibles de répondre au besoin de l'utilisateur exprimé via une requête (Salton, 1968). Ce processus se déroule au sein d'un système de recherche d'informations (SRI), permettant d'effectuer l'ensemble des tâches nécessaires à la RI. Un SRI possède trois fonctions fondamentales qui définissent le modèle de recherche : représenter le contenu des documents, représenter le besoin de l'utilisateur et comparer ces deux représentations pour calculer un score de pertinence entre le document et la requête. Le défi principal de la RI est la modélisation pour traduire théoriquement la notion de pertinence.

Depuis les années 1990, le développement sans cesse de la technologie d'information facilite progressivement l'échange des données. Grâce aux avantages d'Internet (e. g., la facilité de stockage, de recherche, de partage) les documents sont de plus en plus numérisés et diffusés. L'émergence de ces données sur l'Internet s'est traduite par une source massive d'information dont la quantité et la qualité qui ne cessent de se multiplier. Face à ce volume massif de données, particulièrement les données textuelles, les modèles de RI classique, qui estiment le score de pertinence en se basant sur l'indépendance des mots, ont montré des limites dans l'efficacité de recherche. Les documents renvoyés par des SRI ne traitent pas toujours de l'intention de recherche exprimée dans la requête de l'utilisateur. Cette limite est due à un écart entre la représentation des documents, qui utilise souvent l'approche par sacs de mots, et le langage exprimant le besoin de l'utilisateur via une requête.

Ce problème est connu en RI sous le nom de "fossé sémantique" ("*semantic gap*" en anglais). Ce fossé est l'une des principales raisons du défaut d'appariement entre requête et document qui conduit à la dégradation des performances d'un SRI (Crestani, 2000). Le fossé sémantique provient généralement de : 1) la *discordance de vocabulaire*, ce qui signifie que des mots de formes différentes partagent le même sens (e. g., **aperçu** est synonyme de **sommaire**) ; 2) la *discordance de granularité*, ce qui signifie que des mots de formes et de sens différents appartiennent au même concept général (e. g., **chat** et **chien** sont des **animaux**) ; 3) la *polysémie*, ce qui signifie qu'un mot peut couvrir différents sens en fonction des mots qui l'entourent dans le texte et qui représentent son contexte (e. g., **pêche** peut signifier un fruit ou l'action de pêcher).

Pour pallier le problème du fossé sémantique, nous distinguons dans la littérature de la RI, deux principales catégories d'approches de RI dite "sémantique" :

— Approches utilisant des ressources externes structurées qui constituent un référentiel de connaissances établies par des humains, comme les thesaurus, les ontologies, les terminologies, graphes de connaissances, etc. Ces ressources fournissent des informations sur les sens des mots et de leurs relations comme la synonymie, l'homonymie, l'hyponymie, etc. Cette approche correspond à l'usage de la "sémantique relationnelle".

— Approches utilisant des ressources non structurées, essentiellement des corpus de textes. Ces derniers sont exploités pour l'hypothèse distributionnelle qui indique que deux mots qui occurrent dans les mêmes contextes ont tendance à avoir le même sens. Cette hypothèse de "sémantique distributionnelle" a permis d'abord l'émergence de travaux ayant visé la réduction de l'espace du vocabulaire vers un espace latent de concepts construit par comptage de cooccurrences de mots ; on retrouve les modèles comme LSA (Deerwester et al., 1990) et PLSA (Hofmann, 1999). Plus récemment, l'hypothèse distributionnelle a fait émerger des approches neuronales qui ont connu un grand succès (Mikolov et al., 2013a; Pennington et al., 2014).

Dans le cadre de cette thèse, nous nous intéressons particulièrement aux approches de sémantique relationnelle et sémantique distributionnelle utilisant des réseaux de neurones.

• Sémantique relationnelle

Comme indiqué plus haut, la sémantique relationnelle est fondée sur l'utilisation de ressources externes qui inventorient un référentiel de mots-termes et relations. Ces ressources sémantiques consistent en des structures standardisées qui contiennent des informations sur un domaine général comme la linguistique ou spécifique comme la médecine. Ces ressources sémantiques peuvent prendre la forme d'une base lexicale comme WordNet (Miller, 1995), d'un thésaurus comme MeSH (*Medical Subject Headings*), d'une ontologie comme *Gene Ontology*, etc. Une

ressource est souvent construite avec des concepts, qui représentent une entité principale, et leurs informations comme la définition, les caractéristiques, la relation avec les autres concepts, etc. En exploitant ces ressources, on peut mieux capturer le sens d'un terme ainsi que les termes reliés via des relations.

Plusieurs travaux ont exploité des connaissances à travers les concepts et leurs relations définis dans des ressources sémantiques (e. g., base lexicale, thésaurus, ontologie) qualifiées de *sémantique relationnelle*. Dans le contexte de cette thèse, nous nous intéressons particulièrement à améliorer : 1) la représentation des requêtes, 2) la représentation des documents, et 3) l'appariement requêtes-documents.
La première catégorie de modèles intègre la connaissance des ressources sémantiques au niveau de la représentation des requêtes (Hersh et al., 2000; Stokes et al., 2009; Pal et al., 2014; Xiong and Callan, 2015b). Plus spécifiquement, ces modèles améliorent la représentation de la requête initiale d'un utilisateur en y ajoutant des termes utiles, dans le but d'augmenter le rappel. Généralement, les termes d'expansion sont automatiquement sélectionnés à partir d'un ensemble de documents renvoyés par une évaluation initiale de la requête (Lavrenko and Croft, 2017; Rocchio, 1971; Zhai and Lafferty, 2001), qui sont supposés être pertinents.
Les modèles dans la deuxième catégorie exploitent les ressources pour améliorer la représentation des documents (Gobeill et al., 2008; Agirre et al., 2010; Baziz et al., 2005; Gupta et al., 2017). L'intuition est d'augmenter le rappel et la précision du résultat en ajoutant aux représentations des documents les termes utiles qui sont absents dans la requête. Cette expansion conceptuelle des documents aide à améliorer la précision par l'indexation avec les concepts au lieu des termes ambigus.
La troisième catégorie de travaux vise à améliorer la fonction d'appariement document-requête en utilisant les connaissances issues des ressources externes (L'Hadj et al., 2016; Xiong and Callan, 2015a; Koopman et al., 2016). L'idée sous-jacente est de plutôt modéliser un mécanisme d'appariement conceptuel le document et la requête, par exemple avec des approches de traversée du graphe.

- **Sémantique distributionnelle basée sur les approches neuronales**

Une lignée récente de travaux utilise principalement les modèles neuronaux pour en faire émerger la sémantique latente, appelée *sémantique distributionnelle*. Ces modèles exploitent les réseaux de neurones pour apprendre des vecteurs de représentations ou une fonction d'appariement des textes. En effet, les réseaux de neurones artificiels sont conçus pour reconnaître des modèles de comportement dans les données. Les avancées récentes des modèles neuronaux ont stimulé des améliorations considérables dans des domaines d'applications tels que la vision par ordinateur et la traduction automatique. Après les succès dans le domaine de vision par ordinateur, les progrès en traitement de langue ont donné lieu aux modèles neuronaux pour l'apprentissage des vecteurs de représentation distribuée

des mots (Bengio et al., 2003; Pennington et al., 2014; Mikolov et al., 2013a) et des phrases ou documents (Le and Mikolov, 2014; Hill et al., 2016). Une *représentation distribuée* d'un mot est un vecteur qui encode la sémantique de ce mot. En capturant la mesure dans laquelle les mots se produisent dans des contextes similaires, ces vecteurs de représentation sont capables d'encoder la similarité sémantique et syntaxique dans la mesure où les représentations des mots similaires seront proches les unes des autres dans l'espace vectoriel. En associant des mots et d'autres unités textuelles à leurs représentations, les appariements peuvent être calculés dans l'espace de représentation pour mieux capturer la sémantique de ces textes.

Dans le contexte de notre thèse, pour mieux réduire le fossé sémantique, une lignée de travaux utilisent les représentations distribuées dans les approches de RI existantes, selon deux niveaux principaux : l'expansion des requêtes (Nalisnick et al., 2016; Mitra et al., 2016; Zuccon et al., 2015; Rubner et al., 2000) et l'appariement de document-requête (Roy et al., 2016; Zamani and Croft, 2016a; Diaz et al., 2016; Zamani and Croft, 2016b). La première approche consiste à utiliser des représentations distribuées de mots pour trouver de bons candidats à l'expansion de la requête. Cette approche consiste à comparer individuellement le terme candidat à chaque terme de la requête en utilisant leurs représentations vectorielles, puis les scores sont agrégés avec les scores de RI classique pour trouver les meilleurs candidats d'expansion. Dans la deuxième approche, le but des modèles consiste à dériver une représentation vectorielle dense pour la requête et le document dans l'espace de représentation distribuée. Puis la requête et les documents sont comparés à l'aide d'une variété de mesures de similarité, comme la similarité cosinus entre deux vecteurs de représentation.
Une autre lignée de travaux utilise des architectures de neurones profondes pour apprendre une fonction d'appariement, qui se réfère à l'apprentissage d'ordonnancement (Guo et al., 2016; Huang et al., 2013; Shen et al., 2014a; Hu et al., 2014). Ces modèles neuronaux de RI s'appuient sur les informations de pertinence pour apprendre à ordonnancer les requêtes et les documents. Il existe deux catégories principales de travaux : les modèles basés sur la représentation et les modèles basés sur les interactions. La première catégorie, basée sur la représentation, a pour objectif de construire une bonne représentation pour un texte unique avec un réseau neuronal profond, puis procède à l'appariement entre deux représentations textuelles compositionnelles et abstraites (Huang et al., 2013; Shen et al., 2014a,b; Hu et al., 2014; Palangi et al., 2016; Severyn and Moschitti, 2015). La deuxième catégorie, basée sur les interactions, construit d'abord les interactions locales entre deux textes à partir des représentations de base, puis utilise des réseaux de neurones profonds qui apprennent les modèles d'interaction hiérarchique pour l'appariement. (Lu and Li, 2013; Hu et al., 2014; Guo et al., 2016; Mitra et al., 2017; Pang et al., 2016).

1.2 *Problématique de la thèse*

Nos travaux de thèse s'inscrivent dans ce contexte, où l'utilisation des approches neuronales pour apprendre les représentations et l'appariement des textes permet de capturer la sémantique latente des textes à travers le corpus. Cette sémantique latente enrichit la représentation des textes et aide à réduire le fossé sémantique dans le processus de RI. Cependant la sémantique distribuée présente des limites : (1) elle ne permets pas de lever le problème de polysémie puisque tous les sens d'un même mot sont représentés dans un seul vecteur ; en revanche ces sens sont bien distingués dans une ressource structurée ; (2) des similarités explicites entre mots telles qu'elles sont établies dans une ressource externe peuvent ne pas l'être par l'approche de comptage distributionnel si leur apparition dans les mêmes contextes est insuffisante dans le corpus ; (3) des vecteurs distributionnels de mots peuvent s'avérer peu lisibles en ce sens qu'ils ne sont pas alignables avec des ressources externes ; à titre d'exemple, Mrkšić et al. (2016) ont montré que le mot "*cheaper*" se retrouve dans les mots plus proches du mot "*expensive*", en utilisant le vecteur de représentation Glove (Pennington et al., 2014). Pour cela la motivation de nos travaux est guidée par l'hypothèse de complémentarité entre ces deux types de sémantique. Ainsi, nous proposons d'exploiter à la fois la *sémantique relationnelle* et la *sémantique distributionnelle* pour améliorer les performances de la RI. Pour ce faire, il existe des verrous à lever comme : l'alignement de la sémantique relationnelle avec la sémantique distributionnelle ; l'alignement des termes et des concepts dans un processus d'apprentissage ; l'amélioration de l'apprentissage de représentations distribuées aux fins d'efficacité de RI.

Dans ces travaux, nous nous intéressons principalement à cette question de recherche : Comment combiner la sémantique relationnelle exprimée dans les ressources externes avec la sémantique distributionnelle issue de l'apprentissage neuronal du texte pour améliorer les performances de RI ? Pour cela, nous nous concentrons sur les deux sous-questions de recherche suivantes :

1. Comment exploiter au mieux la sémantique relationnelle et la sémantique distributionnelle pour améliorer la représentation des textes à des fins de RI.
   a) Comment exploiter les concepts issus d'une ressource externe dans l'apprentissage pour obtenir de meilleures représentations distribuées de documents et leurs composants (mots, concepts).
   b) Comment intégrer les contraintes de relations entre des mots/concepts pour mieux apprendre les représentations de textes.
2. Comment intégrer la sémantique relationnelle et la sémantique distributionnelle dans un modèle d'apprentissage d'ordonnancement pour améliorer l'efficacité de la RI ?

a) Quelle représentation des requêtes et des documents à l'entrée du réseau de neurones, qui intègre au mieux la connaissance issue des ressources sémantiques ?
b) Comment modéliser l'appariement document-requête par un réseau de neurones ?

2 Contributions

La principale contribution de cette thèse est la définition de modèles de représentation de texte et d'appariement en RI augmentés par des ressources sémantiques. Pour cela, nous adoptons deux approches pour combiner la sémantique relationnelle et la sémantique distributionnelle en vue de la RI. La première approche s'intéresse à améliorer la représentation de textes en injectant des connaissances et des contraintes relationnelles issues des ressources externes. La seconde approche se concentre sur un apprentissage de l'appariement des paires de document-requête étant données des représentations guidées par la ressource externe. Nos contributions sont les suivantes :

1. *Augmenter les représentations distribuées de textes.* Nous proposons deux approches qui intègrent la sémantique relationnelle dans l'apprentissage de représentations distribuées : 1) un modèle hors ligne qui combine deux types de représentations pré-entraînées pour obtenir une représentation finale du document ; 2) un modèle en ligne qui apprend conjointement les représentations de textes à plusieurs niveaux (mots, concepts, documents). Plus précisément, notre première approche vise à améliorer la représentation de document pré-entraînée par le modèle neuronal ParagraphVector (Le et al., 2007) en combinant cette représentation originale basée sur le texte avec une nouvelle représentation conceptuelle apprise par notre modèle *conceptualDoc2vec*. Nous combinons ces deux types de représentations en utilisant une optimisation pour construire un nouvel espace qui rapproche des espaces précédents. Nous obtenons une représentation dite optimale qui prend en compte à la fois la sémantique distributionnelle issue du corpus de documents et la sémantique relationnelle issue de la ressource externe. Nous proposons ensuite une deuxième approche qui apprend simultanément les représentations de documents, de mots et de concepts. Les représentations des documents sont apprises en maximisant la prédiction des vecteurs de mots et de concepts en fonction de leur contexte voisinage. En plus, pour mieux capturer la sémantique relationnelle dans les représentations de textes, nous proposons d'intégrer les contraintes de relations entre les mots/concepts dans ces deux méthodes d'apprentissage. Deux approches sont utilisées pour injecter ces contraintes relationnelles, une basée sur la régularisation de la fonction objec-

tif, une basée sur les instances dans le texte d'entraînement. Ces propositions sont évaluées comparativement sur les tâches de TALN et RI en utilisant des jeux de données génériques (ROBUST04) ainsi que spécifiques au domaine médical (OHSUMED, TREC MED).

2. *Exploitation d'un réseau de neurones pour mieux apparier les documents et les requêtes.* Nous modélisons dans la deuxième contribution un modèle neuronal pour l'appariement des paires de document-requête. À notre connaissance, il s'agit d'une des premières approches combinant la sémantique distributionnelle et relationnelle dans une architecture neuronale pour améliorer l'appariement de requête-document. Dans cette contribution, nous proposons une méthode pour construire une représentation symbolique de texte, qui s'appuie sur les concepts et leurs relations dans une ressource externe. Vu le grand nombre de relations objet-objet dans le texte, nous proposons une méthode *relation mapping* qui vise à projeter des paires dans un espace de groupes d'objets à faible dimension. Nous proposons ensuite un réseau neuronal siamois "de bout en bout" qui apprend une fonction d'appariement des documents en utilisant des vecteurs d'entrée combinant à la fois les représentations distribuées et les représentations basées sur les ressources externes. Le réseau est entraîné pour maximiser la distance entre la similarité des paires requête-document pertinentes et la similarité des paires non pertinentes. Une évaluation expérimentale est menée afin de valider notre approche. Pour cela, nous utilisons deux jeux de données TREC, à savoir TREC PubMed CDS et TREC GOV2 Terabyte et deux ressources sémantiques, respectivement MeSH et WordNet.

3 Organisation de la thèse

Cette thèse est constituée d'un chapitre d'introduction suivie de deux principales parties et d'un chapitre de conclusion. La première partie, composée de deux chapitres, présente la synthèse des travaux de l'état de l'art. La deuxième partie présente nos contributions dans le contexte de la RI sémantique. Nous présentons le détail de cette organisation ci-après.

— Le chapitre 1 introduit le contexte, les problématiques de recherche abordées et les contributions issues de nos travaux.
— Le chapitre 2, *Recherche d'information et ressources sémantiques*, présente les concepts de base ainsi que les types de ressources sémantiques (section 1). Nous introduisons en particulier dans la section 1.1 les notions de base d'une ressource sémantique. Ensuite, les différents types de ressources sémantiques sont présentés dans la section 1.2 en fonction de leur contenu.

Nous présentons aussi dans ce chapitre les travaux qui utilisent des ressources sémantiques pour la RI dans la section 2. Plus précisément, nous décrivons plusieurs modèles regroupés en trois niveaux, selon le niveau d'application de la ressource externe dans les étapes de RI : l'expansion de la requête (section 2.1), l'expansion du document (section 2.2) et l'appariement document-requête (section 2.3).

— Le chapitre 3, *Réseaux de neurones et Recherche d'Information*, donne un aperçu sur les réseaux de neurones (section 1) ainsi que leur application dans l'apprentissage de représentation de textes (section 2). Nous commençons ce chapitre par les concepts de base du réseau de neurones : le modèle d'un neurone (section 1.1), l'architecture des réseaux (section 1.2), les fonctions objectif d'apprentissage (section 1.3) et les algorithmes d'entraînement (section 1.4). Nous présentons par la suite les approches d'apprentissage de représentations distribuées de textes (section 2.1) ainsi que les approches qui augmentent ces représentations par des ressources externes (2.2). Enfin, la section 2.3 présente les applications de ces représentations distribuées en RI.

— Le chapitre 4, *Apprentissage des représentations du texte*, présente la première partie de nos contributions relative à des apprentissages de représentations du texte augmentées par des ressources externes. Nous proposons d'abord une approche de mise à jour des représentations distribuées des documents guidées par les concepts (section 3). Puis nous présentons une approche d'apprentissage conjoint des représentations de documents, de mots et de concepts (section 4). Nous détaillons ensuite l'intégration des contraintes relationnelles issues des ressources externes dans l'apprentissage des représentations distribuées de textes (section 5). Enfin, nous présentons les évaluations expérimentales menées pour valider les modèles proposés ainsi que des discussions des résultats obtenus (section 6).

— Le chapitre 5, *Modèle neuronal pour la RI*, présente la deuxième partie de nos contributions relative à un modèle neuronal d'appariement augmenté par une ressource sémantique. Nous détaillons l'approche utilisée pour construire la représentation symbolique du document dans la section 2.1. Puis, l'architecture ainsi que l'algorithme d'apprentissage du réseau sont présentés dans la section 2.2. La section 3 termine ce chapitre avec les résultats expérimentaux et les discussions.

En conclusion, nous faisons le bilan des travaux réalisés dans le cadre de la thèse, synthétisons des éléments originaux de nos contributions. Nous présentons ensuite les différentes pistes d'évolution de ces travaux.

Partie I

SYNTHÈSE DES TRAVAUX DE L'ÉTAT DE L'ART

2

RECHERCHE D'INFORMATION ET RESSOURCES SÉMANTIQUES

Introduction

La recherche d'information (RI), un des premiers domaines de recherche en informatique, a proposé des premières solutions automatiques pour le stockage et la recherche de texte (Luhn, 1957). Son objectif principal consiste à sélectionner des informations pertinentes (textes, sons, images, ou données multimédia, etc.) dans une collection de documents en réponse à un besoin en information formulé par un utilisateur sous la forme d'une requête. La recherche d'information concerne essentiellement la structuration, l'analyse, l'organisation, le stockage, la recherche et l'extraction de l'information (Salton, 1968). Ce processus se déroule au sein d'un système de recherche d'informations (SRI). Ce dernier est un logiciel permettant d'effectuer l'ensemble des tâches nécessaires à la RI. Un SRI possède trois fonctions fondamentales qui définissent le modèle de recherche : représenter le contenu des documents, représenter le besoin de l'utilisateur et comparer ces deux représentations.

Un système de recherche d'information (SRI) facilite l'accès à un corpus de documents pour mieux répondre au besoin en information de l'utilisateur. Pour réduire la complexité de l'appariement requête-document, ce système utilise l'index inversé comme le stockage des documents. Cette structure d'index inversé est optimisée pour garantir un accès rapide aux documents pertinents. Cette approche traditionnelle s'appuie fondamentalement sur le principe de "sac de mots" où l'appariement entre une requête et un document est basé sur un rapprochement lexical entre les mots qui les composent. L'une des limites de ces approches repose sur la difficulté de capter la sémantique des mots en raison de leur variation lexicale (e. g., acronymes, homonymes, synonymes, etc.) qui s'ajoute à l'ambiguïté du besoin en information caché derrière une requête. Cet écart significatif entre la représentation de la requête et des documents est un problème bien connu en RI sous le nom de "fossé sémantique" ("*semantic gap*" en anglais). Ce fossé est l'une des principales raisons du défaut d'appariement entre requête et document qui

conduit à la dégradation des performances d'un système de RI (Crestani, 2000). Cet enjeu a été abordé sous différents angles dont principalement la *recherche d'information sémantique* qui a pour objectif d'améliorer les représentations du document et de la requête en explicitant les associations entre les mots de la requête et du document au delà de l'appariement lexical. Pour pallier le problème du fossé sémantique, plusieurs travaux ont utilisé des ressources externes structurées qui constituent un référentiel de connaissances établies par des humains, comme les thesaurus, les ontologies, les terminologies, graphes de connaissances, etc. Ces ressources fournissent des informations sur les sens des mots et de leurs relations comme la synonymie, l'homonymie, l'hyponymie, etc.

Dans le contexte de cette thèse, nous nous intéressons à l'exploitation des ressources sémantiques dans les modèles d'indexation et d'appariement requête-document. Nous citons dans la section suivante les différents types de ressources qui sont les plus utilisées en RI sémantique (Section 1). Puis nous décrirons des travaux connexes qui utilisent des ressources sémantiques pour la RI (Section 2).

1 Ressources sémantiques

Pour faciliter l'accès à la quantité de données volumineuse, la conception et le développement des ressources sémantiques (e. g., ontologies, thésaurus, bases lexicales, etc.) est devenu un des champs de recherche les plus populaires en Informatique (Bast et al., 2016). En effet, les travaux sur les ressources sémantiques sont de plus en plus répandus dans les différentes communautés comme l'ingénierie des connaissances, le traitement automatique du langage naturel, le Web ou la bio-informatique. Les ressources exploitées pour la RI sémantique peuvent être *structurées* ou *non-structurées*. Quelques exemples des ressources non-structurées sont des corpus de documents sans l'annotation du sens (e. g., *Brown* Corpus (Francis, 1971), *British National* Corpus (Burnard, 1995)) et avec l'annotation sur le sens de mots (e. g., *SemCor* Corpus (Miller et al., 1993), *interest* Corpus (Bruce and Wiebe, 1994)). Dans notre travail, nous nous intéressons dans un premier temps aux ressources structurées. Ces ressources peuvent varier d'une liste de termes techniques d'un domaine (une *terminologie*) à un ensemble structuré des termes et de concepts représentant la connaissance d'un champ d'informations (une *ontologie*). Nous présentons dans ce qui suit un aperçu sur les notions de base et la typologie des ressources sémantiques (Zargayouna et al., 2015).

1.1 *Notions de base*

• **Terme**. Un terme est constitué d'un mot ou d'un groupe de mots qui s'applique à un seul objet ou une idée dans un domaine donné. Un terme constitué d'un seul mot (e. g., "animal", "ordinateur") est dit terme simple ou uni-terme, alors que celui formé de plusieurs mots est appelé terme complexe ou multi-termes (e. g., "caisse d'épargne", "chemin de fer").

• **Concept**. Un concept, défini comme un élément de la pensée, représente une idée abstraite. Il est la construction mentale qui représente la signification du terme et qui fait référence à l'objet. Un *concept terminologique* représente la signification normalisée des termes par le biais d'une définition en langue naturelle. Dans un contexte particulier, un concept est exprimé par un terme (simple ou complexe). Un seul terme, intitulé parfois "label préféré", est choisi comme étiquette du concept terminologique selon la définition des linguistes. Chaque concept a un seul terme préféré, qui est souvent le nom du concept. Par exemple, le terme "Céphalée" est *préféré* pour le concept "Céphalée", et les autres termes *non-préférés* sont "Céphalodynie", "Douleur crânienne", "Mal de tête", etc.

• **Classe**. Dans un langage informatique (e. g., langage objet de représentation des connaissances), un concept peut être représenté par une *classe* au lieu d'un *concept terminologique*. La classe est une représentation partielle et orientée du concept. Un concept est représenté par un concept terminologique et/ou une classe. Le concept terminologique est associé à une définition en langue naturelle et des termes synonymes, alors que la classe est associée à l'ensemble des propriétés et des relations qui caractérisent le concept (Charlet et al., 2004).
Une classe peut se définir par un ensemble de caractéristiques aussi appelées *propriétés* ou *attributs*. Les propriétés peuvent avoir des valeurs, qui varient suivant la classe à laquelle on fait référence.
Une classe peut également être définie par l'ensemble de ses *instances*, autrement dit l'ensemble des *objets* qui sont caractérisés par cette classe. Par exemple, le "canal du Midi" est une instance de la classe *Canal*.

• **Relation**. Il existe plusieurs types de relations. La *relation lexicale* connecte deux termes par un lien de type synonymie, antonymie, hyponymie, hyperonymie, ou méronymie. Par exemple, les termes "douleur crânienne" et "mal de tête", qui réfèrent au même concept ("Céphalée"), sont dits synonymes.
Les classes sont aussi reliées entre elles par des relations. La *relation des classes* la plus populaire est la hiérarchie de spécialisation ou de généralisation (e. g., la classe "Animal" est la classe mère de "Chat"). D'autres relations entre classes peuvent être définies en fonction des usages. Par exemple, les classes représentant

des objets spatiaux comme "Chemin" sont associées entre elles par une relation d'inclusion spatiale dite de "localisation".

1.2 *Typologie des ressources*

Nous présentons à ce niveau les différents types de ressources sémantiques en fonction de leur contenu.

• **Terminologie**. Une terminologie, dans le sens d'une ressource sémantique, est un ensemble de termes, rigoureusement définis, qui sont spécifiques à une science, une technique, un domaine particulier de l'activité humaine (Larousse, 2016). Le but essentiel d'une terminologie est de faciliter la gestion et le partage de masses d'informations, en réduisant l'ambiguïté entre les termes d'un domaine. Dans une terminologie, des termes du domaine sont normalisés par la notion de *concept*. Un terme est la combinaison indissociable 1) d'une expression linguistique représentant un mot métier et 2) d'un concept qui représente sa signification. Comme une norme pour un domaine donné est déterminée dans une terminologie de référence, la signification de chaque terme est fixé. Ainsi, il y a une seule interprétation possible pour l'utilisateur. Il existe différentes terminologies alignées aux différents objectifs de traitement de l'information : *nomenclature*, *thésaurus*, *base lexicale* et *ontologie*.

• **Nomenclature**. Une nomenclature est un ensemble des *termes* en usage dans une science, un art, ou relatifs à un sujet donné, présentés selon une classification méthodique (Larousse, 2016). Elle représente une instance de classification (code, tableau, liste, règles d'attribution d'identité...) faisant autorité et servant de référence à une discipline donnée. Il n'y a pas d'arrangement particulier des termes ni de définition explicite ; l'objectif visé est l'exhaustivité. Les *concepts* d'un domaine sont décrits dans une nomenclature de manière complète sans se restreindre à un objectif spécifique. Un exemple d'une nomenclature importante dans le domaine médical est la Nomenclature Systématique des Médecines Humaine et Vétérinaire (*Systematized Nomenclature of Medicine - SNOMED*). La SNOMED est une nomenclature pluri-axiale couvrant tous les champs de la médecine et de la dentisterie humaines, ainsi que de la médecine vétérinaire. Dans chaque axe, les concepts sont représentés par une série de termes au sein de laquelle on peut distinguer une formulation préférée et des synonymes de diverses natures syntaxiques. La Figure 2.1 (extrait du *SNOMED CT Browser*[1]) illustre l'information du concept *"Solar degeneration"* avec ses termes synonymes comme *"Farmer's skin"* ou *"Sun damaged skin"*.

1. https://uts.nlm.nih.gov/snomedctBrowser.html

Concept: [43982006] Solar degeneration

Descriptions (14)

Id	Description	Type	Status
781057010	Solar degeneration (disorder)	Fully specified name	Active
73334019	Solar degeneration	Synonym	Active
493740015	Actinic ageing	Synonym	Active
493741016	Actinic aging	Synonym	Active
73335018	Actinic degeneration	Synonym	Active
3029490012	Actinic degeneration of skin	Synonym	Active
73336017	Actinic elastosis	Synonym	Active
2950846011	Dermatoheliosis	Synonym	Active
73338016	Farmer's skin	Synonym	Active
493739017	Photoageing	Synonym	Active
493742011	Photoaging	Synonym	Active
73337014	Sailor's skin	Synonym	Active
73339012	Solar elastosis	Synonym	Active
2920732015	Sun damaged skin	Synonym	Active

Figure 2.1 – Exemple du concept *"Solar degeneration"* dans SNOMED.

• **Thésaurus**. Un thésaurus forme un répertoire alphabétique de termes normalisés pour l'analyse de contenu, le classement et donc l'indexation de documents d'information. Il aide à la normalisation des mots-clés utilisés dans un système de RI. Ainsi la liste de mots-clés qui représentent le contenu documentaire est construite par des experts (linguistes, documentalistes) à partir des descripteurs dans un thésaurus.
Un thésaurus est constitué d'une structuration hiérarchisée de *termes* désignant les *concepts*. Les termes y sont organisés de manière conceptuelle et reliés entre eux par des *relations lexicales*. Il normalise son vocabulaire pour être cohérent : chaque *concept terminologique* a un seul *terme descripteur* et plusieurs *termes non descripteurs*, et un *terme descripteur* ne doit être associé qu'à un seul concept. Pour un *concept*, les *termes non descripteurs* renvoient aux descripteurs par une relation d'équivalence. On peut trouver dans un thésaurus des informations sur des termes descripteurs (la définition) et ses relations ("synonyme de", "relié à") à d'autres termes. Un thésaurus peut fournir aussi les informations comme "des termes plus spécifiques", "des termes plus larges", ou des "termes connexes". A titre d'exemple, le thésau-

rus biomédical MeSH (*Medical Subject Heading* [2]) est utilisé pour indexer, classer et rechercher des documents de la base MEDLINE (PubMed) [3]. La Figure 2.2 illustre un exemple sur les informations du concept *"Headache"* telles que *"MeSH Heading"* (terme descripteur), *"Entry Terms"* (termes non descripteur ou termes préférés), *"Tree Number"* (code du concept dans la hiérarchie). La Figure 2.3 présente un extrait de la hiérarchie arborescente dans MeSH.

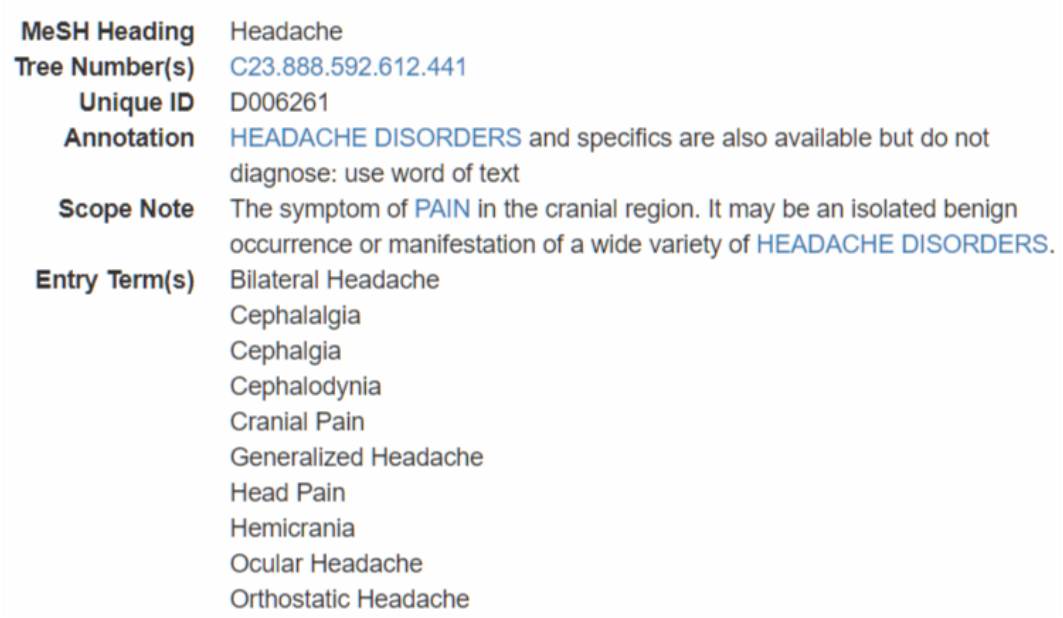


Figure 2.2 – Exemple de l'information sur le concept *"Headache"* dans MeSH.

• **Base lexicale**. L'organisation d'une base lexicale ressemble à celle du thésaurus. Les *termes* sont regroupés par un *concept terminologique* qui est connecté aux autres concepts par des *relations lexicales*. L'objectif d'une base lexicale est de distinguer tous les sens possibles qu'un terme peut prendre dans un texte et non de sélectionner le sens le plus commun dans un domaine d'étude. WordNet (Miller, 1995) est une base lexicale (initialement en anglais) très utilisée en TALN et RI du fait de sa couverture quasi-totale de la langue anglaise. WordNet couvre la majorité des noms, verbes, adjectifs et adverbes structurés en un réseau de nœuds et de liens (Fellbaum, 1998). Les noms, verbes, adjectifs et adverbes sont regroupés par synonyme, appelés *synsets* qui expriment un concept terminologique distinct. Chaque *synset* représente un sens unique d'un mot particulier. La relation de base entre les termes d'un même *synset* est la synonymie. Les *synsets* sont reliés entre eux par des relations lexicales : l'hyperonymie/hyponymie, la méronymie, l'antonymie, etc.

2. https://www.nlm.nih.gov/mesh/
3. https://www.ncbi.nlm.nih.gov/pubmed/

Figure 2.3 – Extrait de la hiérarchie MeSH pour le concept *"Headache"*.

• **Ontologie**. En philosophie, l'ontologie est une étude de l'être en tant qu'être, indépendamment de ses déterminations particulières. En informatique, ce terme est repris dans les années 90, par la première définition : "une ontologie est une spécification d'une conceptualisation" (Gruber, 1993). Ainsi, une ontologie est une description des concepts et des relations qui existent pour un objet ou un ensemble d'objets. L'objectif principal d'une ontologie, comme celui d'une terminologie en général, est de partager et de réutiliser des connaissances d'un domaine donné. Dans une ontologie, un *concept* est représenté par une *classe* avec des *attributs* (ou *propriétés*). Une classe est peuplée par des *instances*, qui sont des objets, entités ou événements réels. Ces instances sont reliées par des *relations* entre eux.
Le développement du Web de données (*Linked Open Data*) facilite la création et la valorisation des ontologies. La ressource DBpedia (Auer et al., 2007) est une base de connaissance récemment utilisée dans les communauté Web Sémantique, Ingénierie de Connaissance ainsi que Recherche d'Information. Elle contient une ontologie interdomaine qui a été créée manuellement à partir des infoboxes les plus utilisées dans Wikipedia. L'ontologie couvre actuellement 685 classes qui forment une hiérarchie de subsomption et sont décrites par 2 795 propriétés différentes. L'ontologie de DBpedia contient actuellement environ 4 233 000 instances. La Figure 2.4 illustre une partie de la hiérarchie des classes dans l'ontologie de DBpedia.

Un exemple d'une ontologie importante dans le domaine médical est l'ontologie de gènes (GO - *Gene Ontology*) qui est une ressource terminologique destinée à structurer la description des gènes et des produits géniques dans le cadre d'une ontologie commune à toutes les espèces. La base GO est conçue comme un graphe

Figure 2.4 – Extrait de la hiérarchie des classes dans DBpedia.

orienté acyclique, chaque terme étant en relation avec un ou plusieurs termes du même domaine, et parfois d'autres domaines.

2 Méthodes de RI basées sur l'utilisation des ressources externes

Les bases lexicales (e.g., WordNet), les thésaurus (e.g., MeSH), les ontologies (e.g., DBpedia) représentent des ressources externes qui fournissent des informations pertinentes sur la sémantique des mots modélisée à travers des objets (e.g. des termes, des entités ou des concepts) et leurs relations associées. Les modèles de RI basés sur ces ressources externes se distinguent des modèles classiques par la prise en compte de la ressource sémantique pour le choix des index ainsi que l'appariement requête-document. Nous présentons dans ce qui suit trois catégories de travaux selon le niveau d'application des ressources sémantiques : la représentation des requêtes, la représentation des documents et l'appariement requête-document.

2.1 *Représentation des requêtes*

La manipulation au niveau de la représentation des requêtes présentée dans cette partie consiste en l'expansion des requêtes initiée par Rocchio (1971). En particulier, l'expansion des requêtes améliore la représentation de la requête initiale d'un utilisateur en y ajoutant des termes utiles, dans le but d'augmenter le rappel. Généralement, les termes d'expansion sont automatiquement sélectionnés à partir d'un ensemble de documents renvoyés en premier temps (Lavrenko and Croft, 2017; Rocchio, 1971; Zhai and Lafferty, 2001), qui sont supposés être pertinents. Cette technique s'appelle *Pseudo-Relevance Feedback* (PRF). Dans notre contexte de travail, nous nous intéressons aux travaux qui visent à reconstruire les requêtes en utilisant les termes et leur relations recensés dans les ressources sémantiques (Wang and Akella, 2015; Amini and Usunier, 2007; Stokes et al., 2009; Fu et al., 2005; Pal et al., 2014; Xiong and Callan, 2015b).

Voorhees (1994) est l'un des premiers auteurs à avoir proposé une approche d'expansion de requêtes avec les concepts et les relations de WordNet. Les synsets de WordNet sont utilisés pour représenter les concepts étendus. Les termes de la requête sont annotés manuellement par le sens approprié. Puis, les termes d'expansion sont rajoutés automatiquement selon les relations dans WordNet comme les synonymes ou les hyponymes. La requête est représentée par un modèle vectoriel étendu. Ce vecteur est constitué de trois sous-vecteurs de différents types de concepts (*ctypes*) : (1) les radicaux (*stems*) des mots simples qui n'existent pas dans WordNet ; (2) les synsets des noms désambiguïsés ; (3) les radicaux des mots désambiguïsés. La similarité entre le document d et la requête q est estimée par la somme des similarités de chaque sous-vecteur $\vec{q_i}$ et le vecteur du document $\vec{d}$:

$$sim(d, q) = \sum_{ctype_i} \alpha_i \vec{q_i} \cdot \vec{d} \tag{2.1}$$

où α_i est une pondération qui correspond à l'importance du type de concepts $ctype_i$. L'expérimentation du modèle sur des collections TREC n'a pas donné des améliorations significatives surtout quand la requête est longue. Cependant, pour les requêtes courtes, l'auteur a constaté que son approche peut apporter des améliorations avec l'expansion automatique.

Avec la même application de WordNet, Navigli and Velardi (2003) utilisent l'information sur les sens de mots (*Synsets*) pour l'expansion des requêtes. Ils appliquent cinq méthodes d'expansion :

1. Expansion par synset : les termes de la requête sont remplacés par leur sens (synsets)
2. Expansion par hyperonyme : les termes de la requête sont augmentés par leur hyperonyme direct

3. Expansion par définition de synset : les termes de la requête sont augmentés par les synsets de leur définition (*gloss* dans WordNet)
4. Expansion par définition de terme : les termes de la requête sont augmentés par les termes dans leur définition
5. Expansion par nœuds en commun : les termes de la requête sont augmentés par les termes qui ont les mêmes synsets (les termes synonymes)

L'expérimentation est menée en utilisant la collection de TREC Web 2001, WordNet et le moteur de recherche Google. Les auteurs affirment que l'expansion avec des synonymes et des hyperonymes a un effet limité sur la performance de recherche d'information sur le Web. Ils suggèrent que d'autres types d'informations sémantiques dérivables d'une ontologie sont plus efficaces, par exemple des mots de la définition et des nœuds communs. En effet, ils trouvent que les mots dans le même domaine sémantique et le même niveau de généralité sont les meilleurs candidats à l'expansion. La ressource sémantique est utilisée pour extraire la sémantique d'un mot, puis la requête est étendue en utilisant des mots co-occurrents. L'efficacité de l'utilisation des ressources pour améliorer la performance dépend du type de tâche (i. e., recherche par sujet et recherche par site) et de la longueur de la requête. Avec la même remarque que Voorhees (1994), Navigli and Velardi (2003) concluent que l'expansion des requêtes convient aux requêtes courtes.

Baziz et al. (2003) exploitent les relations de synonymie et de hyperonymie de WordNet. Les auteurs limitent l'expansion à un ensemble de concepts (mono ou multi-termes) accessibles par des liens ontologiques à partir d'un concept de requête. Le processus d'expansion se déroule en trois étapes :

1. Lemmatisation et étiquetage des mots de la requête : lors de l'identification des concepts, les groupes nominaux de la requête sont projetés sur WordNet et les plus longs (couvrant le maximum de mots non vides de la requête) sont conservés.
2. Exploitation des relations sémantiques : détection de termes ou multi-termes liés à ceux de la requête par la synonymie, l'hyperonymie (généralisation et spécialisation).
3. Pseudo-désambiguïsation par superposition : lorsque plusieurs concepts (synsets) ont la même valeur de similitude avec la requête, les concepts ayant le plus grand nombre de mots différents sont retenus.

Cette approche d'expansion est expérimentée sur la collection de CLEF 2001 en utilisant le moteur de recherche *Mercure* (Boughanem, 1992). Les auteurs ont montré une amélioration significative en termes de précision par rapport au système de recherche sans expansion.

Fu et al. (2005) présentent des techniques d'expansion des requêtes spatiales basées à la fois sur une ontologie du domaine et sur une ontologie géographique

(geo-ontologie). L'ontologie du domaine modélise les terminologies d'un domaine d'application et est utilisée pour résoudre l'aspect "*quoi*" d'une requête. L'aspect "où" de la requête est traité avec la géo-ontologie, qui est construite pour fournir une structure de connaissance de l'espace géographique intéressé. Contrairement aux techniques d'extension de requêtes basées sur des termes, les techniques proposées par Fu et al. (2005) permettent d'étendre une requête en essayant de déduire son *territoire*, et cela est spécialement conçu pour résoudre une requête spatiale. Le *territoire* d'une requête concerne l'espace de recherche spatiale d'une requête. Différentes sources d'évidence telles que les types de termes spatiaux encodés dans la géo-ontologie, les types de termes non-spatiaux encodés dans l'ontologie de domaine, la sémantique des relations spatiales, leur contexte d'utilisation et la satisfaction du résultat de la recherche initiale sont pris en compte pour effectuer l'expansion d'une requête spatiale. Les expérimentations ont montré que cette méthode permet d'améliorer la performance de la recherche.

Pal et al. (2014) proposent une technique d'expansion de requête utilisant des mots extraits de plusieurs sources d'information. Ils choisissent les termes d'expansion candidats à partir d'un ensemble de documents pseudo pertinents ; cependant, le profit de ces termes est mesuré en fonction de leurs définitions fournies dans une ressource lexicale manuscrite comme WordNet. Pour chaque mot d'une requête, les termes d'expansion candidats sont choisis dans les premiers documents renvoyés pour cette requête. Plusieurs méthodes d'expansion des requêtes ont prouvé que les documents pseudo pertinents sont les bonnes sources de termes d'expansion candidats (Carpineto et al., 2001; Xu and Croft, 2000; Amati and Van Rijsbergen, 2002). L'importance de la similarité entre un terme candidat t_c et un terme t_i de la requête est calculé sur le nombre de mots en commun dans la définition, qui se trouve dans WordNet, de ces termes. Le *score de relation* entre les termes $Rel(t_c, t_i)$ est calculé sur ce nombre de mots en commun en utilisant l'indice de Jaccard ou l'indice de Sørensen-Dice. Le score final d'expansion de chaque terme candidat t_c est combiné avec son *score de relation* $Rel(t_c, t_i)$, son idf_{t_c} et le *score de similarité* $sim(d, q)$ entre la requête et les documents pseudo pertinents contenant ce terme.

$$score(t_c, t_i) = Rel(t_c, t_i) * idf_{t_c} * \sum_{d \in PRD} \frac{\text{sim}(d, q)}{\max_{d' \in PRD_q} \text{sim}(d', q)} \tag{2.2}$$

où $\text{sim}(d, q)$ dénote le score de similarité entre le document d et la requête q ; PRD_q est l'ensemble de documents pseudo pertinents de la requête q
Les auteurs ont expérimenté les combinaisons de leur méthode avec celles proposées par Pal et al. (2013). Les résultats montrent que la combinaison de diverses méthodes semble bien fonctionner et donne des résultats qui sont meilleurs que les méthodes individuelles impliquées dans la combinaison.

Xiong and Callan (2015b) proposent deux algorithmes basés sur la catégorisation de mots dans FreeBase comme les ressources sémantiques externes pour entraîner les représentations des catégories. Ils examinent deux approches pour effectuer l'expansion de la requête, une non-supervisée et une supervisée.
Ils effectuent l'expansion non-supervisée de la requête avec les ressources sémantiques externes pour entraîner les représentations des catégories en deux étapes : annotation sémantique et sélection de terme. Ils implémentent deux approches pour l'annotation sémantique, soit récupérer les noms d'entité de FreeBase comme les ressources sémantiques externes pour entraîner les représentations des catégories directement via Google Search API, soit filtrer dans l'annotation FACC1 [4] avec un calcul de score pour les entités. Une fois que les textes sont annotés, les auteurs développent ensuite deux méthodes pour sélectionner les termes d'expansion depuis les entités identifiées : (1) sélection par PRF et (2) sélection par Catégorie. Etant donné l'ensemble entités E dont chaque entitié $e_k \in E$ a un score $r(e_k)$, la première méthode calcule un PRF appliqué sur la description des entités. Le score d'un terme candidat t_c est calculé comme suit :

$$score(t_c) = \sum_{e_k \in E} \frac{tf(desc(e_k), t_c)}{|desc(e_k)|} * r(e_k) * \log \frac{|E_R|}{df(t_c)} \tag{2.3}$$

où $tf(desc(e_k), t_c)$ est l'occurrence du terme candidat t_c dans la description $desc(e_k)$ de l'entité e_k, $desc(e_k)$ est la longueur de la description, $df(t_c)$ est la fréquence de document du terme t_c dans l'ensemble du corpus de description de Freebase, et $|E_R|$ est le nombre total des entités qui ont des descriptions dans la ressource Freebase.
La deuxième méthode estime la distribution des termes et de la requête sur les catégories de FreeBase et sélectionne les termes qui ont des distributions de catégories similaires avec la requête. D'abord, les auteurs définissent la probabilité $P(t|c_u)$ qu'un terme t soit généré par une catégorie d'entités c_u comme suit :

$$P(t|c_u) = \frac{\sum_{e_k \in c_u} tf(desc(e_k), t)}{\sum_{e_k \in c_u} |desc(e_k)|} \tag{2.4}$$

En utilisant les règles de Bayes, la probabilité que le terme t_c appartienne à la catégorie c_u est calculée par :

$$P(c_u|t_c) = \frac{P(t_c|c_u)}{\sum_{c_v \in C} P(t_c|c_v)} \tag{2.5}$$

4. FACC1 Annotation on ClueWeb09.
http://lemurproject.org/clueweb09/FACC1/

De façon similaire, la distribution des catégories c_u sur une requête q est calculée comme suit :

$$P(c_u|q) = \frac{P(q|c_u)}{\sum_{c_v \in C} P(q|c_v)} \tag{2.6}$$

Le score d'un terme candidat t_c est la similarité, estimée par la divergence négative de Jensen-Shannon, entre deux distributions $P(c_u|t_c)$ et $P(c_u|q)$:

$$score(t_c) = -\frac{1}{2}\mathrm{KL}(P(c_u|q)||P(c_u|q,t_c)) - \frac{1}{2}\mathrm{KL}(P(c_u|t_c)||P(c_u|q,t_c)) \tag{2.7}$$

où $P(c_u|q,t_c)) = \frac{1}{2}(P(c_u|q) + P(c_u|t_c))$
En croisant ces approches d'annotation sémantique et de sélection de termes, ils ont finalement proposé quatre méthodes d'expansion non-supervisées. Enfin, pour une meilleure expansion, un modèle supervisé est formé pour combiner l'information provenant de différents algorithmes d'annotation sémantique (récupération directe, filtrage dans l'annotation) et de sélection de termes (tf.idf, catégorisation). Les expérimentations sur le jeu de données ClueWeb09 et les requêtes TREC Web démontrent que leurs méthodes sont presque 30% plus efficaces par rapport aux méthodes d'expansion de référence (Metzler and Croft, 2005). Selon les auteurs, ce travail est le premier à montrer l'efficacité de Freebase pour l'expansion des requêtes sur le corpus ClueWeb09.

2.2 *Représentation des documents*

L'application des ressources sémantiques au niveau de la représentation des documents peut se traduire par deux approches principales : l'expansion conceptuelle des documents (Chalendar et al., 2002; Gobeill et al., 2008; Agirre et al., 2010) et l'indexation conceptuelle des documents (Baziz et al., 2005; Chahine et al., 2011; Gupta et al., 2017). Comme l'expansion des requêtes, les techniques d'expansion/indexation conceptuelle de documents ont été abordées dans la littérature de la RI pour réduire le fossé sémantique entre les documents et les requêtes. Tandis que l'expansion des requêtes vise à augmenter le rappel en ajoutant les termes utiles qui sont absents dans la requête, l'expansion et l'indexation conceptuelles des documents aident à améliorer la précision par l'indexation avec les concepts au lieu des termes ambigus.

Baziz et al. (2005) déclarent que la recherche d'information basée sur la ressource sémantique est encourageante pour améliorer la qualité des résultats puisque la sémantique des documents est capturée. Dans leur modèle, le contenu du document est représenté à l'aide d'un réseau sémantique optimal. Plus précisément,

les termes du document sont d'abord projetés sur la base lexicale WordNet afin d'identifier les sens (ou concepts) correspondants. Un terme est dit ambigu quand il correspond à plus d'un sens dans WordNet. Les auteurs proposent une approche de désambiguïsation basée sur le principe que, parmi les différents sens possibles (concepts candidats), le sens le plus approprié pour un terme donné est celui qui est connecté avec la plupart des autres sens du même document. Leur approche consiste à affecter un poids à chaque concept candidat en sommant les valeurs de similarité entre celui-ci et les autres concepts candidats (correspondant aux différents sens des autres termes du document). Les concepts sont pondérés par la fréquence conceptuelle $CF * IDF$, où la fréquence de concept CF (dans un document) est définie comme suit :

$$CF(c) = Count(c) + \sum_{sc(c)} \frac{length(sc)}{length(c)} Count(sc) \tag{2.8}$$

où $Count(c)$ est le nombre d'occurrences du concept candidat c dans le document, $length(c)$ est la taille du concept c en mots, $sc(c)$ fournit le nombre des sous-concepts de c correspondant à des entrées du réseau et sc est un sous-concept. Le concept ayant le poids le plus élevé est retenu comme sens approprié du terme associé. Finalement, le document est représenté comme un réseau de concepts (sens)

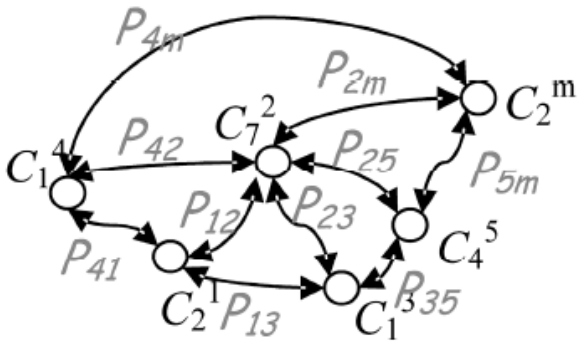


Figure 2.5 – Exemple de réseau sémantique construit à partir des concepts d'un document (Baziz et al., 2005).

et de liens entre concepts (cf. la Figure 2.5). Les liens (arcs) entre les différents concepts (nœuds) sont pondérés par plusieurs mesures de similarité sémantique (Leacock and Chodorow, 1998; Lesk, 1986; Lin et al., 1998; Resnik, 1999). Les auteurs ont évalué leur approche sur deux collections : MuchMore [5] et GIRT de CLEF 2014. Ils ont constaté que l'indexation conceptuelle n'est efficiente que quand elle est combinée avec l'indexation classique. Cette dernière compense le silence de

5. http://muchmore.dfki.de

l'indexation conceptuelle et permet une plus large couverture du contenu des documents. Ils ont remarqué aussi l'impact positif de l'utilisation des mesures de similarité telles que la mesure de Resnik (1995) sur la précision des résultats.

Chahine et al. (2011) utilisent le réseau de catégories de Wikipedia comme taxonomie conceptuelle pour identifier les concepts importants et sujets principaux des documents. Un graphe orienté acyclique (GOA) est construit pour chaque document en reliant des termes (un ou plusieurs mots) à un concept dans le réseau de catégories Wikipedia. Plus précisément, les termes du document sont d'abord assignés aux concepts candidats (i. e., titres de Wikipédia). Un terme peut être assigné à plusieurs concepts qui correspondent aux différents sens de ce terme. Ensuite, l'ensemble des concepts "englobés" de chaque concept candidat est calculé à partir du réseau de catégories, via des relations génériques-spécifiques telles que l'hyperonymie ou l'holonymie, jusqu'à ce que la racine du réseau soit atteinte. Le GOA représentant du document est construit en fusionnant le graphe de chaque terme. Un exemple de GOA pour le texte *"Our students use C++, Java and Python on Linux"*

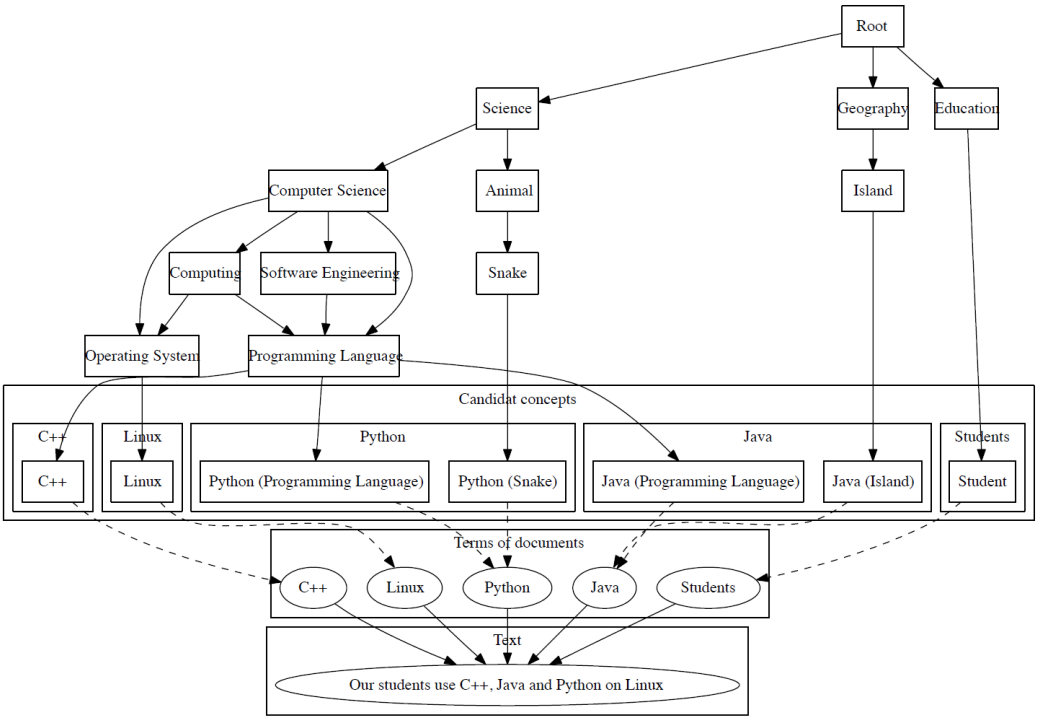


Figure 2.6 – GOA du texte *"Our students use C++, Java and Python on Linux"* (Chahine et al., 2011) ;

est illustré dans la Figure 2.6. Une fois que le graphe est construit, ses propriétés sont utilisées pour pondérer ses concepts et pour mettre en évidence les concepts

importants. Selon ces concepts importants, choisis par des sujets principaux, des mots-clés sont identifiés et la désambiguïsation des termes du document est faite. Leur méthode a été évaluée par les documentalistes sur un corpus de documents pédagogiques français. L'évaluation apporte des résultats très encourageants pour l'extraction des sujets, bien qu'un grand effort soit encore nécessaire pour améliorer la désambiguïsation.

Agirre et al. (2010) proposent une technique d'expansion de documents basée sur l'utilisation d'un algorithme de marche aléatoire identifiant à partir de WordNet les concepts et les termes les plus connexes. Les auteurs représentent WordNet par un graphe comme comme suit : les nœuds du graphe représentent les concepts WordNet (synsets) et les mots du dictionnaire ; les relations entre les synsets sont représentées par des arêtes non orientées ; et les mots du dictionnaire sont liés aux synsets associés par des arêtes orientées. Etant donné un document, les auteurs assignent une distribution uniforme des probabilités aux nœuds du graphe WordNet par rapport aux mots utilisés dans le document. Pour choisir les concepts d'expansion candidats, ils calculent ensuite le PageRank personnalisé (Haveliwala, 2002) sur le graphe en produisant une distribution de probabilité sur les concepts WordNet. Plus la probabilité d'un concept est élevée, plus il est lié au document donné. Les résultats montrent des améliorations significatives, avec quelques configurations optimisées, par rapport à la méthode classique BM25.

Dans le travail de Corcoglioniti et al. (2016a), les auteurs étudient les effets du contenu sémantique extrait automatiquement du texte pour la RI. Les requêtes et les documents sont traités pour extraire le contenu sémantique concernant quatre couches sémantiques (entités, types, cadres sémantiques, informations temporelles). Les auteurs enrichissent l'indexation par termes textuels avec des termes supplémentaires provenant de ces couches sémantiques. L'annotation sémantique est effectuée par PIKES (Corcoglioniti et al., 2016b), un système d'extraction des connaissances qui combine divers outils TALN pour distiller les connaissances à partir du texte, en l'alignant sur les ressources telles que DBpedia et FrameBase. Pour chaque terme dans le document, son contenu sémantique (extrait en quatre niveaux) est intégré dans la pondération d'un modèle vectoriel VSM pour mieux capturer l'information dans le texte. En évaluant leur modèle sur la collection créée par Waitelonis et al. (2015), les auteurs ont montré qu'en complétant l'information textuelle avec le contenu sémantique, ils peuvent surpasser les performances de RI par rapport aux modèles de référence.

Plusieurs travaux ont combiné l'expansion des requêtes et des documents pour améliorer la performance de la RI, spécialement dans le domaine biomédical. Par exemple, Le et al. (2007) exploitent les concepts et leurs relations sémantiques dans le méta-thésaurus UMLS pour évaluer l'impact de l'expansion des documents et/ou des requêtes. Les concepts sont identifiés par l'outil MetaMap (Aron-

son, 2001) à partir du texte (le document ou la requête). Les concepts liés à des concepts dans le texte par la relation directe *IS-A* sont choisis pour étendre le document (et/ou la requête). L'évaluation est effectuée sur la collection ImageCLEF 2005 avec 3 configurations : l'expansion des requêtes seule, l'expansion des documents seule, et l'expansion des requêtes et des documents. Les résultats ont montré une amélioration significative en termes de MAP avec la configuration basée sur l'expansion des requêtes et des documents par rapport à modèle de RI basée sur les mots simples.

Les auteurs de Dinh and Tamine (2012) et Dinh et al. (2013) utilisent aussi l'expansion des documents combinée avec l'expansion des requêtes pour avoir une meilleure efficacité en RI. Ils proposent une approche d'extraction de concepts multi-terminologies pour sélectionner les meilleurs termes candidats à partir d'un texte au moyen de techniques de vote (CombMAX, CombMin, CombSUM, etc.). Leur méthode est évaluée sur quatre terminologies (MeSH, SNOMED, ICD-10 et GO). Les auteurs s'intéressent particulièrement à l'effet de l'intégration des terminologies dans un processus de RI biomédical et sur l'utilité des techniques de vote pour combiner les concepts extraits de chaque document afin de fournir une liste de concepts uniques. Les résultats expérimentaux menés sur les collections de TREC Genomics montrent que leur approche de RI multiterminologique basée sur les techniques de vote est significativement performante par rapport aux modèles de référence (e. g., amélioration de +6,98% en termes de MAP). De plus, les résultats expérimentaux montrent que l'expansion de documents en utilisant des termes préférés en combinaison avec l'expansion de requêtes en utilisant des termes provenant de documents étendus améliore l'efficacité de la RI biomédicale.

2.3 *Appariement requête-document*

Tandis que les travaux dans deux approches précédentes utilisent les connaissances issues des ressources pour améliorer les représentations des documents et des requêtes, un autre groupe de travaux vise à améliorer la fonction d'appariement avec la ressource externe (Cao et al., 2011; L'Hadj et al., 2016; Balaneshinkordan and Kotov, 2016; Xiong and Callan, 2015a; Resnik, 1995; Goodwin and Harabagiu, 2016; Soldaini and Goharian, 2017; Koopman et al., 2016).
Par exemple, Koopman et al. (2016) présente un modèle d'inférence qui intègre des ressources de connaissances structurées, des méthodes d'appariement par statistiques et des inférences dans un cadre unifié. Les éléments clés du modèle sont une représentation de graphe du corpus et un appariement guidé par un mécanisme d'inférence réalisé sous la forme d'une traversée du graphe. Les auteurs divisent le problème du fossé sémantique en différentes questions fondamentales. Pour chaque question, ils décrivent ensuite un type d'inférence nécessaire pour y

remédier. Ils ajustent le mécanisme d'inférence en fonction des problèmes de fossé sémantique (e. g., discordance lexicale, discordance de granularité, discordance conceptuelle). Les analyses ont montré que leur mécanisme d'inférence améliore le rappel en retrouvant de nouveaux documents pertinents qui n'existaient pas dans les approches basées sur des mots clés. De plus, elle augmente aussi la précision par un réordonnancement efficace des documents. Lorsque l'inférence est utilisée, on peut généralement obtenir de meilleures performances pour les requêtes difficiles. Toutefois, l'inférence ne devrait pas être appliquée universellement : pour des questions faciles, sans ambiguïté et avec peu de documents pertinents, l'inférence dégrade l'efficacité.

Les auteurs dans Xiong and Callan (2015a) proposent le modèle d'apprentissage d'ordonnancement EsdRank qui prend en entrée les caractéristiques décrivant les liens entre les requêtes et les objets et entre les objets et les documents. Les auteurs utilisent les entités dans des ressources externes semi-structurées pour modéliser les caractéristiques de relations entre les documents et les requêtes. Plus précisément, le modèle sélectionne d'abord les entités de la ressource sémantique appartenant à la fois au document et à la requête. Ensuite, une méthode d'apprentissage d'ordonnancement est appliquée en utilisant une couche latente supplémentaire dans le processus de génération d'ordonnancements qui est construit sur les caractéristiques d'objets liés au document et à la requête. EsdRank est testé dans deux scénarios : utilisation d'une base de connaissances pour la recherche sur le Web et d'un vocabulaire contrôlé pour la recherche médicale. Les expériences sur les données TREC Web Track et OHSUMED montrent des améliorations significatives par rapport au modèle de référence ListMLE (Xia et al., 2008).

Résumé

Ce chapitre a présenté, dans un premier temps, les notions de base et les types de ressources sémantiques. Dans un second temps, nous avons axé notre état-de-l'art sur la RI sémantique basée sur les ressources externes. Le fossé sémantique, a été abordé comme l'enjeu principal des travaux de l'état de l'art. Nous nous sommes intéressés à la prise en compte des connaissances issues des ressources externes pour réduire le fossé sémantique entre la requête et le document, selon trois niveaux : la représentation des requêtes, la représentation des documents et l'appariement requête-document. Nous présentons dans le chapitre suivant les modèles neuronaux ainsi que leur application dans le cadre de la RI.

3

RÉSEAUX DE NEURONES ET RECHERCHE D'INFORMATION

Introduction

Des améliorations considérables ont été observées ces dernières années dans des domaines d'applications tels que la vision par ordinateur, la reconnaissance de la parole et la traduction automatique. Ces progrès ont été essentiellement stimulés par les récentes avancées en apprentissage automatique, et plus particulièrement en apprentissage profond, basé sur les réseaux de neurones (Krizhevsky et al., 2012; LeCun et al., 2015).

Après les succès dans le domaine de vision par ordinateur, les progrès récents en traitement de langue ont donné lieu aux modèles neuronaux pour l'apprentissage de représentation distribuée des mots (Bengio et al., 2003; Pennington et al., 2014; Mikolov et al., 2013a) et des phrases ou documents (Le and Mikolov, 2014; Hill et al., 2016). Une *représentation distribuée* d'un mot est un vecteur dense des valeurs réelles qui encode la sémantique de ce mot, qu'on appelle un *word embedding* en anglais. Suite à l'adage souvent cité de Firth (1957) selon lequel "vous connaîtrez un mot par sa compagnie" (*"You shall know a word by the company it keeps"*), en capturant la mesure dans laquelle les mots se produisent dans des contextes similaires, ces vecteurs de mots sont capables d'encoder la similarité sémantique et syntaxique dans la mesure où les représentations des mots similaires seront proches les unes des autres dans l'espace vectoriel. En associant des mots et d'autres unités textuelles à leurs représentations, les appariements peuvent être calculés dans l'espace de représentation pour mieux capturer la sémantique de ces textes.

Pourtant, un des verrous qui a été levé est que les approches neuronales ne sont pas suffisantes pour capturer toutes les sémantiques y compris la sémantique relationnelle issue des ressources de connaissance (e. g., ontologies, base lexicale). Afin d'améliorer la sémantique des représentations distribuées, plusieurs travaux utilisent des ressources sémantiques pour injecter la sémantique symbolique dans les modèles d'apprentissage de représentation (Iacobacci et al., 2015; Yamada et al.,

2016; Liu et al., 2016). L'idée sous-jacente de ces approches est d'injecter la connaissance portée par les concepts/entités et relations entre entités/concepts pour pallier le problème de polysémie ou/et régulariser les représentations avec les relations comme le synonyme, l'antonyme, etc. En effet, l'apprentissage de représentation avec la régularisation par la ressource sémantique améliore la qualité des représentations en combinant la *sémantique distributionnelle* dans le corpus et la *sémantique relationnelle* exprimée dans une ressource sémantique.

En paralèlle, l'exploitation de la sémantique distributionnelle a émergé dans la communauté de la recherche d'information (RI) pour appliquer ces méthodes neuronales aux fins d'ordonnancement, ce qui permet de faire progresser l'état de l'art. Dans la RI, le problème de fossé sémantique exige des fonctions de similarité efficaces pour apparier les unités textuelles de différents types. Inspirés par le succès dans d'autres domaines de l'informatique et de l'intelligence artificielle, plusieurs travaux en RI ont utilisé les réseaux de neurones profonds pour un appariement sémantique (Hu et al., 2014; Shen et al., 2014a; Severyn and Moschitti, 2015). Ils consistent en des modèles d'appariement qui apprennent la pertinence des paires document-requête à partir des vecteurs sémantiques latents en utilisant des structures de réseaux de neurones profonds (plusieurs couches cachées).

Dans ce chapitre, nous rappelons tout d'abord les concepts de base des réseaux de neurones (section 1). Nous présentons ensuite, dans la section 2, les travaux qui apprennent les représentations de texte en utilisant les approches neuronales, ainsi que les applications de ces représentations en RI. Enfin, la section 3 résume des travaux qui utilisent les réseaux de neurones profonds pour apprendre la pertinence des paires document-requête.

1 Réseaux de neurones : concepts de base

Inspirés des systèmes nerveux biologiques, les réseaux de neurones artificiels sont conçus pour reconnaître des modèles de comportement dans les données. L'objectif est d'apprendre le modèle qui permet d'encoder toutes les données du monde réel (e. g., image, son, texte) en un vecteur numérique. Les réseaux de neurones sont construits selon le paradigme du neurone formel qui est introduit par Lettvin et al. (1959). La Figure 3.1 illustre les parties équivalentes entre un neurone biologique (A) et un neurone formel (B).

Les réseaux de neurones sont eux-mêmes des approximations de fonctions générales, c'est pourquoi ils peuvent être appliqués à presque tous les problèmes d'apprentissage automatique où le problème est d'apprendre un alignement complexe entre l'espace d'entrée et l'espace de sortie. Un réseau de neurones fonctionne,

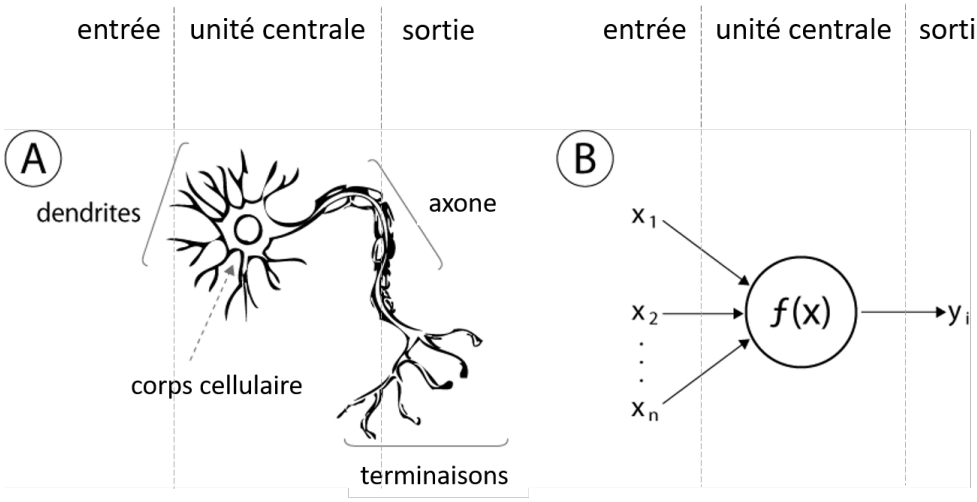


Figure 3.1 – Equivalence entre un neurone biologique (A) et un neurone formel (B)

dans un premier temps, à partir d'exemples pour apprendre les composants (paramètres) du réseau à l'aide d'une fonction objectif qui détermine une erreur d'apprentissage (phase *feed-forward*). Dans un second temps, le réseau propage cette erreur en arrière pour corriger les paramètres (phase *back-propagation*).

Le but de cette section est de rappeler quelques notions de base des réseaux de neurones (plutôt pour la classification et la régression) sans passer par les principes statistiques fondamentaux sous-jacents. Les principes et les notions plus complets et détaillés peuvent être trouvés dans Goodfellow et al. (2016). Nous présentons dans ce qui suit les quatre notions principales : modèle d'un neurone, architecture de réseau, fonction objectif et algorithme d'entraînement.

1.1 *Modèle d'un neurone*

Un réseau de neurones consiste en des nœuds de calcul reliés entre eux par des liens dirigés et pondérés. Les nœuds représentent les *neurones*, les liens pondérés représentent la force, appelée *poids*, des connexions synaptiques reliant les neurones. Un neurone peut être un sommateur des potentiels des signaux synaptiques qui lui parviennent, et transmet une information basée sur cette somme via une fonction de transfert de préférence non linéaire.

Un modèle d'un seul neurone, souvent appelé **perceptron**, est un modèle mathématique qui reçoit l'information sous la forme d'un ensemble de signaux d'entrée numériques. Ces informations sont ensuite intégrées à un ensemble de paramètres

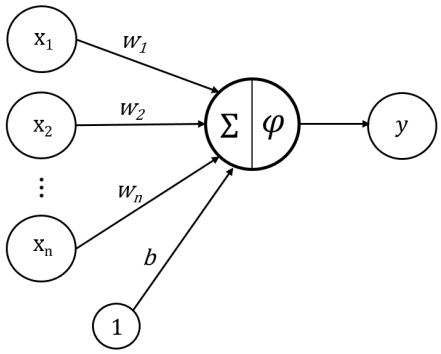


Figure 3.2 – Exemple d'un perceptron

libres pour produire un message sous la forme d'un seul signal de sortie numérique. Considérons l'architecture d'un perceptron (cf. la Figure 3.2). On identifie trois parties essentielles qui transforment des signaux entrants $(x_1, ..., x_n)$ en une seule valeur de sortie y :

— Un ensemble de *paramètres libres* θ, qui consiste en un vecteur des poids $(w_1, ..., w_n)$ et un biais b.
— Une *fonction de combinaison* Σ, qui combine les entrées avec les paramètres libres pour produire une valeur appelée l'*état interne*.
— Une *fonction d'activation* ϕ, qui prend la valeur combinée de Σ et produit la valeur sortie y.

1.1.1 *Paramètres libres*

Les paramètres libres permettent au modèle de neurone d'être entraîné pour accomplir une tâche. Dans cet exemple de perception, l'ensemble de paramètres libres θ est :

$$\theta = (b, \mathbf{w}) \in \mathbb{R} \times \mathbb{R}^n \tag{3.1}$$

où $\mathbf{w} = (w_1, ..., w_n)$ est le vecteur des poids synaptiques qui sont associés au vecteur des entrées $\mathbf{x}$ de taille n, et b est appelé le biais. Le biais est souvent représenté par un poids synaptique θ_0 relié à une entrée imaginaire x_0 fixée à 1.

1.1.2 *Fonction de combinaison*

Dans cet exemple de perceptron, la fonction de combinaison Σ calcule l'état interne du neurone par le produit scalaire entre le vecteur des entrées $\mathbf{x}$ et le

vecteur des poids synaptiques **w**. Cette fonction peut donc être formalisée comme étant une fonction vecteur-à-scalaire :

$$\Sigma(\mathbf{x};\theta) = \sum_{i=0}^{n} \theta_i x_i = b + \sum_{i=1}^{n} w_i x_i \tag{3.2}$$

1.1.3 *Fonction d'activation*

La fonction d'activation (ou fonction de transfert) ϕ sert à introduire une non-linéarité dans le fonctionnement du neurone. Elle calcule la sortie y du neurone à partir la combinaison Σ.

$$y = \phi(\Sigma) = \phi(b + \sum_{i=1}^{n} w_i x_i) \tag{3.3}$$

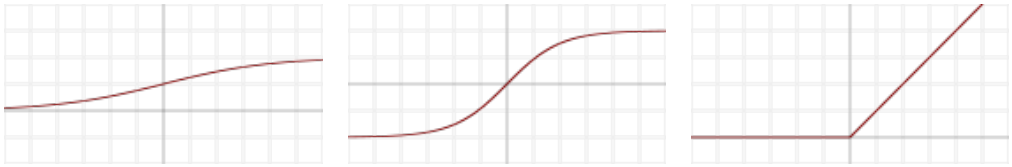
Généralement, la fonction d'activation représente trois intervalles qui correspondent à trois états du neurone :
— en dessous du seuil, le neurone est non-actif
— aux alentours du seuil, le neurone est en phase de transition
— au-dessus du seuil, le neurone est actif
En pratique, il existe plusieurs fonctions d'activation, nous présentons ici les fonctions les plus utilisées : la fonction sigmoïde (ou logistique), la fonction tangente hyperbolique (`tanh`), et la fonction Unité de Rectification Linéaire (`ReLU`). Leurs équations ainsi que leurs dérivées premières sont présentées dans le Tableau 3.1. La Figure 3.3 illustre des graphiques de ces fonctions d'activation.

Fonction	**Equation**	**Dérivée**
Sigmoïde	$\phi(x) = \frac{1}{1+e^{-x}}$	$\phi'(x) = \phi(x)(1-\phi(x))$
TanH	$\phi(x) = \frac{2}{1+e^{-2x}} - 1$	$\phi'(x) = 1 - \phi(x)^2$
ReLU	$\phi(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ x & \text{for} \quad x \geq 0 \end{cases}$	$\phi'(x) = \begin{cases} 0 & \text{for} \quad x < 0 \\ 1 & \text{for} \quad x \geq 0 \end{cases}$

Tableau 3.1 – Quelques fonctions d'activation souvent utilisées

1.2 *Architecture de réseau*

Bien qu'un seul perceptron puisse résoudre certaines tâches d'apprentissage simples, la puissance du calcul neuronal provient de la connexion de nombreux neurones dans une architecture de réseau. Un réseau de neurones est en général



(a) Sigmoïde (b) TanH (c) ReLU

Figure 3.3 – Graphiques des trois fonctions d'activation les plus utilisées

composé d'une succession de couches dont chacune prend en entrée les sorties de la couche précédente. Il existe aussi un type d'architecture récurrent où l'entrée d'un neurone peut prendre sa propre sortie. Un réseau de neurones récurrents (RNN - *Recurrent Neural Network*) est constitué de neurones interagissant non-linéairement et pour lequel il existe au moins un cycle dans la structure. Dans le contexte de cette thèse, nous détaillons dans ce qui suit l'architecture du réseau de perceptron multicouches, qui est le type de réseau classique le plus connu et souvent utilisé en RI.

Le **perceptron multicouche** (PMC) est un réseau composé de couches successives. Le but d'un réseau PMC est d'approximer une fonction $y = f_\theta(\mathbf{x})$ qui définit une projection pour une entrée $\mathbf{x}$ à une catégorie y et de corriger les paramètres θ pour une meilleure approximation de cette fonction $f_\theta(\mathbf{x})$. Le PMC est de type "propagation avant" (*feed-forward neural network* en anglais) parce que la communication est transmise de couche en couche depuis l'entrée, via les couches cachées, vers la couche de sortie. Etant donné un ensemble d'entrées, à un état spécifique des neurones du réseau, l'information est propagée dans le réseau de couche en couche pour calculer les valeurs de sortie. Une couche est un ensemble de neurones n'ayant pas de connexion entre eux.

Considérons la structure illustrée dans la Figure 3.4, le réseau PMC consiste en une séquence de c couches cachées $(l^{(1)}, ..., l^{(c)})$ et une couche de sortie $l^{(c+1)}$, de sorte que les neurones de n'importe quelle couche ne sont connectés qu'aux neurones de la couche suivante. La couche d'entrée $l^{(0)}$ se compose de n entrées externes et n'est pas comptée comme une couche de neurones. Chaque couche cachée $(l^{(1)}, ..., l^{(c)})$ est composée respectivement de $(n_1, ..., n_c)$ neurones et peut avoir sa propre fonction d'activation ϕ^i. La couche de sortie $l^{(c+1)}$ se compose de m neurones. Un neurone $u_j^{(i)}$ dans la couche cachée $l^{(i)}$ reçoit $n_{(i-1)}$ entrées venant de tous les neurones de la couche précédente. Chaque connexion entre ce neurone $u_j^{(i)}$ et un neurone $u_k^{(i-1)}$ de la couche précédente $l^{(i-1)}$ est pondérée par le poids $\theta_{jk}^{(i)}$. De cette façon, la valeur d'activation (la sortie) du neurone $u_j^{(i)}$ est calculée par

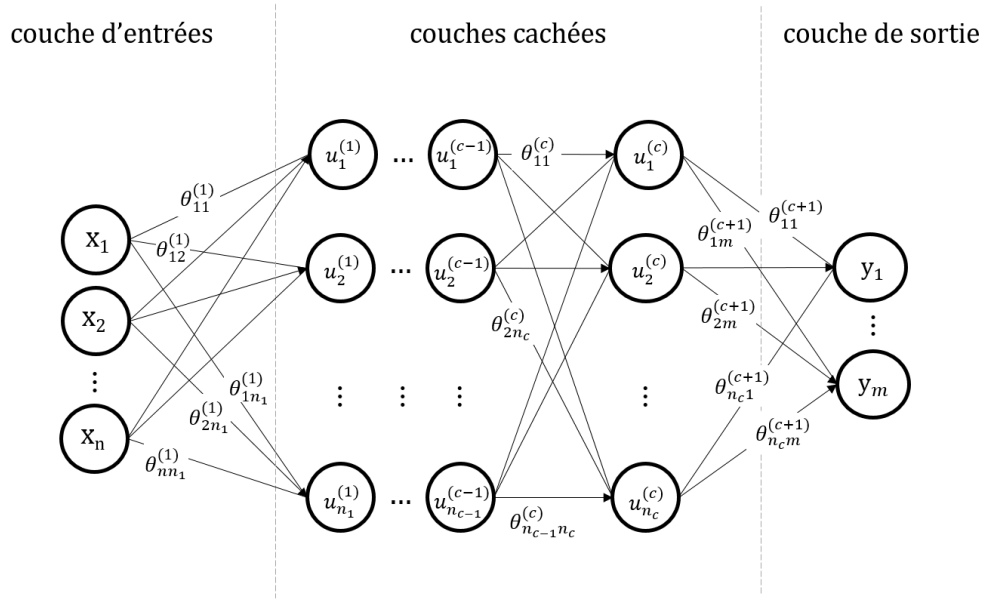


Figure 3.4 – Réseau de perceptron multicouche

l'activation du produit scalaire des poids de la couche actuelle avec les neurones de la couche précédente :

$$u_j^{(i)} = \phi^{(i)} \left(\sum_{k=1}^{n_{i-1}} \theta_{jk}^{(i)} u_k^{(i-1)} \right) \qquad i = 1, ..., c \; ; \; j = 1, ..., n_i \tag{3.4}$$

où n_{i-1} est le nombre de neurones de la couche précédente. Si $i = 1$ (pour la première couche cachée), $u_k^{(0)}$ représente le k^{e} valeur d'entrée. En appliquant le même principe aux neurones des couches cachées, chaque neurone y_j de la couche de sortie ($l^{(c+1)}$) est calculé par l'activation du produit scalaire des poids de la couche de sortie avec les neurones de la dernière couche cachée :

$$y_j = u_j^{(c+1)} = \phi^{(c+1)} \left(\sum_{k=1}^{n_c} \theta_{jk}^{(c+1)} u_k^{(c)} \right) \qquad j = 1, ..., m \tag{3.5}$$

Du point de vue vectoriel, notons $\mathbf{x}$ le vecteur des entrées $(x_0, ..., x_n)$; $\mathbf{u}_i$ le vecteur des neurones $(u_0^{(i)}, ..., u_{n_i}^{(i)})$ de la couche $l^{(i)}$; θ_i la matrice, de dimension $n_i \times n_{i-1}$, des paramètres de la couche $l^{(i)}$; et $\mathbf{y}$ le vecteur des sorties $(y_0, ..., y_m)$. On peut formuler les calculs des couches cachées et des sorties comme suit :

$$\begin{aligned} \mathbf{u}_0 &= \mathbf{x} \\ \mathbf{u}_i &= \phi^{(i)}(\theta_i \cdot \mathbf{u}_{i-1}) \qquad i = 1, ..., c \\ \mathbf{y} &= \phi^{(c+1)}(\theta_{c+1} \cdot \mathbf{u}_c) \end{aligned} \tag{3.6}$$

De cette façon, dans un réseau de neurones à propagation avant, la sortie de chaque neurone est une fonction des entrées. Ainsi, étant donné une entrée à un tel réseau neuronal, les activations de tous les neurones de la couche de sortie peuvent être calculées dans un temps déterministe.

1.3 *Fonction objectif*

Comme les autres algorithmes d'apprentissage automatique, pour évaluer ses capacités il faut concevoir une mesure quantitative de sa performance. Dans le cas de l'apprentissage supervisé, on mesure souvent la précision du modèle. La précision est la proportion des exemples (instances) pour lesquels le modèle fournit une sortie (prédiction) correcte. On peut également obtenir l'information équivalente en mesurant le taux d'erreur qui correspond à la proportion d'exemples pour lesquels le modèle produit une sortie incorrecte.
Une *fonction objectif*, parfois mentionnée comme *fonction de coût*, *fonction de perte* (en anglais *loss function*) est définie pour mesurer l'erreur d'apprentissage à l'égard des résultats. A partir de cette erreur, le réseau s'entraîne par la mise à jour de ses paramètres libres avec l'objectif de minimiser cette erreur. Dans l'apprentissage supervisé, l'erreur est mesurée par la différence entre toutes les valeurs observées des données et les valeurs calculées par le modèle. Des fonctions de coût différentes donneront des erreurs différentes pour la même prédiction et auront donc un effet considérable sur la performance du modèle. Différentes fonctions de coût sont utilisées pour traiter différents types de tâches, c'est-à-dire la régression et la classification.

Nous reprenons les notations du réseau PMC : x, le vecteur d'entrée ; $\hat{y} = f\theta(x)$, le vecteur de sortie ; y, la vraie étiquette (classe, valeur) de x ; et θ l'ensemble des paramètres libres du réseau. Nous présentons ici quelques fonctions de coût $J(\theta; x, y)$ souvent utilisées avec les réseaux de neurones pour la classification et la régression.

Type	Nom de fonction	Equation
Régression	Erreur quadratique	$J = (y - \hat{y})^2$
	Erreur absolue	$J = \lvert y - \hat{y} \rvert$
Classification	Square loss	$J = (1 - y\hat{y})^2$
	Erreur charnière (Hinge)	$J = \max\{1 - y\hat{y}, 0\}$
	Erreur logistique	$J = \frac{1}{\ln 2} \ln(1 + e^{-y\hat{y}})$
	Entropie croisée	$J = -y \ln(\hat{y}) - (1 - y) \ln(1 - \hat{y})$

Tableau 3.2 – Quelques fonctions de coût souvent utilisées

1.4 *Algorithme d'entraînement*

La procédure utilisée pour effectuer le processus d'apprentissage dans un réseau neuronal s'appelle l'algorithme d'entraînement (ou algorithme d'apprentissage). Dans l'exemple d'apprentissage supervisé, la fonction de coût est au regard des poids synaptiques (θ). Elle dispose également d'une borne inférieure, les procédures d'optimisation finissent par aboutir à une configuration stable au sein du réseau de neurones. Le but de l'apprentissage est de, étant donné des exemples x et leur sortie souhaitée y, minimiser la fonction de coût $J(\theta; x, y)$ en corrigeant les paramètres libres θ. Un algorithme d'entraînement est utilisé pour changer et ainsi entraîner le réseau de neurones, de sorte que le réseau produise une sortie souhaitée pour une entrée donnée. Nous rappelons la procédure générale de l'apprentissage supervisé qui consiste en les étapes suivantes :

— **Sélection** d'exemples d'entraînement (*input*)
— **Propagation** des entrées via le réseau, générer la **sortie** (*output*)
— **Calculer l'erreur** (loss) en comparant la sortie avec la valeur souhaitée (*label*)
— **Mettre à jour** (*update*) le réseau en fonction de l'erreur.

Pour un exemple (ou un ensemble d'exemples) en entrée, le temps de finir une telle procédure (propager les valeurs, calculer l'erreur, mettre à jour les paramètres) pour cette entrée est appelé *une itération*. Une période pour finir cette procédure pour tous les exemples d'apprentissage, est appelée *une époque*. Cette procédure d'apprentissage peut être appliquée "*par lots*" (tous les exemples d'en-

traînement sont présentés une fois, l'erreur totale est calculée pour mettre à jour le réseau) ou "*online*" (le réseau est mis à jour pour chaque exemple d'entraînement ou un petit paquet d'exemple (*mini-batch*)). Dans le cas d'apprentissage par lots, une itération est égale à une époque.

1.4.1 *Descente de gradient*

Il existe de nombreux algorithmes d'apprentissage différents pour les réseaux de neurones. Généralement, les algorithmes d'apprentissage corrigent le réseau de neurones par le principe de rétropropagation : calculer le terme de correction à partir de l'erreur (souvent la dérivée de la fonction de coût) pour chaque neurone, de la dernière couche vers la première. La **descente de gradient**, également appelée *rétropropagation du gradient*, est l'un des algorithmes les plus populaires pour réaliser l'optimisation des réseaux de neurones. La descente de gradient est un moyen de minimiser la fonction de coût $J(\theta)$ en mettant à jour les paramètres dans le sens inverse du gradient ∇J par rapport aux paramètres libres θ. Le taux d'apprentissage α détermine la grandeur des pas qu'on fait pour atteindre un minimum (local ou global). La valeur des paramètres θ à l'itération t est calculée par :

$$\theta_t = \theta_{t-1} - \alpha \cdot \nabla J(\theta) \tag{3.7}$$

Il existe trois variantes de la descente de gradient, qui diffèrent par la quantité d'exemples (échantillons) d'entrée utilisés pour calculer le gradient de la fonction de coût. En fonction de la quantité d'exemples donnés, il y a un compromis entre la qualité de la mise à jour des paramètres et le temps nécessaire pour effectuer une mise à jour.

1.4.1.1 Descente de gradient par lots

Le gradient de la fonction de coût est calculé sur l'ensemble de tous les exemples d'entrée x et leur étiquette y. Comme le gradient est calculé sur l'ensemble des données pour effectuer une seule mise à jour, la descente de gradient par lots peut être très lente et même est impossible pour les paquets de données qui ne tiennent pas dans la mémoire. La descente de gradient par lots ne permet pas non plus de mettre à jour le modèle "en ligne", c'est-à-dire avec de nouveaux exemples à la volée.

1.4.1.2 Descente de gradient stochastique (SGD)

La mise à jour des paramètres est effectuée par un tirage aléatoire de chaque exemple d'apprentissage $x^{(i)}$ et de son étiquette $y^{(i)}$. Tandis que la descente de

gradient par lots effectue des calculs redondants pour un grand ensemble de données, la SGD supprime cette redondance en effectuant une mise à jour à chaque fois. Elle est donc généralement beaucoup plus rapide et peut également être utilisée pour apprendre en ligne.

1.4.1.3 Descente de gradient par *mini-batch*

Descente de gradient par *mini-batch* : il s'agit de la meilleure solution combinant les deux approches précédentes. La mise à jour est effectuée pour chaque *mini-batch* (sous-ensemble) k de données en entrée. Autrement dit, une itération est réalisée sur un *mini-batch* d'exemples de taille k. De cette façon, elle (1) réduit la variance des mises à jour des paramètres (par rapport à la SGD), ce qui peut conduire à une convergence plus stable ; et (2) peut utiliser des optimisations matricielles hautement optimisées qui rendent le calcul du gradient beaucoup plus efficace (par rapport à la descente par lot).

1.4.2 *Optimisation de descente de gradient*

La descente de gradient ne garantit pas une meilleure convergence et ainsi pose quelques défis à relever (Ruder, 2016) :

— Le choix du taux d'apprentissage est une question importante. Un taux d'apprentissage trop élevé peut empêcher la convergence et faire varier la fonction de coût autour du minimum ou même diverger, tandis qu'un taux d'apprentissage trop faible conduit à une convergence extrêmement lente.

— En général, pour éviter de fournir les exemples d'apprentissage dans un ordre significatif au modèle, car cela pourrait biaiser l'algorithme d'apprentissage, il vaut mieux mélanger les données d'apprentissage après chaque époque.

— Lors de la minimisation des fonctions de coût non-convexes pour les réseaux de neurones, une grande difficulté est d'éviter leurs minimums locaux sous-optimaux. En effet, Dauphin et al. (2014) mettent en évidence que la plus grande difficulté vient des points-selles, c'est-à-dire des points où une dimension est inclinée vers le haut et une autre vers le bas. Ces points-selles sont généralement entourés d'un plateau de la même erreur, ce qui les rend particulièrement difficiles à éviter, car la pente est proche de zéro pour tous les points aux alentours.

Pour traiter les défis mentionnés ci-dessus, plusieurs algorithmes d'optimisation sont proposés et largement utilisés par la communauté d'apprentissage profond. Nous citons ici quelques méthodes les plus répandues dans la communauté comme : Adagrad (Duchi et al., 2011), AdaDelta (Zeiler, 2012), ADAM (Kingma and Ba, 2014).

2 Réseaux de neurones et représentations de textes

Les réseaux de neurones sont connus pour leur capacité à construire des vecteurs dans un espace latent pour capturer les informations de différents types (e. g., images, son, texte). Dans le contexte de notre thèse, on s'intéresse aux modèles neuronaux pour apprendre les représentations distributionnelles de textes.

Ces modèles, qui se basent sur la théorie du modèle de langue pour apprendre à prédire des mots sachant leur contexte, sont nommés des *modèles de langue neuronaux*. Ces modèles de langue neuronaux, considérés comme une variété des modèles basés sur la sémantique distributionnelle, ont montré qu'ils surpassent les modèles basés sur les statistiques tels que Hyperspace Analog to Language (HAL) (Lund and Burgess, 1996), Latent Semantic Analyse (LSA) (Deerwester et al., 1990), sur les tâches d'analogie des mots et de relations sémantiques (Baroni et al., 2014). Bengio et al. (2003) ont été les premiers à proposer un modèle de langue neuronal en introduisant l'idée d'apprendre simultanément un modèle de langue qui prédit un mot compte tenu de son contexte et de sa représentation, appelée "*word embedding*". Cette idée a été adoptée depuis par de nombreuses études. Les modèles de représentations distribuées les plus connus, Word2Vec (Mikolov et al., 2013a) et GloVe (Pennington et al., 2014), ont été largement utilisés dans des travaux récents dans plusieurs domaines, y compris TALN et RI. Le succès des représentations de mots (*word embeddings*) a également donné lieu à des travaux sur l'apprentissage des représentations distribuées pour des plus grandes unités textuelles, y compris les paragraphes et les documents (Le and Mikolov, 2014).

Cependant la sémantique distributionnelle présente des limites : (1) elle ne permet pas de lever le problème de polysémie puisque tous les sens d'un même mot sont représentés dans un seul vecteur (Iacobacci et al., 2015; Yaghoobzadeh and Schütze, 2016) ; en revanche ces sens sont bien distingués dans une ressource structurée ; (2) des similarités explicites entre mots telles qu'elles sont établies dans une ressource externe peuvent ne pas l'être par l'approche de comptage distributionnel si leur apparition dans les mêmes contextes est insuffisante dans le corpus ; (3) des vecteurs de représentation distribuée de mots peuvent s'avérer peu lisibles en ce sens qu'ils ne sont pas alignables avec des ressources externes ; à titre d'exemple, Mrkšić et al. (2016) ont montré que le mot "*cheaper*" se retrouve dans les mots plus proches du mot "*expensive*", en utilisant le vecteur de représentation Glove (Pennington et al., 2014). Pour aborder ces problèmes, un grand nombre de travaux exploite les ressources sémantiques pour améliorer les représentations de mots. Iacobacci et al. (2015); Yamada et al. (2016); Liu et al. (2016); Mrkšić et al. (2016). L'intuition de ces approches est d'injecter la connaissance portée par les concepts

et leurs relations pour pallier le problème de polysémie ou/et régulariser les représentations avec les relations comme la synonymie, l'antonymie, etc. Ces approches permettent d'obtenir des représentations incluant les différents sens d'un seul mot, ou aussi des représentations de concepts/entités alignées avec celles qui sont issues de la ressource externe.

Les méthodes d'évaluation de la qualité des représentations se répartissent en deux grandes catégories : l'évaluation intrinsèque et l'évaluation extrinsèque. Les évaluations intrinsèques testent directement les relations syntaxiques ou sémantiques entre les mots. Ces tâches impliquent généralement un ensemble présélectionné de termes de la requête et de mots cibles sémantiquement liés avec un score, que nous appelons inventaire de requête. Les collections les plus utilisées sont WordSim-353 (Finkelstein et al., 2001), MEN (Rubenstein and Goodenough, 1965), RG-65 (Bruni et al., 2012). Dans l'évaluation extrinsèque, on utilise les représentations de mots comme éléments d'entrée d'une tâche dédiée comme l'étiquetage grammatical, l'annotation sémantique (Pennington et al., 2014) ou la recherche d'information, puis on observe les changements dans les mesures de performance spécifiques à cette tâche.

Nous détaillons dans cette section les principaux travaux liés à l'apprentissage des représentations de textes ainsi que leurs différents niveaux de granularité, à savoir les mots, les concepts, les documents. Ces travaux sont présentés en deux grandes catégories : la première catégorie apprend des représentations directement et seulement depuis le texte du corpus, la seconde catégorie combine la sémantique distributionnelle venant du corpus et la sémantique relationnelle recensée dans les ressources sémantiques.

2.1 *Représentations distribuées de textes*

2.1.1 *Représentation des mots*

2.1.1.1 Modèle de langue neuronal

Les premiers *modèles de langue neuronaux* n'avaient pas pour objectif premier d'apprendre la représentation distribuée des mots. Cependant, les expérimentations ont démontré que la couche composante des représentations, qui aborde le problème de dimensionnalité des vecteurs de termes en entrée, fournit des représentations distribués utiles, que l'on appelle *embedding* en anglais. Le premier modèle de langage neuronal, publié par Bengio et al. (2003) est appelé ***Neural Network Language Model*** **(NNLM)**. Son architecture générale est présentée dans la Figure 3.5.

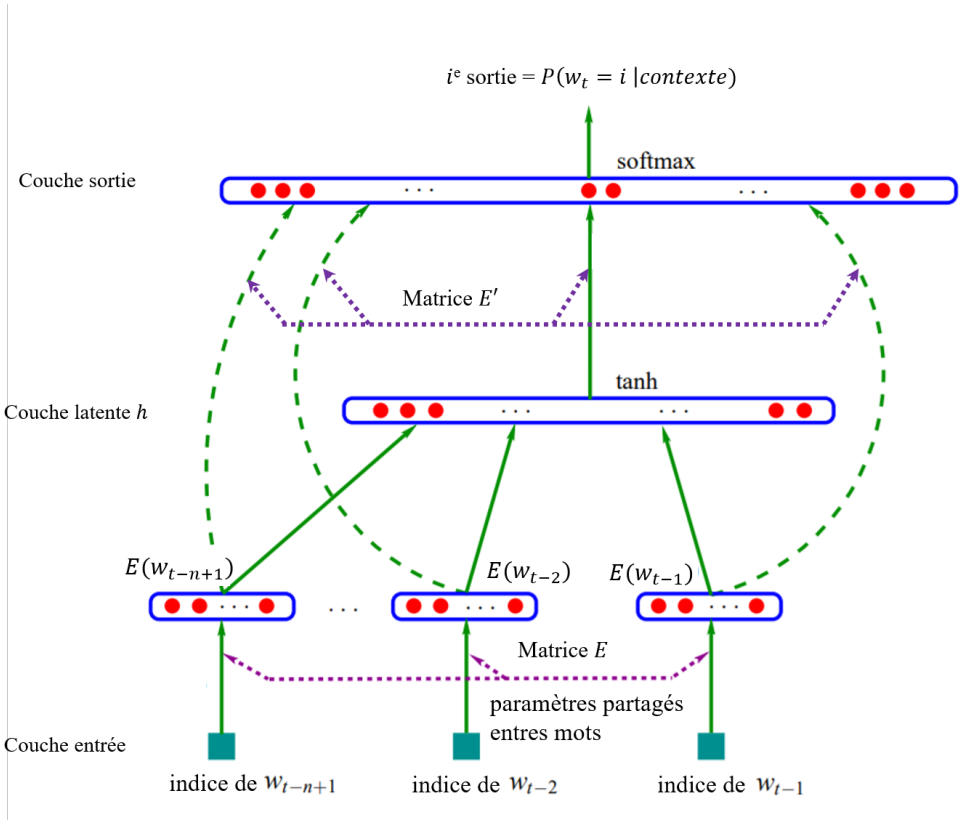


Figure 3.5 – Architecture de *Neural Network Language Model*

Pour rappel, un modèle de langue calcule la probabilité d'obtenir un ensemble de mots $P(w_1, w_2, ..., w_m)$ par l'équation suivante (Ponte and Croft, 1998) :

$$P(w_1, w_2, ..., w_m) = \prod_{t=1}^{m} P(w_t \mid w_1, \ldots, w_{t-1}) \tag{3.8}$$

Les modèles de langue probabilistes généralement approximent la probabilité $P(w_t \mid w_1, \ldots, w_{t-1})$ en considérant seulement un contexte réduit de taille n qui précède w_t :

$$P(w_1, w_2, ..., w_m) \approx \prod_{t=1}^{m} P(w_t \mid w_{t-n}, \ldots, w_{t-1}) \tag{3.9}$$

Dans les modèles de langue neuronaux, la probabilité $P(w|c)$ d'un mot w qui suit le contexte c (une séquence de mots qui précède le mot w) est calculée par un réseau de neurones. Le réseau de neurones prend un contexte c et calcule la

probabilité conditionnelle $P(w|c)$ de chaque mot w dans le vocabulaire V de la langue :

$$P(w|c,\theta) = \frac{\exp(s_\theta(w,c))}{\sum_{w' \in V} \exp(s_\theta(w',c))} \tag{3.10}$$

où $s_\theta(w,c)$ est le score de neuronal pour un mot w compte tenu du contexte c, calculé par la propagation du contexte c à travers le réseau avec l'ensemble de paramètres θ. La probabilité $P(w|c)$ est calculée par la fonction exponentielle normalisée (*softmax*) sur les scores $s_\theta(w,c)$ de tous les mots du vocabulaire.
Le réseau est entraîné sur tous les mots w_t dans le texte T d'un corpus, en utilisant un algorithme d'optimisation basé sur la descente de gradient, avec la fonction de coût :

$$J(\theta) = \sum_{(w_t,c) \in T} \log P(w_t|c,\theta) \tag{3.11}$$

Par exemple, pour une séquence de mots (w_1, w_2, w_3, w_4), l'entrée du réseau consiste en les vecteurs de termes (l'indice dans le vocabulaire) des mots dans le contexte $c = (w_1, w_2, w_3)$ pour prédire la sortie w_4. La dimension d'un vecteur de termes est $1 \times |V|$ et la taille de la matrice des représentations E est $|V| \times d$, où d est la taille de l'espace latent des représentations. Autrement dit, la i^e ligne de la matrice E est le vecteur de représentation de taille d pour le i^e mot du vocabulaire. Cette représentation d'un mot est obtenue en multipliant son vecteur de termes par la matrice E. Ainsi, chaque mot du contexte (w_1, w_2, w_3) est assigné à une représentation (e_1, e_2, e_3) de taille réduite d par rapport à la taille originale V.
La couche latente h prend les représentations (e_1, e_2, e_3) et forme un seul vecteur latent $\vec{h}_c$ du contexte c par une activation non-linéaire (tanh). Puis, en propageant vers la couche sortie, le vecteur latent h est multiplié avec la matrice des poids E' de taille $d \times |V|$ pour calculer $s_\theta(w,c) = tanh(\vec{h}_c E)'$, le score de compatibilité d'un mot w compte tenu du contexte c. Ce score est utilisé pour calculer la probabilité $P(w|c)$ selon le modèle de langue neuronal.

2.1.1.2 Optimisation d'apprentissage des modèles de langue neuronaux

Notons qu'avec la normalisation par *softmax* (Equation 3.10), on obtient ainsi un modèle de langue probabiliste correctement normalisé. Cependant, ce calcul est coûteux, car il doit calculer et normaliser la probabilité de tous les autres mots dans le contexte actuel, à chaque itération. Pour aborder ce problème, plusieurs solutions sont introduites.

Morin and Bengio (2005); Mnih and Hinton (2009) proposent la méthode **softmax hiérarchique** (SH) pour un calcul efficace de la fonction softmax. Le modèle utilise un arbre binaire pour représenter tous les mots du vocabulaire. Les V

mots doivent être des feuilles de l'arbre. On peut prouver qu'il y a $|V| - 1$ nœuds intérieurs. Le softmax hiérarchique permet d'améliorer l'efficience de l'entraînement puisque le vecteur de sortie est déterminé par une traversée arborescente des couches du réseau ; pour un exemple d'entraînement donné, le réseau ne doit calculer que $O(\log_2(|V|))$ au lieu de $O(|V|)$. Pour chaque feuille, il existe un chemin unique de la racine à la feuille et ce chemin est utilisé pour estimer la probabilité du mot représenté par cette feuille. La Figure 3.6 montre un exemple de l'arbre pour la méthode SH.

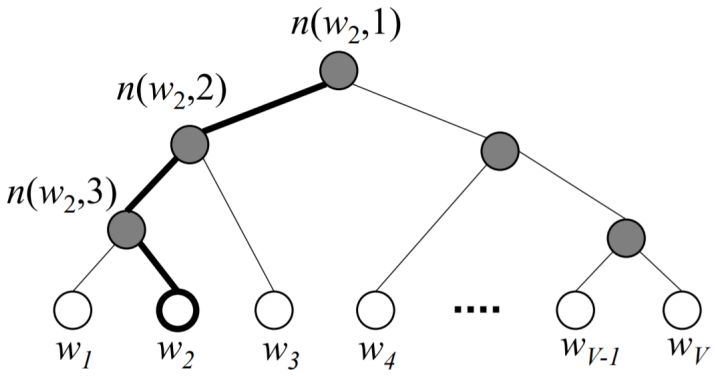


Figure 3.6 – Exemple d'un arbre pour Softmax Hiérarchique. Le chemin depuis la racine à la feuille w_2 est surligné, sa longueur est $L(w_2) = 4$. $n(w, j)$ signifie le j^e nœud sur le chemin de la racine à la feuille w

La probabilité d'un mot sachant le contexte (Equation 3.10) est remplacée par :

$$P(w|c) = \prod_{j=1}^{L(w)-1} \sigma\left([\![n(w, j+1) = \text{ch}(n(w, j))]\!] \cdot e'_{n(w,j)} \vec{h}_c\right) \tag{3.12}$$

où σ est la fonction sigmoïde ; $\text{ch}(n)$ est le fils gauche du nœud n ; $e'_{n(w,j)}$ est la représentation du nœud $n(w, j)$; $\vec{h}_c$ est le vecteur latent du contexte c ; $[\![x]\!]$ est une fonction spéciale définie comme suit :

$$[\![x]\!] = \begin{cases} 1 & \text{si } x \text{ est vrai} \\ -1 & \text{sinon} \end{cases} \tag{3.13}$$

Une autre approche proposée dans Collobert and Weston (2008) remplace la fonction softmax (Equation 3.14) par une fonction de coût (de type *Hinge*) qui ne nécessite pas de normaliser les scores sur tout le vocabulaire. Le modèle de langue neuronal est entraîné afin de calculer des scores s_θ plus élevés pour les paires de mot-contexte observées (w_t, c) par rapport aux échantillons négatifs

(w', c) construits en remplaçant w_t par un autre mot w' dans V. Le contexte est défini par les mots dans une fenêtre symétrique autour du mot central $c = (w_{t-n}, w_{t-n+1}, ..., w_{t+n-1}, w_{t+n})$.

$$J = \sum_{(w_t,c)\in T} \sum_{w'\in V} \max(0, 1 - s_\theta(w_t, c) + s_\theta(w', c)) \tag{3.14}$$

Mnih and Teh (2012) proposent l'approche d'estimation contrastive par bruit (**NCE - *Noise Contrastive Estimation***) pour aborder le problème de calcul du softmax. L'idée est de convertir un problème de classification multinomiale (softmax) en un problème de classification binaire. C'est-à-dire, au lieu d'utiliser softmax pour estimer une distribution de probabilité réelle du mot de sortie, on utilise une régression logistique binaire (classification binaire). Un ensemble de données d'apprentissage (à deux classes) est créé à partir du corpus d'entraînement en traitant les paires de mot-contexte observées (w_t, c) comme des échantillons positifs et des paires négatives (w', c) construites en remplaçant w_t par un mot w' échantillonné à partir de la distribution du bruit $\mathbb{Q}$, comme échantillons négatifs. La fonction de coût de NCE définie pour k échantillons négatifs est donnée dans l'Equation 3.15.

$$J_{NCE_k} = -\sum_{(w_t,c)\in T} \left(\log P(l = 1|w_t, c) + \sum_{i=1}^{k} \log P(l = 0|w', c) \right) \tag{3.15}$$

où les probabilités conditionnelles des classes $p(l|w, c)$ sont calculées comme suit :

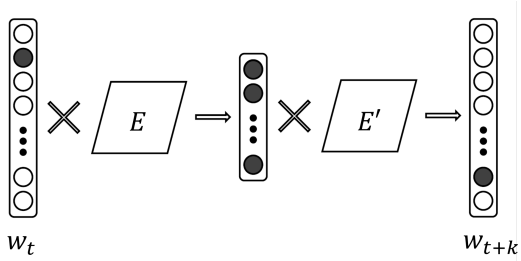
$$\begin{aligned} P(l = 1|w, c) &= \frac{\exp(s_\theta(w, c))}{\exp(s_\theta(w, c)) + k\,\mathbb{Q}(w)} \\ P(l = 0|w', c) &= 1 - \frac{\exp(s_\theta(w', c))}{\exp(s_\theta(w', c)) + k\,\mathbb{Q}(w')} \end{aligned} \tag{3.16}$$

Dans la pratique, $\mathbb{Q}$ est une distribution empirique d'unigramme uniforme, ou "aplatie" (en exponentiant chaque probabilité de $0 < \alpha < 1$ et puis re-normalisant).

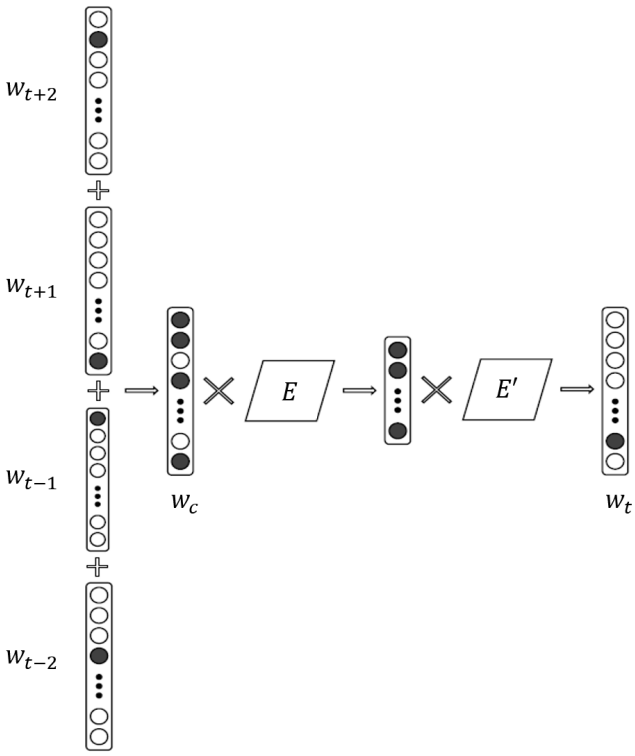
2.1.1.3 Modèles dérivés du NNLM

Deux travaux de Mikolov et ses collègues (Mikolov et al., 2013a,b) présentent des modèles efficaces, appelés **`word2vec`**, pour le calcul des représentations distribuées du texte. Le modèle `word2vec` a deux configurations CBOW et Skip-Gram. Ces modèles suivent l'architecture du modèle NNLM pour apprendre les représentations des mots. Pourtant, les auteurs ont adapté plusieurs techniques pour améliorer l'efficacité de l'apprentissage ainsi que la qualité des représentations de résultat. L'architecture des modèles `word2vec` est illustrée dans la Figure 3.7. Le premier modèle CBOW, similaire aux travaux de Collobert and Weston (2008) est

entraîné pour prédire le mot compte tenu des mots dans son contexte symétrique. La représentation de la couche cachée est calculée par la somme (ou la moyenne) des représentations des mots d'entrée. Au contraire, le modèle Skip-Gram est entraîné pour prédire chaque mot du contexte symétrique, étant donné le mot du centre.



(a) Skip-Gram



(b) CBOW

Figure 3.7 – Architecture des modèles `word2vec` : (a) Skip-Gram et (b) CBOW. Dans l'exemple de CBOW, la fenêtre de contexte est de taille deux. ©Mitra and Craswell (2018)

Dans les deux versions de `word2vec` (CBOW et SkipGram), pour chaque paire (w, c) où w est le mot à prédire et c est son contexte, la fonction de coût théorique

reprend celle de NNLM (ref. Equation 3.10). Comme `word2vec` n'utilise pas l'activation non-linéaire dans la couche cachée, le score neuronal $s_\theta(w, c)$ est calculé par le produit entre la représentation d'entrée du contexte c et la représentation de sortie du mot à prédire w :

$$s_\theta(w, c) = e'_w \cdot e_c = (\vec{w}E') \cdot (\vec{c}E) \tag{3.17}$$

où $\vec{x}$, e_x sont respectivement le vecteur de termes et la représentation du mot x. Dans le cas du modèle CBOW, w est le mot central à prédire w_t ; c est son contexte de k mots avant et k mots après, le vecteur du contexte $\vec{c}$ est généralement calculé par la somme des vecteurs dans la fenêtre k :

$$\vec{c} = \sum_{i=-k, i\neq 0}^{k} \vec{w}_{t+i} \tag{3.18}$$

Dans le cas du modèle Skip-Gram, le mot à prédire w consiste en un mot dans la fenêtre de voisinage ; et le mot central w_t devient le mot du contexte c (entrée du réseau). Pour chaque mot central w_t, le réseau va itérer $2k$ fois avec la même entrée (w_t) pour prédire $w_i \in \{w_{t-k}, ..., w_{t-1}, w_{t+1}, ..., w_{t+k}\}$ dans la fenêtre de voisinage.

L'entraînement des modèles Skip-Gram et CBOW est plus efficace par rapport à NNLM grâce à l'utilisation du softmax hiérarchique avec un arbre Huffman et l'échantillonnage négatif (**NEG - *Negative Sampling***). Le NEG est une variante du NCE (Mnih and Teh, 2012) dont la principale différence est que le NEG approxime la probabilité conditionnelle $P(l = 1|w, c)$ en la rendant plus facile à calculer. Pour ce faire, le NEG fixe le terme $k\,\mathbb{Q}(w)$ à 1, la probabilité conditionnelle $P(l = 1|w, c)$ devient :

$$P(l = 1|w, c) = \frac{\exp(s_\theta(w, c))}{\exp(s_\theta(w, c)) + 1} \tag{3.19}$$

$k\,\mathbb{Q}(w) = 1$ est toujours vrai quand $k = |V|$ et $\mathbb{Q}$ est une distribution uniforme. On peut alors transformer la probabilité $P(l = 1|w, c)$ en fonction sigmoïde σ :

$$P(l = 1|w, c) = \frac{1}{1 + \exp(-s_\theta(w, c))} = \sigma(s_\theta(w, c)) \tag{3.20}$$

En insérant dans la fonction de coût NCE (Equation 3.15), on obtient la fonction de coût NEG :

$$J_{NEG} = -\sum_{(w_t, c)\in T} \left(\log \sigma(s_\theta(w, c)) + \sum_{i=1}^{k} \log \sigma(-s_\theta(w', c)) \right) \tag{3.21}$$

Contrairement aux modèles de langue neuronaux, où les représentations sont optimisées pour maximiser la probabilité des contextes locaux (contexte autour du mot à prédire), les représentations **GloVe** (*Global Vectors*) (Pennington et al., 2014) sont entraînées pour s'adapter à la matrice de cooccurrence globale (cooccurrence de la collection). Ce modèle combine le contexte global et le contexte local dans la fonction objectif pour l'apprentissage des représentations de mots. Sa fonction de coût est définie comme suit :

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(e_i \cdot e_j + b_i + b_j - \log X_{ij} \right)^2 \tag{3.22}$$

où e_i, e_j sont les représentations des mots, b_i, b_j sont les biais, X_{ij} est le nombre de cooccurrences des mots w_i, w_j ; et $f(x)$ est une fonction de poids définie par :

$$f(x) = \begin{cases} (x/x_{max})^{\alpha} & \text{si } x < x_{max} \\ 1 & \text{sinon} \end{cases} \tag{3.23}$$

Les représentations générées par Glove et word2vec ont des performances similaires dans les évaluations de TALN. Les avantages supplémentaires de GloVe par rapport à `word2vec` sont qu'il est plus facile de paralléliser l'implémentation, ce qui signifie qu'il est plus facile de l'entraîner sur un plus grand nombre de données.

L'approche des modèles `word2vec` et Glove est efficace et utile parce qu'elle met l'accent sur le mot spécifique et les mots avec lesquels il se produit généralement. Intuitivement, on s'attend à voir des mots comme "Oktoberfest, bière, Allemagne" ou "France, Allemagne, Angleterre" être proches dans l'espace latent. Pourtant ces techniques représentent chaque mot du vocabulaire par un vecteur distinct. En particulier, elles ignorent la structure interne des mots, ce qui est une limite importante pour les langues morphologiquement riches. Bojanowski et al. (2017) proposent une nouvelle approche, appelé **fastText** basée sur le modèle Skip-Gram, où chaque mot est représenté comme un sac de caractères de n-grammes. Pour le mot *france*, par exemple, si $n = 3$, ce mot est représenté par les caractères n-grammes {`<fr, fra, ran, anc, nce, ce>`} et une séquence spéciale `<france>`. Le vocabulaire est converti pour obtenir un dictionnaire G de n-grammes. Pour un mot w donné, il est décomposé par $G_w \subset G$. Ils associent ensuite une représentation vectorielle e_g à chaque n-gramme g. Un mot est présenté par la somme des représentations de ses n-grammes. On obtient ainsi la fonction de score $s_\theta(w, c)$:

$$s_\theta(w, c) = \sum_{g \in G_w} e_g \cdot e_c \tag{3.24}$$

Ce score est utilisé pour optimiser la fonction de coût du Skip-Gram, avec l'approche de softmax hiérarchique. Les évaluations ont montré que fastText est rapide à entraîner et plus efficace dans les analyses syntaxiques ou morphologiques,

par rapport au *word2vec*. Par ailleurs, fastText peut générer de meilleures représentations pour des mots rares. En effet, même si les mots sont rares, leurs caractères n-grammes sont toujours partagés avec d'autres mots, par conséquent, les représentations peuvent aussi être bien apprises. Tandis que *word2vec* et Glove ne sont pas capables de générer les représentations des mots hors du vocabulaire, fastText peut résoudre ce problème en combinant les caractères n-grammes composants de ces nouveaux mots.

2.1.2 *Représentation des phrases, paragraphes*

Pour représenter les unités textuelles plus longues comme la phrase, le paragraphe ou le document entier, une approche simple est d'utiliser la somme ou la moyenne des représentations des mots composants (Yin and Schütze, 2015; Weston et al., 2014). Cependant, une telle composition de sac de mots ne tient pas compte de l'ordre des mots et la simple moyenne traite tous les mots avec la même importance dans la composition (bien que certains travaux aient considéré des pondérations (Vulić and Moens, 2015)). Pour aborder ce problème, plusieurs méthodes alternatives ont été proposées pour apprendre des représentations distribuées de ces unités textuelles (que nous appellerons "texte" dans ce qui suit). On peut classifier les modèles en deux catégories selon l'approche utilisée pour générer la représentation latente (*embedding*) du texte (phrase, paragraphe). La première catégorie de travaux forme le vecteur latent du texte en se basant sur les représentations des mots composants (Kenter et al., 2016; Hill et al., 2016; Arora et al., 2016). Ces derniers sont moyennés pour obtenir le vecteur latent du texte qui sera optimisé par l'apprentissage. La seconde catégorie de travaux obtient directement un vecteur de texte sans utiliser les représentations des mots composants. Ce vecteur latent est souvent constitué par un encodeur qui prend les termes comme entrée (Le and Mikolov, 2014; Zhao et al., 2018; Kiros et al., 2015; Logeswaran and Lee, 2018). Par exemple, Le and Mikolov (2014) ont proposé le `ParagraphVector` (PV) pour encoder du texte à longueur variable. Les modèles sont entraînés pour prédire un mot compte tenu de son contexte composé par des mots voisins et le document source. Contrairement au `ParagraphVector`, qui considère les phrases (paragraphes) comme des unités atomiques au lieu d'une fonction compositionnelle de ses mots, les travaux de Hill et al. (2016); Kiros et al. (2015); Kenter et al. (2016) ont gardé la compositionnalité sémantique dans l'apprentissage des représentations distribuées des phrases. Par ailleurs, dans le domaine médical, Choi et al. (2016) proposent un modèle pour apprendre des représentations appropriées des concepts médicaux tels que le diagnostic, les médicaments, les codes de procédure ainsi que les visites dans les Dossiers Electroniques du Patient (EHR - *Electronic Health Records*). Leur modèle considère les documents comme un contexte temporel directement injecté dans le processus d'apprentissage, basé sur l'architecture

Skip-Gram, qui apprend à prédire les documents voisins compte tenu d'un document et ses entités.
Nous détaillons dans ce qui suit les modèles les plus utilisés dans l'état de l'art.

• **ParagraphVector**

Le modèle **`ParagraphVector`** (Le and Mikolov, 2014) étend `word2Vec` afin d'apprendre des représentations des *paragraphes*, qui peuvent être des unités textuelles de n'importe quelle longueur. Il est composé de deux modèles distincts, à savoir *Distributed Memory* (PV-DM) et *Distributed Bag of Words* (PV-DBOW). Les architectures de PV-DM et PV-DBOW sont illustrées dans la Figure 3.8.

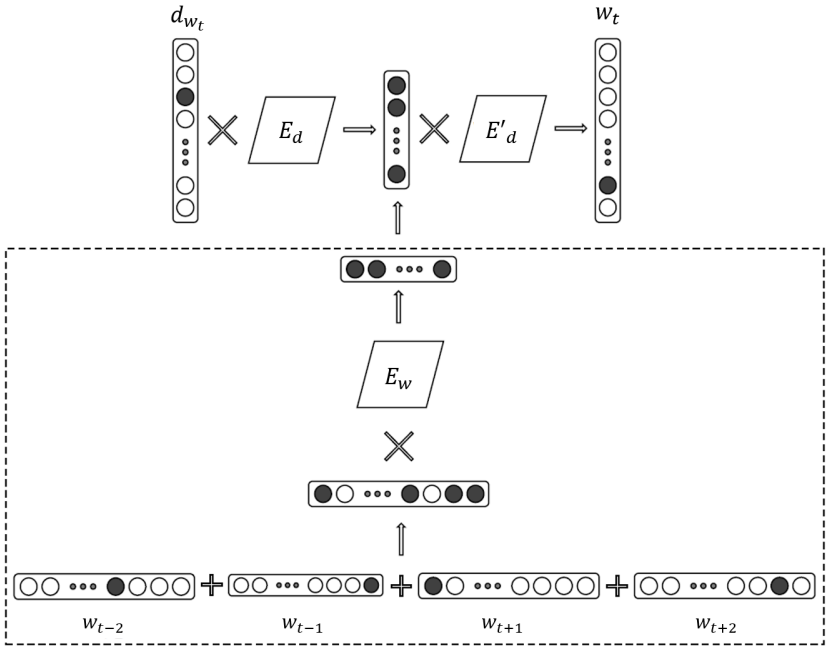


Figure 3.8 – Architecture de *ParagraphVector*. Le modèle PV-DBOW est entraîné à prédire un mot en tenant compte du document (ou paragraphe) qui contient ce mot. Dans la variante PV-DM, les mots voisins sont aussi fournis comme l'entrée - illustré par la zone en pointillés. ©Mitra and Craswell (2018)

Le modèle PV-DBOW est un Skip-Gram où l'entrée est un paragraphe au lieu d'un mot. La matrice des représentations de mots E_w est remplacée par la matrice des représentations de documents E_d (partie hors de la zone en pointillés dans la Figure 3.8). Le réseau est entraîné pour prédire un mot compte tenu du paragraphe qui contient ce mot. Le vecteur contexte c est le vecteur de représentation du document e_d, le score neuronal $s_\theta(w, c)$ est estimé comme suit :

$$s_\theta(w, c) = e_w \cdot e_d \tag{3.25}$$

En revanche, le modèle PV-DM est un CBOW étendu avec un paragraphe dans la couche d'entrée et une matrice des représentations de documents E_d (toute la Figure 3.8, y compris la zone en pointillés). PV-DM est entraîné pour prédire un mot en tenant compte du contexte d'entrée qui se compose par les mots voisins et le paragraphe qui les contient. Le vecteur contexte c est composé par la représentation du document e_d et la combinaison (la somme ou la concaténation) des représentations des mots voisins dans la fenêtre de k (comme CBOW). Le score neuronal $s_\theta(w, c)$ est calculé comme suit :

$$s_\theta(w, c) = e_w \cdot e_c = e_w \cdot (e_d \oplus e_{t-k} \oplus ... e_{t-1} \oplus e_{t+1} ... \oplus e_{t+k}) \tag{3.26}$$

où $\oplus$ peut être la somme ou la concatenation des vecteurs.

Les modèles `ParagraphVector` sont entraînés de la même façon que `word2vec` (avec l'approximation par Softmax Hiérarchique ou NEG). L'entraînement est effectué sur des collections de paragraphes (ou documents) étiquetés. Une représentation pour chaque paragraphe de la collection est apprise à la fin de l'apprentissage. La représentation d'un nouveau paragraphe (qui ne se trouve pas dans le corpus d'entraînement) peut être obtenue par une étape d'*inférence* supplémentaire. Dai et al. (2015) montrent que `ParagraphVector` surpasse les trois représentations vectorielles : (1) l'Allocation de Dirichlet latente (LDA) (Blei et al., 2003), (2) la moyenne des représentations distribuées de mots et (3) le vecteur de termes tf-idf, sur une tâche de similarité de triplets de documents, en utilisant des documents Wikipedia [1] et arXiv [2].

- **Skip-Thought**

En suivant les succès des "encodeur-décodeur" (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014), Kiros et al. (2015) proposent le modèle **Skip-Thought** qui consiste en un encodeur de type réseau récurrent (RNN) qui produit une représentation vectorielle de la phrase source et un décodeur RNN qui prédit séquentiellement les mots des phrases adjacentes. L'hypothèse sous-jacente est que, dans le contenu d'une phrase, tout ce qui conduit à une meilleure reconstruction des phrases voisines est aussi essentiel à la représentation de la phrase. L'architecture générale du Skip-Thought est illustrée dans la Figure 3.9.

Étant donné un tuple (s_{i-1}, s_i, s_{i+1}) de phrases contiguës et s_i la i^e phrase du document, la phrase s_i est encodée et entraînée à reconstruire la phrase précédente s_{i-1} et la phrase suivante s_{i+1}. Soit ${w_i}^1, ..., {w_i}^N$ les mots de la phrases$_i$ où N est le nombre de mots de la phrase. A chaque itération, l'encodeur produit un vecteur latent ${h_i}^t$ qui peut être interprété comme la représentation de la séquence ${w_i}^1, ..., {w_i}^t$. Puis, un décodeur est utilisé pour la phrase précédente, un autre est

1. https://www.wikipedia.org/
2. https://arxiv.org/

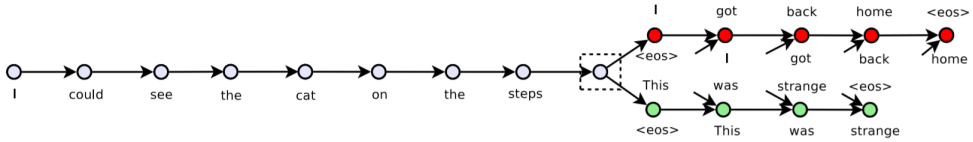


Figure 3.9 – Modèle Skip-Thought. Les flèches détachées sont connectées à la sortie de l'encodeur (carré pointillé). Les couleurs indiquent les composants qui partagent les mêmes paramètres. ⟨eos⟩ est la balise "fin de phrase".

utilisé pour la phrase suivante. Chaque décodeur a ses propres paramètres, mais les deux partagent une matrice des représentations de mots E, qui est la matrice de poids reliant le vecteur latent du décodeur pour calculer une distribution sur les mots. En utilisant la sortie ${h_i}^t$ de l'encodeur, un décodeur apprend à produire un vecteur latent h^t_{i+1} de la phrase suivante s_{i+1}, au temps t (un calcul similaire est utilisé en parallèle pour calculer h^t_{i-1} de la phrase précédente s_{i-1}). Pour prédire un mot ${w_i}^t$ dans la phrase suivante, le Skip-Thought estime la probabilité de ce mot ${w_{i+1}}^t$ sachant les $t-1$ mots précédents $w^{<t}_{i+1}$ et le vector latent de l'encodeur h_i, qui représente la phrase actuelle s_i :

$$P(w^t_{i+1}|w^{<t}_{i+1}, h_i) \propto \exp(e_{w^t_{i+1}} h^t_{i+1}) \tag{3.27}$$

Puis, l'objectif d'apprentissage est de maximiser la somme des probabilités d'obtenir la phrase suivante et la phrase précédente, en fonction de la représentation latente de l'encodeur h_i :

$$J = \sum_t \log P(w^t_{i+1}|w^{<t}_{i+1}, h_i) + \sum_t \log P(w^t_{i-1}|w^{<t}_{i-1}, h_i) \tag{3.28}$$

- **Siamese CBOW**

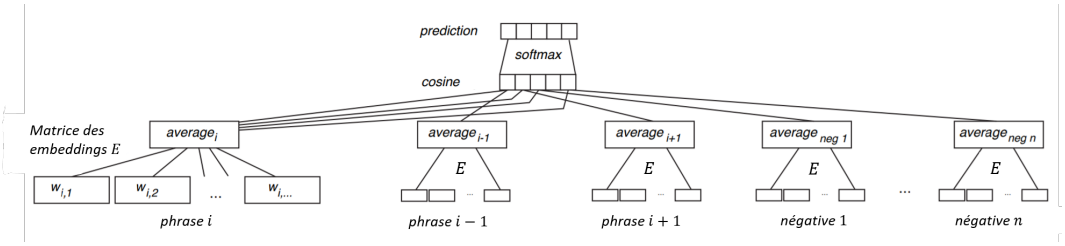


Figure 3.10 – Architecture du modèle Siamese CBOW.

Avec le même principe de prédiction d'une phrase en tenant compte des phrases adjacentes, Kenter et al. (2016) proposent le modèle **Siamese CBOW** qui calcule

la representation des phrases en moyennant les représentations des mots mais les représentations sont optimisées directement par le modèle dans le but d'obtenir la moyenne. La Figure 3.10 illustre l'architecture du modèle Siamese CBOW. Pour une phrase donnée, la couche d'entrée prend des vecteurs de termes des mots composants, puis sélectionne leurs représentations via la matrice des poids E. Les représentations des mots sont ensuite moyennées pour obtenir un vecteur latent représentant de la phrase ($average_i$ dans la Figure 3.10). Les similarités cosinus entre les représentations des phrases sont calculées dans l'avant-dernière couche, puis ces similarités sont introduites dans le calcul de la distribution de probabilité finale par un softmax.

Plus particulièrement, la probabilité $P(s_i, s_j)$ que deux phrases soient adjacentes est calculée par :

$$P(s_i, s_j) = \frac{\exp(\cos(\boldsymbol{s}_i^\theta, \boldsymbol{s}_j^\theta))}{\sum_{s' \in \{S^- \cup S^+\}} \exp(\cos(\boldsymbol{s}_i^\theta, \boldsymbol{s}_{s'}^\theta)} \tag{3.29}$$

où $\boldsymbol{s}_x^\theta$ dénote les représentations de la phrase s_x calculé par les paramètres θ du modèle. S^+ est l'ensemble de phrases adjacentes de la phrase s_i et S^- est l'ensemble de phrases d'échantillon négatif, qui ne se trouvent pas à côté de la phrase s_i. Le modèle est entraîné en utilisant le SGD pour minimiser la fonction de coût :

$$J = - \sum_{s_j \in \{S^- \cup S^+\}} f(s_i, s_j) \cdot \log P(s_i, s_j) \tag{3.30}$$

où $f(s_i, s_j)$ la probabilité cible que le réseau devrait produire, calculée par :

$$f(s_i, s_j) = \begin{cases} \frac{1}{|S^+|} & \text{si } s_j \in S^+ \\ 0 & \text{si } s_j \in S^- \end{cases} \tag{3.31}$$

L'évaluation sur la tâche SemEval (Agirre et al., 2015) a montré que le modèle Siamese CBOW surpasse le modèle Skip-Thought et les combinaisons de word2vec. Les auteurs ont aussi analysé la stabilité du modèle sur différentes configurations, et ils ont conclu que leur modèle fournissait un moyen robuste de générer des représentations de phrases de haute qualité.

- **FastSent**

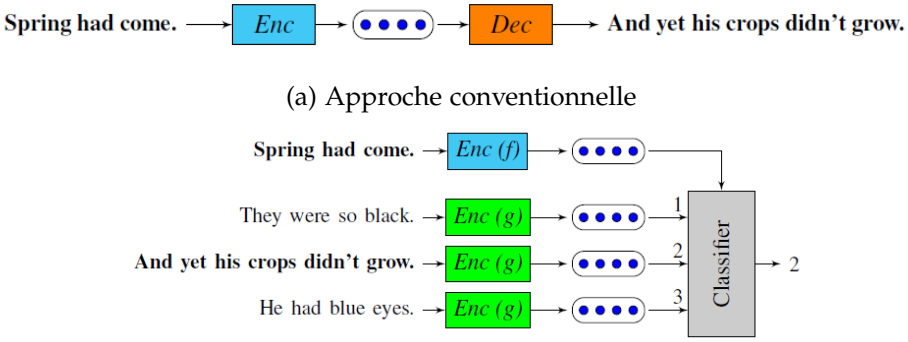
Similaire à ces deux approches précédentes, Hill et al. (2016) proposent deux modèles différents. Le premier modèle **SDAE** consiste en un auto-encodeur de débruitage, où le bruit est introduit dans une phrase en supprimant des mots et en échangeant des bi-grammes, puis le décodeur est demandé de reconstruire la phrase originale. Le deuxième modèle **FastSent** ressemble à Skip-Thought sur

l'apprentissage pour prédire des phrases adjacentes, mais la représentation des phrases est une approche de sac de mots comme dans Siamese CBOW. Plus formellement, FastSent apprend une représentation source e_w et une représentation cible e'_w pour chaque mot dans le vocabulaire. Pour un exemple d'apprentissage de phrases contigües (s_{i-1}, s_i, s_{i+1}), la représentation de s_i est représentée par la somme des représentations source e_w de ses mots : $\boldsymbol{s}_i^\theta = \sum_{w \in s_i} e_w$. La fonction de coût est aussi basée sur la fonction softmax $\phi(\bullet, \bullet)$:

$$J = - \sum_{w \in \{s_{i-1} \cup s_{i+1}\}} \phi(\boldsymbol{s}_i^\theta, e'_w) \tag{3.32}$$

Les auteurs ont montré les bonnes performances de leurs modèles sur des tâches de détection des paraphrases et de similarité sémantique. Ils ont aussi effectué une comparaison systématique de différentes méthodes d'apprentissage de représentations distribuées des phrases. Ils ont conclu que l'approche optimale dépend fortement de l'application des représentations dans des tâches supervisées ou non supervisées. Dans le dernier cas, les modèles sac de mots qui sont rapides et faciles à entraîner, peuvent toujours atteindre les meilleures performances.

- **Quick thoughts**



(a) Approche conventionnelle

(b) Approche proposée

Figure 3.11 – Aperçu des approches basées sur l'encodeur pour apprendre la représentation des phrase : (a) Approche adoptée par la plupart des travaux où le modèle tente de générer une phrase de contexte. (b) Approche proposée dans Logeswaran and Lee (2018), en remplaçant le décodeur par un classifieur.

Les approches basées sur l'encodeur-décodeur sont connues par leur performance. Elles sont cependant très lentes à entraîner sur de grandes quantités de données. En revanche, les approches basées sur le sac de mot (BOW) sont efficaces dans l'entraînement en ignorant l'ordre des mots dans le texte. Plus récemment, Logeswaran and Lee (2018) proposent un modèle appelé **Quick thoughts** qui intègre le meilleur de ces deux approches en conservant la flexibilité de l'architec-

ture de l'encodeur, tout en étant capable de s'entraîner efficacement. Similaire aux approches présentées avant, ce modèle utilise la représentation de la phrase actuelle pour prédire le sens des phrases adjacentes. Pourtant, au lieu de prendre la moyenne des représentations de mots composants, la représentation latente d'une phrase est calculée par un encodeur. Cette approche est illustrée dans la Figure 3.11b. Elle reprend le principe de Skip-gram en remplaçant les mots par les phrases. Plus formellement, on dénote f et g les fonctions d'encodeur qui prennent une phrase en entrée et encodent en un vecteur latent (*embedding*), s la phrase cible à prédire, et s_c la phrase de contexte, S_{cand} l'union de S^+, l'ensemble de phrases adjacentes de la phrase s, et S^-, l'ensemble de phrases d'échantillon négatives, qui ne se trouvent pas à côté de la phrase s. L'objectif d'entraînement est de maximiser la probabilité d'identifier les phrases de contexte correctes pour chaque phrase dans la collection D :

$$J = -\sum_{s \in D} \sum_{s^+ \in S^+} \log P(s^+ | s, S_{cand}) \tag{3.33}$$

où la probabilité $P(s_c|s, S_{cand})$ qu'une phrase $s_c \in S_{cand}$ soit correcte (i.e., la phrase adjacente de s) est calculée par le softmax, avec une fonction de score neuronal c :

$$P(s_c|s, S_{cand}) = \frac{\exp\left[c(f(s), g(s_c))\right]}{\sum_{s' \in S_{cand}} \exp([c(f(s), g(s'))])} \tag{3.34}$$

2.2 *Représentations distribuées de textes augmentées par des ressources externes*

Les représentations distribuées (*embedding*) des mots sont largement utilisées pour traiter plusieurs aspects de TALN comme la syntaxe (Turian et al., 2010), la sémantique (Socher et al., 2013), la morphologie (Luong et al., 2013). Dans de nombreux modèles TALN, elles sont sur le point de remplacer complètement les représentations distributionnelles plus traditionnelles comme le LSA (Kiela et al., 2015). Cependant, on constate que ces représentations de mots sont très sensibles à la fenêtre contextuelle définie, et malheureusement cette fenêtre contextuelle varie beaucoup selon la collection. L'un des effets de ce problème est que, dans une fenêtre contextuelle prédéfinie, les significations sémantiques de certains mots peuvent ne pas être bien représentées par les vecteurs appris. Un autre inconvénient est que la représentation d'un mot par un vecteur ne permet pas de saisir les différents sens d'un même mot, ce qui est connu par le problème de polysémie (Iacobacci et al., 2015). Par exemple, la représentation du mot "louer" ne distingue pas entre "donner en location" et "vanter les mérites de quelqu'un". D'ailleurs, l'apprentissage basé seulement sur la sémantique distributionnelle entraîne une tendance à fusionner les différentes relations sémantiques. Par exemple, Mrkšić

et al. (2016) ont démontré qu'il est difficile de distinguer les synonymes des antonymes dans les espaces de distribution.

Pour pallier ces problèmes, plusieurs approches exploitant les ressources sémantiques dans l'apprentissage des représentations distribuées des mots sont proposées. L'intuition qui guide ces travaux est d'injecter la connaissance portée par les concepts/entités et relations entre entités/concepts pour pallier le problème de polysémie ou/et régulariser les représentations avec les relations comme le synonyme, l'antonyme, etc. Particulièrement, Wikipedia fournit des alignements de mot-entité prédéfinis appelés des *ancres*. On peut considérer une ancre comme une annotation désambiguïsée d'un mot dans le texte. En utilisant ces ancres, on peut facilement obtenir de nombreuses occurrences d'entités et leurs mots de contexte directement à partir de la ressource (i.e. Wikipedia). Une autre motivation consiste à représenter des relations entre les paires de mots qui sont inventoriées dans une ressource mais pour lesquelles leurs contextes sont peu fréquents dans le corpus.

Une approche standard pour incorporer l'information externe dans les espaces vectoriels consiste à rapprocher les représentations de mots similaires (Yu and Dredze, 2014). L'intervention des ressources sémantiques peut se trouver dans différents niveaux de l'apprentissage. Le Tableau 3.3 regroupe les travaux par leur utilisation de la ressource sémantique et la nature de l'apprentissage. Nous classifions les travaux en deux grandes catégories selon l'étape d'incorporation la connaissance externe dans les représentations de mot : "en ligne" et "hors ligne".

	"En ligne"	"Hors-ligne"
Régularisation seule de mots	(Yu and Dredze, 2014; Xu et al., 2014; Liu et al., 2016; Nguyen et al., 2017d; Glavaš and Vulić, 2018)	(Faruqui et al., 2015; Mrkšić et al., 2016; Nguyen et al., 2016; Vulić et al., 2017; Vulić et al., 2018; Vulić and Mrkšić, 2018)
Apprentissage conjoint de mots et de concepts	(Cheng et al., 2015; Kiela et al., 2015; Iacobacci et al., 2015; Mancini et al., 2017; Yamada et al., 2016)	

Tableau 3.3 – Catégories des modèles d'apprentissage des représentations de mots en exploitant les ressources sémantiques.

• L'approche "*en ligne*" exploite la connaissance issue des ressources externes pendant la phase d'apprentissage de représentation distribuée des mots (Yu and Dredze, 2014; Iacobacci et al., 2015; Mancini et al., 2017; Nguyen et al., 2017d). Ces modèles modifient l'objectif original d'apprentissage distributionnel en y intégrant

les contraintes issues des ressources externes. Dans cette catégorie, les travaux de Yu and Dredze (2014); Xu et al. (2014); Liu et al. (2016) proposent d'améliorer la lisibilité des représentations des mots en utilisant les relations entre ces mots dans une ressource externe. L'intuition derrière est que des mots liés via des relations sémantiques établies dans une ressource externe sont supposés avoir des représentations proches dans l'espace latent. Un autre groupe de travaux s'intéresse à un apprentissage conjoint des éléments du corpus (à savoir les mots) et des éléments des ressources sémantiques (à savoir les concepts). Cet apprentissage conjoint dans des espaces partagés permet de mieux discriminer le sens des mots et par conséquent, résoudre le problème de la polysémie (Mancini et al., 2017; Yamada et al., 2016; Cheng et al., 2015; Iacobacci et al., 2015).
• La deuxième catégorie diffère par l'utilisation des ressources hors de l'apprentissage distributionnel (Faruqui et al., 2015; Mrkšić et al., 2016; Vulić et al., 2018; Vulić and Mrkšić, 2018). Cette approche de correction a posteriori (*hors ligne*) affine des représentations distribuées obtenues de n'importe quel modèle pour satisfaire les contraintes des ressources externes. Les méthodes issues des deux catégories utilisent des ressources lexicales ou graphes de connaissance comme WordNet (Miller, 1995), DBpedia (Lehmann et al., 2015) ou Paraphrase Database (PPDB) (Ganitkevitch et al., 2013).
Nous détaillons ces deux approches dans ce qui suit.

2.2.1 *Apprentissage en ligne des représentations de textes*

Les modèles de type "en ligne" intègrent les contraintes issues des ressources externes dans la procédure d'apprentissage distributionnel des modèles comme CBOW ou Skip-Gram. Ils modifient la fonction objectif en ajoutant une régularisation (Yu and Dredze, 2014; Xu et al., 2014; Liu et al., 2016) ou en injectant directement les contraintes dans le calcul des probabilités de contextualisation de mots permettant d'apprendre leurs représentations (Kiela et al., 2015; Cheng et al., 2015; Mancini et al., 2017).

2.2.1.1 Régularisation seule de mots

Dans cette première catégorie de travaux, Yu and Dredze (2014) utilisent des ressources externes comme une source d'évidence pour la prédiction d'un mot sachant son contexte. Ils étendent l'objectif du modèle CBOW en injectant la connaissance préalable des synonymes à partir de ressources sémantiques comme Paraphrase Database (PPDB) (Miller, 1995) et WordNet (Miller, 1995). Sur la base d'une ressource, ils apprennent les représentations afin de prédire un mot à partir d'un autre mot connecté. Pour ce faire, les auteurs dénotent $\mathbb{R}$ l'ensemble de relations entre deux mots w et w' (dans ce travail, ils supposent un seul type de relation).

En notant $\mathbb{R}_w$ le sous-ensemble de relations dans $\mathbb{R}$ qui implique le mot w, la contrainte de relation (RCM - *Relation Constraint Model*) est apprise en maximisant la probabilité d'obtenir un mot w sachant son mot relié w_i :

$$J_{RCM} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{w\in\mathbb{R}_w}\log P(w|w_i) \tag{3.35}$$

où $P(w|w_i)$ est la fonction softmax calculée avec les représentations des mots (ref. Equation 3.10). Puis les auteurs ont défini la fonction objectif du modèle joint en utilisant une combinaison linéaire (pondérée par C) entre RCM et la fonction objectif du modèle CBOW comme suit :

$$J = J_{CBOW} + J_{RCM} = -\frac{1}{|T|}\sum_{t=1}^{|}T|\log P(w_t|w_{t-k}^{t+k}) - \frac{C}{N}\sum_{i=1}^{N}\sum_{w\in\mathbb{R}_w}\log P(w|w_i) \tag{3.36}$$

Similaire au travail de Yu and Dredze (2014), les auteurs dans Xu et al. (2014) ont proposé un modèle appelé RC-NET pour exploiter des connaissances à la fois relationnelles (R) et catégoriques (C) afin de produire de meilleures représentations des mots. La connaissance relationnelle (comme *is-a*, *part-of*, etc.) encode la relation entre les entités afin de différencier les paires de mots avec des relations analogiques ; la connaissance catégorique (comme le sexe, l'emplacement, etc.) encode les attributs et les propriétés des entités, selon lesquelles des mots similaires peuvent être regroupés dans les catégories significatives.
Afin de modéliser la contrainte de la connaissance relationnelle E_r, les auteurs ont suivi l'approche de Bordes et al. (2013) qui construit des relations entre les entités en les interprétant comme des traductions fonctionnant sur les représentations des entités. Plus précisément, les connaissances relationnelles dans les ressources sont représentées dans le triplet *(tête, relation, queue)* (dénoté par $(t,r,q) \in S$, où S est l'ensemble de connaissances relationnelles), qui se compose de deux mots $t,q \in W$ (W est l'ensemble des mots) et une relation $r \in R$ (R est l'ensemble des relations). Le principe de leur modèle est que s'il existe un triplet (t,r,q), la représentation du mot de queue q doit être proche de l'addition entre la représentation du mot de tête t et le vecteur de représentation de la relation r, c'est-à-dire $(t+r)$; sinon, $(t+r)$ doit être éloigné de q. Selon ce principe, les auteurs définissent E_r comme une fonction de régularisation basée sur la marge (*hinge*), calculée sur l'ensemble des connaissances relationnelles S.

$$E_r = \sum_{(t,r,q)\in S}\sum_{(t',r,q')\in S'_{(t,r,q)}}\left[\gamma + d(t+r,q) - d(t'+r,q')\right]_+ \tag{3.37}$$

où $[x]_+$ dénote la partie positive de x ; $\gamma > 0$ est un hyper-paramètre de marge ; et $d(x,y)$ est la distance euclidienne entre x et y. L'ensemble des échantillons négatifs

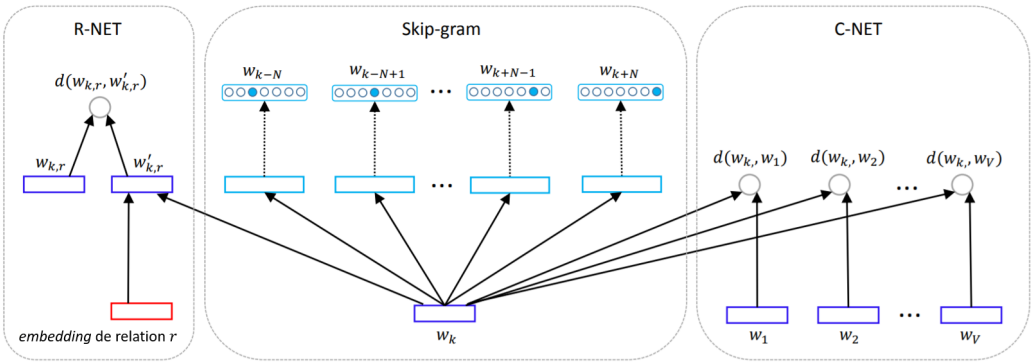


Figure 3.12 – Architecture du modèle RC-NET (Xu et al., 2014).

$S'_{(t,r,q)}$ est défini à partir de S en remplaçant soit la tête t soit la queue q par un autre mot sélectionné au hasard tel que $S \cap S' = \emptyset$.

$$S'_{(t,r,q)} = \{(t', r, q) | t' \in W\} \cup \{(t, r, q') | q' \in W\} \tag{3.38}$$

Par ailleurs, les connaissances catégoriques encodent les attributs ou propriétés des mots, à partir desquels on peut regrouper des mots similaires en fonction de leurs attributs. Xu et al. (2014) suggèrent que les représentations des mots qui appartiennent à la même catégorie doivent être proches l'une de l'autre. Ils ont observé que les connaissances catégoriques qui ont peu d'éléments sont susceptibles de contenir des informations plus spécifiques, permettant d'être plus confiant dans le regroupement des mots à partir de ces connaissances. Au contraire, les connaissances catégoriques ayant plus d'éléments sont susceptibles de contenir des informations plus générales, impactant la confiance dans le regroupement des mots correspondants. Avec cette heuristique, la somme des scores catégoriques s_{cat} entre un mot et ses mots reliés est restreinte sur tout le vocabulaire V comme suit :

$$\sum_{j=1}^{|} V | s_{cat}(w_i, w_j) = 1 \tag{3.39}$$

où si un mot w_i partage la même catégorie avec beaucoup d'autres mots, leur score de similarité mutuelle deviendra faible. Ensuite, le modèle de connaissance catégorique qui pondère la distance entre deux mots par leur score catégorique est calculé par la fonction de régularisation E_c suivante :

$$E_c = \sum_{i=1}^{|} V | \sum_{j=1}^{|} V | s_{cat}(w_i, w_j) d(w_i, w_j) \tag{3.40}$$

Finalement, le modèle global minimise la fonction de coût combinant les informations de relation et de catégorie comme ci-dessous :

$$J = \alpha E_r + \beta E_c + J_{skipgram} \tag{3.41}$$

où $J_{skipgram}$ est la fonction de coût du modèle Skip-Gram ; α et β sont respectivement des coefficients de combinaison des connaissances relationnelles et catégoriques. La Figure 3.12 illustre l'architecture générale du modèle RC-NET.

Comparé au modèle de référence (Skip-Gram), RC-NET montre de fortes améliorations en termes de qualité des représentations sur les tâches de *raisonnement analogique* (Mikolov et al., 2013a), *similarité de mots* (Finkelstein et al., 2001) et *prédiction de catégorie*.

Par ailleurs, Liu et al. (2016) ont constaté deux défauts au modèle de Yu and Dredze (2014) : (1) toute paire de mots liés dans la ressource est considérée avec une contrainte d'importance identique dans la régularisation et (2) la fonction objectif de CBOW et la régularisation sont mises à jour séparément en apprentissage, ce qui signifie que lors de la mise à jour des paramètres de la régularisation, le contexte d'un mot (considéré dans l'objectif de CBOW) n'est plus pris en compte. Ils proposent des modifications pour résoudre les problèmes évoqués ci-dessus. Une régularisation plus précise est utilisée : la fonction de coût originale CBOW est combinée avec la condition que si un mot w_t peut être bien généré à partir d'un contexte donné, son mot relié w_s, dans la ressource $\mathbb{R}$, devrait également être bien généré à partir du même contexte.

$$J = -\sum_{t=1}^{T}\left[\log P(w_t|w_{t\pm k}) - \alpha \sum_{w_s:(w_t,w_s)\in\mathbb{R}} \mathbb{W}_{st}\left[\log P(w_t|w_{t\pm k}) - \log P(w_s|w_{t\pm k})\right]^2\right] \tag{3.42}$$

où α est le coefficient de combinaison ; $\mathbb{W}_{st}$ est le poids pour distinguer les mots $w_s$reliés à un mot w_t donné, calculé par sa fréquence $f(w_s)$ dans la collection, normalisée par la somme de fréquence de tous les mots w' reliés au mot w_t :

$$\mathbb{W}_{st} = f(w_s) \Big/ \sum_{(w_t,w')\in\mathbb{R}} f(w') \tag{3.43}$$

L'entraînement du modèle est effectué sur des collections médicales OHSUMED et CLEF eHealth, qui sont aussi des collections d'évaluation pour la RI. Les auteurs ont mesuré la qualité des représentations de mot à travers une tâche de réordonnancement des documents. Le score final d'un document d compte tenu d'une requête q est une combinaison linéaire de leur score RI classique (BM25 ou modèle de langue) et leur score neuronal (cosinus de deux vecteurs). Les vecteurs

du document et de la requête sont calculés par la somme des représentations des mots composants. Les résultats obtenus sur des collections ont montré des améliorations significatives, en termes de P@10 et de MAP par rapport aux modèles de RI classique comme BM25 ou le Modèle de langue.

Dans Nguyen et al. (2017d), les auteurs proposent un modèle neuronal *HyperVec* pour apprendre les représentations hiérarchiques qui (1) discriminent l'hyperonyme des autres relations, et (ii) distinguent entre l'hyperonyme et l'hyponyme dans une paire de relations donnée. Ce modèle apprend à renforcer la similarité distributionnelle des paires d'hyperonymes en comparaison avec d'autres paires de relations, en rendant les vecteurs hyponymes et hyperonymes proches les uns des autres. *HyperVec* étend le modèle Skip-Gram (avec *Negative Sampling*) en ajoutant deux fonctions d'objectif afin d'apprendre les représentations hiérarchiques pour l'hyponyme. La première fonction $L(w, c)$ (Equation 3.44) minimise la différence distributionnelle entre l'hyponyme w et l'hyperonyme u en exploitant le contexte commun c. La deuxième fonction $L(v, w, c)$ (Equation 3.45) minimise la différence distributionnelle entre l'hyponyme w et l'hyperonyme v en exploitant le contexte commun c, qui est une caractéristique distinctive de l'hyperonyme v (i.e. le contexte commun c est plus proche de l'hyperonyme v que de l'hyponyme w).

$$L(w, c) = \frac{1}{\#(w, u)} \sum_{u \in \mathbb{H}^+(w,c)} \partial(\vec{w}, \vec{u}) \tag{3.44}$$

$$L(v, w, c) = \sum_{v \in \mathbb{H}^-(w,c)} \partial(\vec{v}, \vec{w}) \tag{3.45}$$

où $\#(w, u)$ est le nombre d'occurrences de la paire (w, u) dans la collection ; le terme $\partial(\cdot, \cdot)$ représente la dérivée cosinus entre deux vecteurs ; l'ensemble $\mathbb{H}^+(w, c)$ (ou $\mathbb{H}^-(w, c)$) consiste en, respectivement, tous les hyperonymes, (ou les hyponymes) du mot w qui partagent le contexte c et satisfont la contrainte que la différence de similarité cosinus entre les paires (w, c) et (u, c) est supérieure (ou inférieure) au seuil θ. Dans l'étape finale, la fonction objectif qui est utilisée pour apprendre les représentations hiérarchiques combine les Equations 3.44 et 3.45 avec la fonction objectif de Skip-Gram :

$$J = \sum_{w \in V_w} \sum_{c \in V_c} \left[J_{skipgram}(w, c) + L(w, c) + L(v, w, c) \right] \tag{3.46}$$

Les expérimentations effectuées sur les tâches de détection d'hyperonymie (supervisée et non-supervisée) montrent que le modèle *HyperVec* renforce la similarité de l'hyperonymie et est capable de capturer la hiérarchie distributionnelle de l'hyperonymie. Avec une mesure non supervisée *HyperScore* nouvellement proposée, les résultats ont aussi montré des améliorations significatives par rapport aux mesures de référence, ainsi que la meilleure capacité de généraliser l'hyperonymie et d'apprendre la relation au lieu de mémoriser les hyperonymes prototypiques.

2.2.1.2 Apprentissage conjoint de mots et de concepts

Les travaux de cette catégorie s'alignent aussi sur l'exploitation des connaissances dans les ressources sémantiques afin d'améliorer la cohérence sémantique ou la couverture des représentations des mots existants. Ces travaux consistent en un apprentissage conjoint des mots du corpus et des éléments des ressources sémantiques (à savoir les concepts ou les entités). Cet apprentissage conjoint dans des espaces partagés, parfois régularisés par les connaissances relationnelles issues des ressources, permet de mieux discriminer le sens des mots et par conséquent, aide à résoudre le problème de la polysémie. Par exemple, le travail de Iacobacci et al. (2015) utilise Babelfy (Moro et al., 2014), un algorithme de désambiguïsation pour obtenir les sens des mots dans le corpus Wikipedia. A partir de ce corpus annoté, où le sens désambiguïsé de chaque de mot se trouve à côté de ce mot, les auteurs appliquent le modèle CBOW pour apprendre les représentations des mots et des sens. Comme résultat, les représentations des sens peuvent capturer efficacement les distinctions claires entre les différents sens d'un même mot.

Dans Cheng et al. (2015), les auteurs proposent d'estimer dans le processus d'apprentissage, la probabilité d'association d'un concept à un mot dans la fenêtre de contexte. Ce modèle étend le modèle Skip-Gram en identifiant les paires mot-concept (vues comme des paires de mot-sens candidats) dans un contexte donné en effectuant l'entraînement conjoint de leurs représentations latentes. Par conséquent, les représentations des mots et des concepts (sens de mot) sont apprises dans le même espace latent. Les auteurs ont proposé plusieurs variantes, deux variantes de type *Generative Word-Concept Skip-Gram* (GWCS) et trois variantes de type *Parallel Word-Concept Skip-gram* (PWCS). Les architectures de ces variantes sont présentées dans la Figure 3.13. On suppose que les mots qui apparaissent souvent dans des contextes similaires ont tendance à avoir des significations similaires et qu'il faut donc leur attribuer des représentations similaires. La première variante PWCS-1 met l'accent sur les relations de cooccurrence entre le concept cible e_t et les mots contextuels w_c. La fonction objectif modifiée est de maximiser la probabilité d'obtenir le mot du contexte w_c sachant un mot central w_t et son concept e_t :

$$J = \frac{1}{T}\sum_{i=1}^{T}\sum_{\substack{c=-k \\ c\neq 0}}^{k} \log P(w_c|w_t)P(w_c|e_t) \tag{3.47}$$

Dans la deuxième variante PWCS-2, le mot cible w_t est utilisé pour prédire les mots contextuels w_c et leurs concepts e_c, avec la fonction objectif suivante :

$$J = \frac{1}{T}\sum_{i=1}^{T}\sum_{\substack{c=-k \\ c\neq 0}}^{k} \log P(w_c|w_t)P(e_c|w_t) \tag{3.48}$$

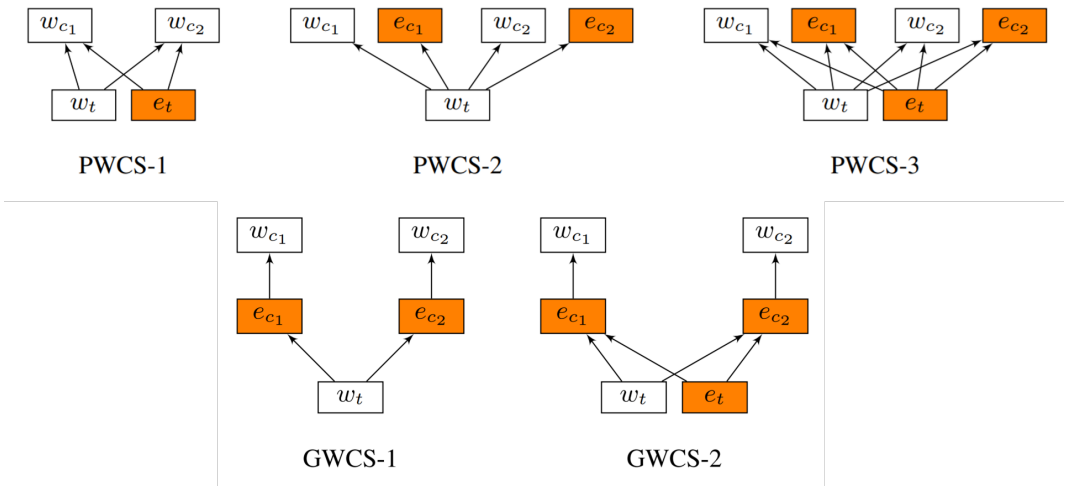


Figure 3.13 – Architecture des modèles PWCS et GWCS (Cheng et al., 2015).

En combinant les deux variantes de modèle ci-dessus, la variante PWCS-3 adopte une fonction objectif plus complète englobant toutes les relations prédicatives possibles :

$$J = \frac{1}{T}\sum_{i=1}^{T}\sum_{\substack{c=-k \\ c\neq 0}}^{k} \log P(w_c|w_t)P(w_c|e_t)P(e_c|w_t)P(e_c|e_t) \tag{3.49}$$

PWCS entraîne les représentations de concepts et de mots d'une manière parallèle, où un mot et son concept correspondant sont supposés être conditionnellement indépendants. Pour mieux mettre l'accent sur les liens entre un mot et son concept à l'intérieur d'un seul processus de prédiction, les variantes GWCS sont proposées, en décomposant la tâche de choisir un mot pour l'adapter au contexte en deux étapes : localiser le bon concept contextuel e_c et rechercher un mot w_c adapté au concept choisi. La variante GWCS-1 adopte cette fonction objectif :

$$J = \frac{1}{T}\sum_{i=1}^{T}\sum_{\substack{c=-k \\ c\neq 0}}^{k} \log P(w_c|w_t) \tag{3.50}$$

où $P(w_c|w_t) = P(e_c|w_t)P(w_c|w_t, e_c)$

GWCS-1 met l'accent sur la relation intrinsèque d'un mot contextuel w_c et de son concept e_c, mais le concept du mot cible e_t n'est pas inclus. GWCS-2 est proposé

pour lier le mot cible w_t et son concept e_t dans un processus génératif, avec la fonction objectif suivante :

$$J = \frac{1}{T}\sum_{i=1}^{T}\sum_{\substack{c=-k \\ c\neq 0}}^{k} \log P(w_c|w_t)P(w_c|e_t) \tag{3.51}$$

$$\text{où } P(w_c|w_t)P(w_c|e_t) = P(e_c|w_t)P(w_c|w_t,e_c)P(e_c|e_t)P(w_c|e_t,e_c)$$

Les modèles proposés sont évalués sur des tâches de TALN comme la similarité de mots et de groupes nominaux, la détection de paraphrase, et la classification de question-réponse. Les résultats ont montré que les représentations contextuelles des mots apprises par GWCS et PWCS surpassent de façon significative les représentations obtenues par des modèles de référence en termes de qualité et d'efficacité d'apprentissage. En comparant entre GWCS et PWCS, le modèle GWCS est légèrement supérieur à PWCS dans les tâches d'évaluation comme l'identification de paraphrase ou la similarité des mots.

Dans le même esprit d'apprendre les représentations dans un espace vectoriel partagé pour les mot et les concepts, Yamada et al. (2016) proposent des extensions du modèle Skip-Gram spécialement conçues pour la désambiguïsation d'entité nommée (*Entity Linking*). Le modèle *KB-graph* apprend la similitude des entités en utilisant les relations issues de la ressource sémantique, tandis que le modèle *anchor-context* vise à aligner les vecteurs de sorte que des mots et entités similaires se produisent à proximité les uns des autres dans l'espace vectoriel en exploitant des ancres de la ressource externe et leurs mots contextuels. Plus spécifiquement, inspiré de la mesure basée sur les liens Wikipedia (Milne and Witten, 2008), le modèle *KB-graph* apprend à rapprocher, dans l'espace vectoriel, des entités ayant des liens entrants similaires. Sa fonction objectif est calculée comme suit :

$$J_e = \sum_{e_i\in E}\sum_{\substack{e_o\in C_{e_i} \\ e_o\neq e_i}} \log P(e_o|e_i) \tag{3.52}$$

où $P(e_o|e_i)$ est calculée par une fonction softmax comme l'Equation 3.10 ; E est l'ensemble de toutes les entités dans la ressource ; C_e est l'ensemble d'entités connecté à l'entité e. Le modèle est entraîné pour prédire les liens entrants C_e pour une entité e. Ainsi, C_e joue un rôle similaire à celui des mots de contexte dans le modèle Skip-Gram. La combinaison du modèle *KB-graph* avec Skip-Gram n'a pas une connexion entre l'espace de représentation de mots et l'espace de représentation d'entités, les représentations de mots et d'entités peuvent être placées dans différents sous-espaces vectoriels. Pour résoudre ce problème, le modèle *anchor-context* est introduit avec l'idée sous-jacente d'utiliser les ancres de la ressource et leurs mots de contexte pour entraîner le modèle. Comme dans le modèle Skip-Gram,

le modèle *anchor-context* est entraîné pour prédire les mots de contexte sachant l'entité assignée par l'ancre sur ces mots. La fonction objectif est la suivante :

$$J_a = \sum_{(e_i,Q)\in A} \sum_{w\in T_i} \log P(w|e_i) \tag{3.53}$$

où A désigne l'ensemble des ancres dans la ressource, dont chacune contient une paire d'une entité désambiguïsée e_i et une séquence de mots contextuels T_i. En combinant les deux objectifs précédents avec le modèle Skip-Gram, les auteurs forment la fonction objectif suivante :

$$J = J_{skipgram} + J_e + J_a \tag{3.54}$$

Une fois obtenu des représentations entraînées avec le modèle final, les auteurs ont proposé leur propre méthode de désambiguïsation d'entité nommée. Les évaluations ont montré que leur méthode de désambiguïsation, basée sur les représentations apprises par leur modèle, surpasse plusieurs méthodes de désambiguïsation de l'état de l'art.

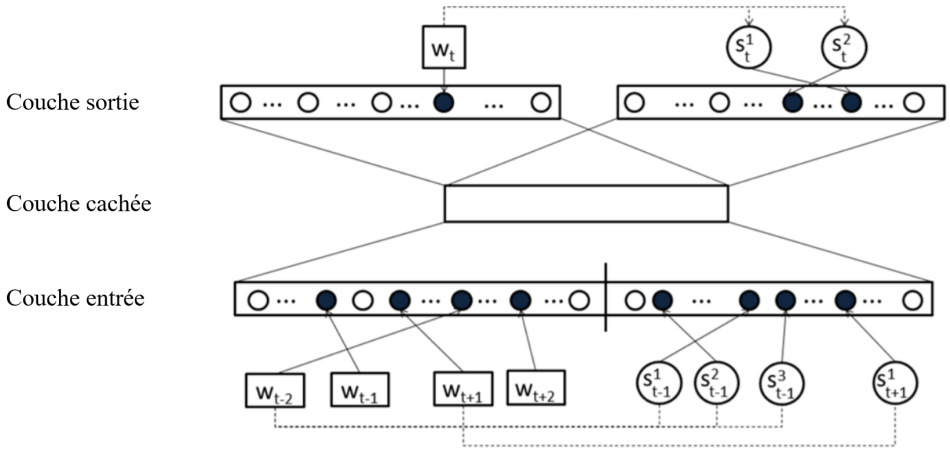


Figure 3.14 – Architecture du modèle SW2V (Mancini et al., 2017) avec la fenêtre de contexte de taille 2. Les lignes pointillées représentent le lien virtuel entre les mots et les sens associés.

Mancini et al. (2017) proposent aussi un modèle d'apprentissage conjoint des représentations de mots et de concepts (sens de mot) en exploitant les connaissances issues des ressources sémantiques, appelé SW2V. Basé sur l'architecture du modèle CBOW, leur modèle ajoute aux couches d'entrée et de sortie des sens de mots en exploitant la relation intrinsèque entre les mots et les sens. L'intuition est que, comme un mot est un symbole d'un sens sous-jacent, la mise à jour de la représentation du mot devrait produire une mise à jour conséquente de la représentation

de ce sens spécifique, et vice-versa. En appliquant un algorithme d'identification des sens basée sur WordNet, un mot donné peut avoir zéro, un ou plusieurs sens assignés. Dans ce modèle, chaque mot cible w_t prend comme contexte à la fois les mots qui l'entourent $w_{t\pm k}$ (dans la fenêtre k) et tous les sens associés à ces mots $S_{t\pm k}$. Contrairement à l'architecture originale du CBOW, où l'objectif d'apprentissage est de classifier correctement w_t, cette approche vise à prédire le mot w_t et son ensemble des sens associés S_t. Cela équivaut à minimiser la fonction de coût suivante :

$$J = -\log P(w_t|w_{t\pm k}, S_{t\pm k}) - \sum_{s\in S_t} \log P(s|w_{t\pm k}, S_{t\pm k}) \tag{3.55}$$

où S_i est l'ensemble de sens associés au mot w_i. L'architecture de ce modèle est illustrée dans la Figure 3.14. Entraîné sur le corpus Wikipédia et UMBC (Han et al., 2013), leur modèle est capable de construire un espace vectoriel des mots et des sens sémantiquement cohérent. Les résultats quantitatifs (tâche similarité de mots, clustering de sens) et qualitatifs (les mots/sens plus similaires) ont montré des améliorations significatives du modèle proposé par rapport aux modèles de référence y compris CBOW.

2.2.2 *Apprentissage a posteriori des représentations*

Les travaux dans cette catégorie tentent d'améliorer la représentation de mots pré-entraînée dans une étape de correction a posteriori, aussi appelé par *retrofitting*. Compte tenu de la représentation de mots pré-entraînée, l'idée principale de la correction a posteriori est de rapprocher des mots qui sont reliés par une relation définie dans une ressource sémantique donnée. Particulièrement, Vulić et al. (2017) élargissent cette lignée de travaux en injectant des contraintes morphologiques générées à l'aide des règles simples au lieu d'utiliser les relations issues des ressources sémantiques.

La première introduction de la correction a posteriori (*retrofitting*) est le travail de Faruqui et al. (2015), qui propose une méthode pour affiner les représentations dans l'espace vectoriel à l'aide des informations relationnelles issues des lexiques sémantiques en incitant les mots connectés à avoir des représentations vectorielles similaires. Cette méthode ne fait aucune hypothèse sur la façon dont les représentations d'entrée ont été construites. Elle encourage les nouvelles représentations à être (i) similaires aux représentations de mots reliés dans la ressource et (ii) similaires à leurs représentations purement distribuées. Selon la modélisation des auteurs, une ressource Ω est composée d'un ensemble de nœuds $\mathcal{V}$ (qui correspondent aussi aux mots du vocabulaire de la collection), d'un ensemble de types de relation R et d'un ensemble des liens L où chaque lien $l \in L$ est un triplet (i, j, r) dont la relation $r \in R$ connecte deux nœuds $i \in \mathcal{V}$ et $j \in \mathcal{V}$. Etant données

des représentations de mots $\hat{E} = \hat{e}_i : i \in \mathcal{V}$ entraînées par n'importe quel modèle, le but de la correction a posteriori est d'apprendre un ensemble de nouvelles représentations $E = e_i : i \in \mathcal{V}$ qui contiennent les informations encodées à la fois les connaissances de la sémantique distributionnelle à partir des corpus de texte et de la structure de la ressource sémantique externe. En utilisant la distance euclidienne comme la distance sémantique entre deux représentations de mots, les auteurs ont défini l'objectif du *réajustement* des représentations par la minimisation du problème des moindres carrés pondérés.

$$\Psi(E) = \sum_{i=1}^{|\mathcal{V}|} \left[\alpha_i \|e_i - \hat{e}_i\|^2 + \sum_{(i,j)\in L} \beta_{ij} \|e_i - e_j\|^2 \right] \tag{3.56}$$

où α et β sont des valeurs qui contrôlent les puissances relatives de la combinaison. Les auteurs ont expérimenté leur méthode sur les différents types de représentation (GloVe, Skip-Gram, etc.) avec des ressources lexicales comme PPDB, WordNet, FrameNet (Baker et al., 1998). Les résultats sur différentes tâches (Similarité de mot, Relations Syntactiques, etc.) ont montré que cette méthode améliore significativement la qualité des représentations et aussi qu'elle dépasse la performance des méthodes alternatives comme celle de Yu and Dredze (2014) et Xu et al. (2014).

Dans le même esprit de la correction a posteriori, Mrkšić et al. (2016) proposent une méthode appelé *counter-fitting* qui injecte les contraintes d'antonymie et de synonymie dans la représentation vectorielle pour améliorer la capacité des vecteurs à évaluer la similarité sémantique. L'intuition est d'approcher les paires de mots synonymes et d'éloigner les paires antonymes, en conservant la sémantique distributionnelle apprise sur la collection. Etant donné l'ensemble des représentations entraînées $\hat{E} = \hat{e}_i : i \in \mathcal{V}$, le but de l'apprentissage est d'obtenir un ensemble de nouvelles représentations $E = e_i : i \in \mathcal{V}$ augmentées par les contraintes d' antonymie et de synonymie. L'antonymie et la synonymie ont un ensemble de contraintes A et S, respectivement, qui contient des paires de mots reliés par la relation correspondante. La fonction objectif utilisée pour adapter les vecteurs de mots $\hat{E}$ préentraînés aux ensembles de contraintes linguistiques A et S contient trois termes différents. Le premier terme AR sert à éloigner les vecteurs de mots antonymes les uns des autres dans l'espace vectoriel E :

$$AR(E) = \sum_{(i,j)\in A} \tau(1 - d(e_i, e_j)) \tag{3.57}$$

où $d(e_i, e_j) = 1 - cos(e_i, e_j)$ est la distance dérivée du cosinus et $\tau(x) = max(0, x)$ impose une fonction de coût de type *hinge-loss*.
Le deuxième terme SA cherche à rapprocher les vecteurs de mots synonymes :

$$SA(E) = \sum_{(i,j)\in S} \tau(d(e_i, e_j)) \tag{3.58}$$

Le dernier terme VSP permet de rapprocher le plus possible l'espace vectoriel transformé de l'espace vectoriel original afin de préserver les informations sémantiques contenues dans le vecteur original.

$$VSP(E, \hat{E}) = \sum_{i=1}^{|\mathcal{V}|} \sum_{j \in N(i)} \tau(d(e_i, e_j) - d(\hat{e}_i, \hat{e}_j)) \tag{3.59}$$

où $N(i)$ désigne l'ensemble des mots dans un certain rayon ρ autour du i^e vecteur du mot dans l'espace vectoriel d'origine. La fonction objectif finale de la procédure est donnée par la somme pondérée des trois termes.

$$C(E, \hat{E}) = k_1 AR(E) + k_2 SA(E) + k_3 VSP(E, \hat{E}) \tag{3.60}$$

où $k1, k2, k3 \geq 0$ sont des hyper-paramètres qui contrôlent l'importance relative de chaque terme.
Les auteurs ont montré que la méthode apprend efficacement les vecteurs de mots pour améliorer leur performance dans les tâches de similarité sémantique (e.g., SimLex-999). L'accent mis sur la séparation des représentations des paires de mots antonymes conduit à des améliorations substantielles sur les tâches d'estimation de similarité de mots.

Vulić and Mrkšić (2018) utilisent aussi l'idée d'"attirer-repousser" (*attract-repel*) pour rapprocher les vraies paires hyponymie-hyperonymie (l'implication lexicale) dans l'espace euclidien transformé en utilisant des ressources linguistiques (e.g., WordNet).

L'intuition, illustrée par la Figure 3.15, est de rapprocher les paires de mots souhaitables (*attirer*) décrites par les contraintes, tout en éloignant les paires de mots indésirables (*repousser*) les unes des autres. En même temps, cette méthode restreint les normes vectorielles de sorte que les valeurs des normes dans l'espace euclidien reflètent l'organisation hiérarchique des concepts en fonction des contraintes d'hyponymes données : les concepts plus génériques auront des normes plus grandes. Similaire au travail de Mrkšić et al. (2016), les auteurs ont utilisé une fonction objectif combinée par des termes qui attirent et repoussent des paires de mots souhaitables en conservant la sémantique distributionnelle des représentations. Particulièrement, ils ont ajouté à la fonction objectif (ref. Equation 3.60) un terme LE (*Lexical Entailment)* pour mettre en relief la distance hiérarchique de l'implication lexicale. Contrairement à la similarité symétrique, l'implication lexicale impose une distance asymétrique qui encode un ordre hiérarchique entre

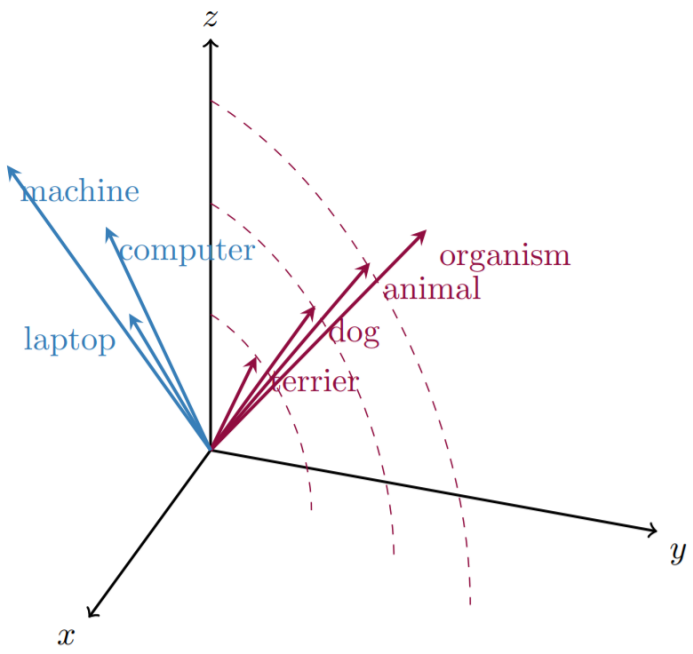


Figure 3.15 – Illustration de l'espace vectoriel transformé proposé dans Vulić and Mrkšić (2018). Le modèle contrôle la disposition des vecteurs dans l'espace vectoriel transformé en 1) mettant l'accent sur la similarité symétrique des paires d'hyponymie par la distance cosinus (en imposant de petits angles entre $\overrightarrow{terrier}$ et $\overrightarrow{dog}$ ou $\overrightarrow{dog}$ et $\overrightarrow{animal}$) ; et en 2) imposant un ordre d'hyponymie en utilisant des normes vectorielles, en les ajustant de sorte que les concepts de niveau supérieur aient des normes plus larges (e.g., $|\overrightarrow{animal}| > |\overrightarrow{dog}| > |\overrightarrow{terrier}|$).

les concepts. Trois différentes distances sont utilisées, en utilisant les normes du vecteur de mot pour imposer un ordre entre les concepts parents et enfants.

$$D_1(x, y) = |x| - |y| \tag{3.61}$$

$$D_2(x, y) = \frac{|x| - |y|}{|x| + |y|} \tag{3.62}$$

$$D_3(x, y) = \frac{|x| - |y|}{max(|x|, |y|)} \tag{3.63}$$

Etant donné L, l'ensemble des paires d'implication lexicale directe comme *(terrier,dog)*, *(dog,animal)*, avec le concept plus spécifique à gauche, le concept plus générique à droit. Le terme LE_j (pour la j^e distance asymétrique) dans la fonction objectif est défini par :

$$LE_j(L) = \sum_i = 1^{|L|} D_j(x_i, y_i) \tag{3.64}$$

Expérimentés sur des tâches de génération des implications lexicales, les résultats ont montré des améliorations significatives par rapport aux modèles de référence, et aussi la capacité de généraliser l'hyperonymie.

2.3 *Utilisation des représentations distribuées de texte en RI*

Les modèles de RI traditionnels utilisent des représentations locales (discrètes) des termes pour l'appariement des requêtes et des documents. Cependant, il est important d'inspecter dans le document les termes non liés (différents) à la requête pour obtenir des preuves de pertinence. Dans les approches de RI basées sur le comptage des termes traditionnels, ces signaux sont souvent ignorés. L'avantage principal d'utiliser les représentations distribuées de mots dans la RI est de permettre un appariement inexact dans l'espace vectoriel. Les représentations de mots peuvent être incorporées dans les approches de RI existantes en deux grandes catégories : celles qui **comparent la requête avec le document directement dans l'espace de représentation** ; et celles qui **utilisent la représentation pour générer des candidats d'extension** de la requête appropriés. Une autre approche qui utilise les représentations de mot dans les réseaux de neurones pour RI sera présentée dans la section 3. Nous présentons dans ce qui suit les approches pour incorporer les représentations distribuées dans les modèles de RI.

	Travaux
Appariement document-requête	(Nalisnick et al., 2016; Mitra et al., 2016; Zuccon et al., 2015; Rubner et al., 2000)
Expansion de la requête	(Roy et al., 2016; Zamani and Croft, 2016a; Diaz et al., 2016; Zamani and Croft, 2016b)

Tableau 3.4 – Catégories des modèles qui utilisent des représentations distribuées de mots pour la RI.

2.3.1 *Utilisation dans l'appariement document-requête*

L'une des stratégies d'utilisation des représentations pour la RI consiste à dériver une représentation vectorielle dense pour la requête et le document à partir de la représentation des mots individuels dans les textes correspondants. Les représentations de mots peuvent être agrégées de différentes manières dont la moyenne des représentations de mot (AWE - *Average Word Embeddings*) est la plus courante (Vulić and Moens, 2015). La requête et les documents peuvent être comparés à l'aide d'une variété de mesures de similarité, comme la similarité cosinus (ref.

Equation 3.65) ou le produit scalaire. Une autre façon d'agréger les représentations de mots est de pondérer l'AWE par le TF-IDF des mots.

$$sim(q,d) = cos(\vec{q},\vec{d}) = \frac{\vec{q}\cdot\vec{d}}{||\vec{q}||\,||\vec{d}||} \tag{3.65}$$
$$\text{où, } \vec{q} = \frac{1}{|q|}\sum_{t_q\in q}\frac{\vec{t}_q}{||\vec{t}_q||} \text{ et } \vec{d} = \frac{1}{|d|}\sum_{t_d\in d}\frac{\vec{t}_d}{||\vec{t}_d||}$$

Nalisnick et al. (2016); Mitra et al. (2016) ont analysé les caractéristiques des représentations word2vec pour l'estimation de la pertinence d'un document pour une requête. Ils ont mis en évidence qu'il est plus approprié de calculer la similarité IN-OUT entre la requête et les termes du document. Autrement dit, les termes de la requête doivent être représentés à l'aide de la représentation IN de mots $t_{q,IN}$ et les termes du document à l'aide de la représentation OUT de mots $t_{d,OUT}$ (ref. Section 2.1.1 pour les matrices de représentations IN-OUT). Le modèle à double espace de représentations (DESM - *Dual Embedding Space Model*) estime la pertinence du document-requête comme suit :

$$DESM(q,d) = \frac{1}{q}\sum_{t_q\in q}\frac{\vec{t}_{q,IN}\cdot\vec{d}_{OUT}}{||\vec{t}_{q,IN}||\,||\vec{d}_{OUT}||} \tag{3.66}$$
$$\vec{d}_{OUT} = \frac{1}{|d|}\sum_{t_d\in d}\frac{\vec{t}_{d,OUT}}{||\vec{t}_{d,OUT}||}$$

Le DESM est évalué dans la tâche d'ordonnancement des documents, avec à la fois le retour de pertinence explicite et implicite proposé dans Mitra et al. (2016). Sur le jeu de données avec le retour explicite, DESM surpasse les modèles de base comme BM25 et LSA. Cependant, lorsque le DESM est utilisé seul pour classer les documents d'une grande collection de tests, sa performance est significativement inférieure à celle des modèles LDA et BM25. Le DESM est démontré être un signal efficace pour ré-ordonner les documents.

Une autre possibilité de représenter des requêtes et des documents sous la forme d'une agrégation de leurs représentations de mots est d'intégrer les représentations des mots dans les modèles de RI existants, comme le modèle de langue de traduction neuronale (NTLM - *Neural Translation Language Model*) proposé dans Zuccon et al. (2015). La probabilité $P(t_q|d)$ qu'un terme t_q de la requête soit dans un document d est estimé en utilisant une translation de probalité d'obtenir ce terme t_q via chaque terme t_d du document.

$$P(t_q|d) = \sum_{t_d\in d} P(t_q|t_d)\times P(t_d|d) \tag{3.67}$$

où $P(t_d|d)$ est la probabilité qu'un terme t_d appartienne au document d et $P(t_q|t_d)$ est la probabilité de conversion de terme à terme. Cette dernière est estimée en utilisant la similarité entre les représentations de mots. Plus précisément, la probabilité de conversion de terme à terme $P(t_q|t_d)$ est estimée par une fonction softmax (ref. Equation 3.10) sur les représentations de mots :

$$P(t_q|t_d) = \frac{cos(\vec{t}_q, \vec{t}_d)}{\sum_{t \in V} cos(\vec{t}, \vec{t}_d)} \tag{3.68}$$

Une autre approche, basée sur l'*Earth Mover's Distance* (EMD (Rubner et al., 2000), consiste à estimer la similarité entre les paires de documents en calculant la distance minimale dans l'espace de représentation que chaque terme du premier document doit parcourir pour atteindre les termes du second document. Cette mesure, appelée *Word Mover's Distance* (WMD), a été proposée à l'origine par Wan and Peng (2005), qui utilisent WordNet et les catégories de sujets pour définir la distance entre les termes. Kusner et al. (2015) et Huang et al. (2016) proposent d'utiliser les représentations distribuées de mots pour calculer le WMD. Chaque document est représenté sous la forme d'un ensemble de points dans l'espace de représentation de mots. Le WMD de deux documents est la distance euclidienne cumulative minimale que tous les mots du premier document doivent parcourir pour correspondre exactement au deuxième document. La mesure de WMD est évaluée sur la classification de documents *kNN* sur différents jeux de données et différents types d'unités textuelles telles que des recettes, des journaux, des résumés médicaux, des tweets et des phrases. Les résultats ont montré que WMD surpasse les modèles de référence BoW, BM25, LSI, LDA en termes de taux d'erreur.

Les modèles basés sur la représentation de mots ont souvent de faibles performances lorsque l'appariement est effectué sur l'ensemble de la collection de documents (Mitra et al., 2016). Cependant, plusieurs modèles de RI arrivent à surpasser les modèles d'appariement classique en réordonnant un sous ensemble de documents candidats. Cette méthode est courante en RI neuronale (Huang et al., 2013; Shen et al., 2014a; Guo et al., 2016; Mitra et al., 2017) et les résultats sont représentatifs sur les tâches de réordonnancement. Toutefois, Mitra et al. (2016) ont montré que de bons résultats dans les tâches de réordonnancement n'impliquent pas automatiquement de bonnes performances sur des collections de documents plus importantes.

2.3.2 *Utilisation dans l'expansion de la requête*

Au lieu d'apparier la requête et le document directement dans l'espace latent, une autre approche consiste à utiliser des représentations distribuées de mots pour

trouver de bons candidats à l'expansion de la requête. La sélection des mots candidats s'effectue à partir d'un vocabulaire global, puis la requête étendue est utilisée pour récupérer les documents en utilisant un modèle de RI classique. Différents travaux ont proposé des méthodes pour estimer la pertinence des termes candidats par rapport à la requête. Ces approches consistent à comparer individuellement le terme candidat à chaque terme de la requête en utilisant leurs représentations vectorielles, puis les scores sont agrégés pour chaque candidat.

Par exemple, Diaz et al. (2016) et Roy et al. (2016) estiment la pertinence du terme candidat t_c de la façon suivante :

$$score(t_c, q) = \frac{1}{|q|} \sum_{t_q \in q} cos(\vec{t}_c, \vec{t}_q) \tag{3.69}$$

Plus précisément, Roy et al. (2016) proposent un ensemble de méthodes d'expansion de requêtes basées sur la sélection de k plus proches voisins des termes de la requête dans l'espace de représentation. Puis ces termes étendus sont ordonnés en fonction de leur similarité avec l'ensemble de la requête. L'espace de recherche pour les voisins couvre l'ensemble du vocabulaire ou se limite aux termes des documents les mieux classés à partir d'une recherche initiale. Toutes les méthodes d'expansion ont donné des scores d'efficacité inférieurs à ceux d'une méthode de retour de pseudo-pertinence (PRF), dans les expérimentations sur les jeux de données de TREC 6, 7 et 8 et TREC Robust par le modèle de langue avec lissage Jelinek Mercer.
L'expansion des requêtes basée sur la représentation de mots est moins performante que le (PRF) (Roy et al., 2016). Mais comme les modèles de réordonnancement, les performances sont meilleures lorsqu'ils sont combinés avec le PRF (Zamani and Croft, 2016a).
Dans Diaz et al. (2016), les représentations de mots sont utilisées pour définir un nouveau modèle de langue de requête (QLM). Le QLM spécifie une distribution de probabilité $P(t_c|q)$ sur tous les termes du vocabulaire. Dans l'expansion de requêtes avec le modèle de langue, les m termes t_c qui ont la valeur $p(t_c|q)$ la plus élevée sont sélectionnés comme termes d'expansion. Diaz et al. (2016) ont exploré l'idée de la représentation de mots spécifiques aux requêtes et ont constaté qu'ils sont plus efficaces pour identifier les bons termes d'expansion qu'une représentation globale. Ce modèle "local" intègre le retour de pertinence dans le processus d'apprentissage des représentations - un ensemble de documents spécifiques au sujet de la requête est récupéré pour entraîner les représentations. Pour choisir les documents spécifiques au sujet d'une requête, les auteurs calculent le score de divergence Kullback-Leibler D_{KL} entre les modèles de langue du document M_d et

de la requête M_q (Equation 3.70). Ces modèles de langue sont calculés en utilisant l'approche dans Croft and Lafferty (2013).

$$D_{KL}(M_q||M_d) = \sum_{w \in V} M_q(w) \log \frac{M_q(w)}{M_d(w)} \tag{3.70}$$

Les documents dont les modèles de langage sont plus similaires au modèle de langage de requête auront un score de divergence KL plus faible. Puis, cet ensemble de documents spécifiques à la requête est utilisé pour entraîner un modèle de représentations de mots reliés au sujet de la requête QLM (e. g., appliquant le modèle word2vec sur chaque ensemble de documents). Ce modèle de représentations est ensuite utilisé pour identifier les candidats à l'expansion de la requête. Les termes candidats sont pondérés en utilisant la similarité des représentations distribuées de mots par rapport à la requête. Soit E la matrice de représentations distribuées des mots du vocabulaire, $\mathbf{q}$ le vecteur de termes de la requête, le poids des termes candidats est $EE^{\intercal}\mathbf{q}$. Les représentations entraînées localement sont comparées aux représentations globales sur les corpus de web : TREC12, Robust et ClueWeb 2009. Il est montré que la représentation locale permet d'obtenir des scores NDCG@10 plus élevés.

Zamani and Croft (2016b) proposent un modèle théorique pour l'estimation des vecteurs de représentation de requêtes basés sur les vecteurs de représentation individuels des termes. L'intuition est de trouver la représentation optimale $\vec{q}^*$ qui maximise la vraisemblance entre un modèle de langue de la requête $P(w|M_q)$ et une distribution probabiliste $P(\vec{w}, \vec{q})$ (ref. Equation 3.71).

$$\vec{q}^* = \arg\max_{\vec{q}} \sum_{w \in V} P(w|M_q) \log P(\vec{w}, \vec{q}) \tag{3.71}$$

où la probabilité $P(w|M_q)$ d'obtenir le mot w sachant le modèle de langue de la requête M_q peut être estimée par le maximum de vraisemblance $P_{MLE}(w|M_q)$ par la fréquence de mot w dans la requête q ; la distribution probabiliste $P(\vec{w}, \vec{q})$ est calculée par la similarité vectorielle $\delta(\vec{w}, \vec{q})$ entre la représentation du mot $\vec{w}$ et la représentation de la requête $\vec{q}$. Puis, pour pondérer les termes d'expansion t_c à la requête q, les auteurs calculent d'abord la distribution probabiliste $P(\vec{w}, \vec{q^*})$ entre ce terme t_c et la représentation optimale $\vec{q}^*$ de la requête. Ce score est combiné (avec le coefficient α) avec l'estimation du maximum de vraisemblance de la requête originale $P_{MLE}(w|M_q)$ comme suit :

$$P(t_c|M_{q^*}) = \alpha P_{MLE}(t_c|M_q) + (1 - \alpha) P(\vec{t_c}, \vec{q}^*) \tag{3.72}$$

3 Réseaux de neurones profonds pour la RI

Au cours des dernières années, les réseaux neuronaux profonds ont mené à des progrès remarquables dans plusieurs domaines comme la reconnaissance de la parole, la vision par ordinateur, le traitement du langage naturel, et plus récemment la recherche d'information. Dans cette section, nous étudions les travaux sur les modèles neuronaux pour l'appariement sémantique requête-documents. Ce type de travaux utilise des architectures de neurones profondes pour apprendre une fonction d'appariement, ce qui réfère à l'apprentissage d'ordonnancement. Cela se distingue des modèles de la section précédente, qui ne s'appuient pas sur les informations de pertinence pour l'ordonnancement ou pour l'apprentissage des représentations des termes. Pour rappel, les modèles d'apprentissage d'ordonnancement ont pour but de classifier un élément (e. g., une paire requête-document) à partir d'un vecteur des caractéristiques $\vec{x} \in \mathbb{R}^n$. Le modèle d'ordonnancement $f : \vec{x} \rightarrow \mathbb{R}$ est entraîné pour apparier le vecteur à un score réel de telle sorte que, pour une requête donnée, les documents les plus pertinents sont mieux notés et qu'une métrique choisie basée sur le rang est maximisée. L'entraînement du modèle est dit "de bout en bout" (*end-to-end*) si les paramètres de f sont appris en une seule fois plutôt qu'en parties, et si le vecteur $\vec{x}$ consiste en des caractéristiques simples plutôt que des modèles. Les modèles neuronaux de RI "de bout en bout" peuvent apprendre à la fois les vecteurs de représentation et la similarité entre les textes donnés en entrée.

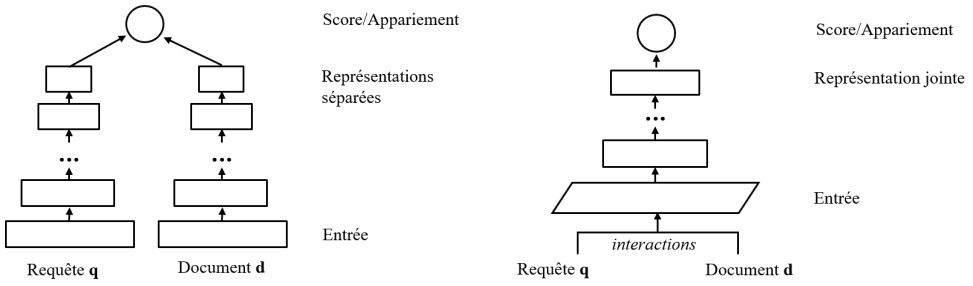


Figure 3.16 – Deux types d'architecture des modèles neuronaux profonds pour RI : (a) modèles basés sur la représentation et (b) modèles basés sur les interactions.

En appliquant des réseaux de neurones profonds à la recherche ad-hoc, la tâche est généralement formalisée comme un problème d'appariement entre deux textes (i. e., la requête et le document). Une telle formalisation d'un problème d'appariement est souvent généralisée en ce sens qu'elle peut couvrir à la fois des tâches

de recherche ad-hoc ainsi que de nombreuses tâches de TALN telles que l'identification des paraphrases, les réponses aux questions (QA) et les conversations automatiques (Hu et al., 2014; Lu and Li, 2013). Comme discuté dans Guo et al. (2016), les modèles neuronaux pour la RI ont tendance à suivre l'une des deux architectures générales que nous décrivons dans la Figure 3.16. Dans le premier type (Figure 3.16a), les requêtes et les documents sont traités séparément à travers des réseaux de neurones (avec des paramètres partagés) et les scores de similarité sont calculés à partir des représentations latentes des sorties respectives. Les exemples de ce type sont DSSM (Huang et al., 2013), C-DSSM (Shen et al., 2014a), ARC-I (Hu et al., 2014). En revanche, la Figure 3.16b décrit une approche dans laquelle une représentation commune pour les requêtes et les documents, qui prend en compte les interactions entres les entités dans les deux textes, est construite et ensuite un réseau de neurones profond est utilisé pour apprendre un appariement basé sur les modèles d'interaction. Les travaux adoptant cette approche incluent DeepMatch (Lu and Li, 2013), ARC-II (Hu et al., 2014), MatchPyramid (Pang et al., 2016) et DRMM (Guo et al., 2016).

Le problème fondamental de la recherche ad-hoc (i. e., le calcul de la pertinence d'un document pour une requête spécifique) peut être formalisé comme un problème de l'appariement de deux textes comme suit. Etant donnés deux textes T_1 et T_2, le niveau d'appariement est mesuré comme un score produit par une fonction de score basée sur la représentation Φ de chaque texte :

$$RSV(T_1, T_2) = F(\Phi(T_1), \Phi(T_2)) \tag{3.73}$$

où Φ est une fonction assignant chaque texte à un vecteur représentation, et F est la fonction de score basé sur les interactions entre les deux textes.
On trouve deux catégories en fonction du choix des fonctions F et Φ.

• La première catégorie, basée sur la représentation, cherche à construire une bonne représentation pour un texte unique avec un réseau neuronal profond, puis procède à l'appariement entre deux représentations textuelles compositionnelles et abstraites. Dans cette approche, Φ est une fonction de transformation complexe tandis que F est une fonction d'appariement relativement simple. Par exemple, dans DSSM (Huang et al., 2013), Φ est un réseau de neurones de type *feed-forward*, tandis que F est la fonction de similarité cosinus. Dans ARC-I (Hu et al., 2014), Φ est un réseau de neurones convolutif, tandis que F est un perceptron multicouche. En général, l'architecture de ces modèles peut être classifiée dans le type de réseau Siamois (symétrique) comme illustré dans la Figure 3.16a.

• La seconde catégorie, basée sur les interactions, construit d'abord les interactions locales entre deux textes à partir des représentations de base, puis utilise des réseaux de neurones profonds qui apprennent les modèles d'interaction hiérarchique pour l'appariement. Dans cette approche, Φ est généralement une

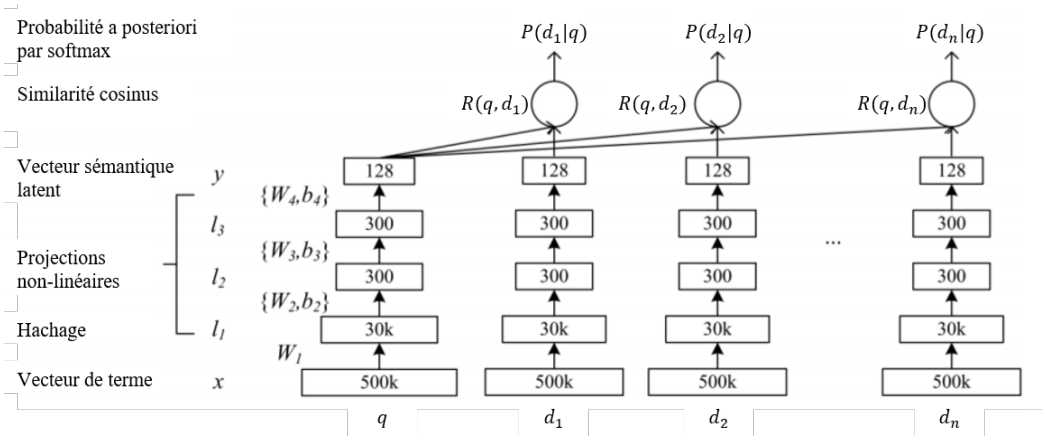


Figure 3.17 – Architecture de réseau d'appariement DSSM proposé dans Huang et al. (2013).

simple fonction de représentation tandis que F est un réseau profond complexe. Par exemple, dans DeepMatch (Lu and Li, 2013), Φ applique simplement chaque texte à une séquence de mots, tandis que F est un réseau de neurones augmenté par un *topic model* sur la matrice d'interaction des mots. Dans ARC-II (Hu et al., 2014), Φ applique pour chaque texte une séquence de vecteurs de mot, tandis que F est un réseau convolutif appliqué sur la matrice d'interactions entre les vecteurs de mots. Les modèles de ces deux approches sont présentés dans ce qui suit.

	Travaux
Basé sur la représentation	(Huang et al., 2013; Shen et al., 2014a,b; Hu et al., 2014; Palangi et al., 2016; Severyn and Moschitti, 2015)
Basé sur les interactions	(Lu and Li, 2013; Hu et al., 2014; Guo et al., 2016; Mitra et al., 2017; Pang et al., 2016)

Tableau 3.5 – Catégories des modèles qui utilisent des réseaux de neurones profonds pour la RI.

3.1 *Modèles basés sur la représentation*

Comme présentés précédemment, les modèles basés sur la représentation adoptent l'architecture du réseau de neurones de type Siamois pour apprendre à apparier deux textes. Le travail pionnier de Huang et al. (2013) propose un modèle sémantique de structure profonde (DSSM) pour la recherche ad-hoc sur le Web (Figure 3.17). Le réseau consiste en deux branches profondes symétriques dont les paramètres sont partagés - pour la requête q et le document d - appli-

quées sur les vecteurs de termes comme l'entrée. Toutes les couches cachées sont de type perceptron multicouches, enchaînées pour obtenir une représentation sémantique latente intermédiaire. Ces représentations latentes (du document et de la requête) sont utilisées ensuite dans la couche de similarité mesurée par le cosinus. Les auteurs proposent d'entraîner le modèle sur les données de clics où chaque échantillon d'apprentissage consiste en une requête q, un document pertinent d^+ (un document qui a été cliqué par un utilisateur pour cette requête), et un ensemble de documents négatifs D^- échantillonnés au hasard avec une probabilité uniforme à partir de la collection entière. L'appariement est appris par DSSM en maximisant la probabilité conditionnelle d'un document pertinent à une requête donnée. Ainsi, pour chaque paire de requête-document pertinent, le modèle est entraîné à minimiser la fonction de coût suivante :

$$J(q, d^+) = -\log\left(\frac{e^{\gamma cos(\vec{q},\vec{d^+})}}{\sum_{d\in D} e^{\gamma cos(\vec{q},\vec{d})}}\right) \tag{3.74}$$

où $D = \{d^+\} \cup D^-$ est l'ensemble des documents négatifs et le document pertinent d^+ ; $\vec{x}$ est la représentation sémantique latente obtenue de la dernière couche cachée ;γ est un facteur de lissage qui est défini empiriquement sur un ensemble de données. La requête et le document sont d'abord modélisés en entrée sous la forme de deux vecteurs de termes de grande dimension. DSSM apprend une représentation de documents et de requêtes via un réseau de type *feed-forward* pour obtenir un vecteur à faible dimension projeté dans un espace sémantique latent. Compte tenu de la taille importante du vocabulaire et de la nécessité d'un apprentissage à grande échelle, les auteurs proposent une méthode de hachage de mots qui transforme le vecteur de termes à haute dimension de la requête/document en un vecteur des lettre-trigramme à dimension réduite. Le hachage trigramme permet également d'aborder des termes hors vocabulaire qui n'apparaissent pas dans les données d'entraînement. Il convient de noter que les documents sont indexés uniquement par le texte du titre plutôt que par le texte intégral. Le travail ultérieur de Guo et al. (2016) effectue les expériences avec DSSM et les résultats indiquent que la recherche en texte intégral avec DSSM ne fonctionne pas aussi bien que les modèles de RI traditionnels.

Alors que DSSM (Huang et al., 2013) utilise une architecture de perceptron multicouches pour apparier la requête et le document, des architectures plus sophistiquées ont également été explorées en impliquant des couches convolutives (Shen et al., 2014a,b; Hu et al., 2014) ou des couches récurrentes (Palangi et al., 2016). Par exemple, les travaux de Shen et al. (2014a,b) étendent DSSM en introduisant un réseau convolutif (CNN) dans l'architecture DSSM (C-DSSM). Ce modèle est aussi appelé Modèle de Sémantique Latente à Convolution (CLSM - *Convolutional Latent Semantic Model*). Son architecture générale adopte le réseau Siamois comme DSSM, avec la couche d'entrée qui reçoit les vecteurs de termes du document et

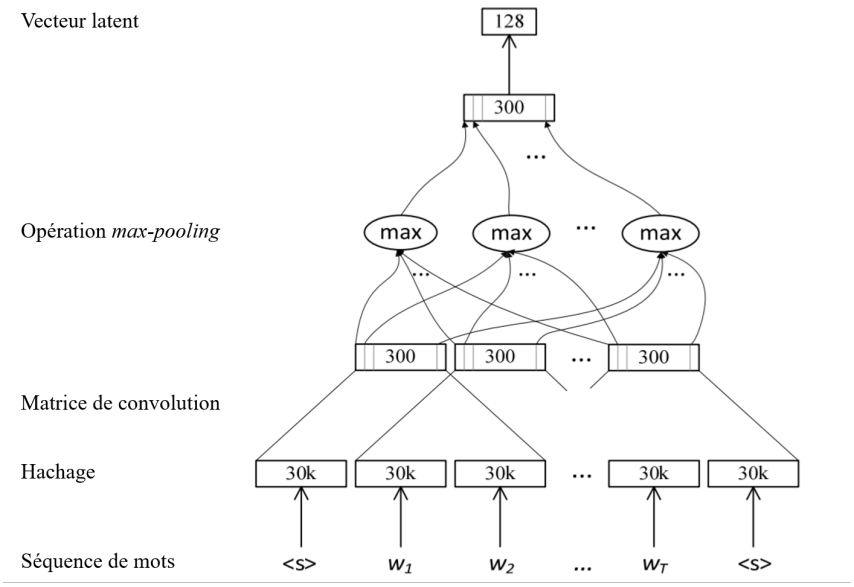


Figure 3.18 – Architecture de réseau d'appariement C-DSSM proposé dans Shen et al. (2014a).

de la requête. Une branche du réseau est illustrée dans la Figure 3.18. Les auteurs appliquent aussi la technique de hachage de mots pour transformer en vecteur des lettre-trigramme. Une couche de convolution transforme ces trigrammes vers un vecteur contextuel. Le réseau intègre également une couche *max-pooling* pour extraire les caractéristiques locales les plus significatives pour former un vecteur global de longueur fixe pour les requêtes et les documents. La principale motivation du *max-pooling* est que la signification sémantique globale d'une phrase est souvent déterminée par quelques mots clés, une combinaison simple de tous les mots (e. g., la somme de tous les vecteurs de caractéristiques locales) peut permettre d'introduire des divergences inutiles et nuire à l'efficacité globale de la représentation sémantique. Il s'agit de la différence principale entre DSSM et C-DSSM. Enfin, en utilisant la transformation non linéaire *tanh* et le perceptron multicouches, le réseau calcule le vecteur sémantique latent final pour la requête et le document. Les paramètres sont entraînés à maximiser la même fonction de coût utilisée pour DSSM. Même si C-DSSM introduit la technique de convolution pour capturer l'information contextuelle, il souffre des mêmes problèmes que DSSM. Par exemple, le CLSM obtient de moins bons résultats lorsqu'il est entraîné sur un document entier par rapport à un entraînement sur le titre du document, comme signalé dans Guo et al. (2016).

La fonction de similarité peut également être paramétrée et implémentée en tant que couches supplémentaires du réseau neuronal comme dans Severyn and Moschitti (2015). Les auteurs proposent aussi un réseau de neurones profond à convo-

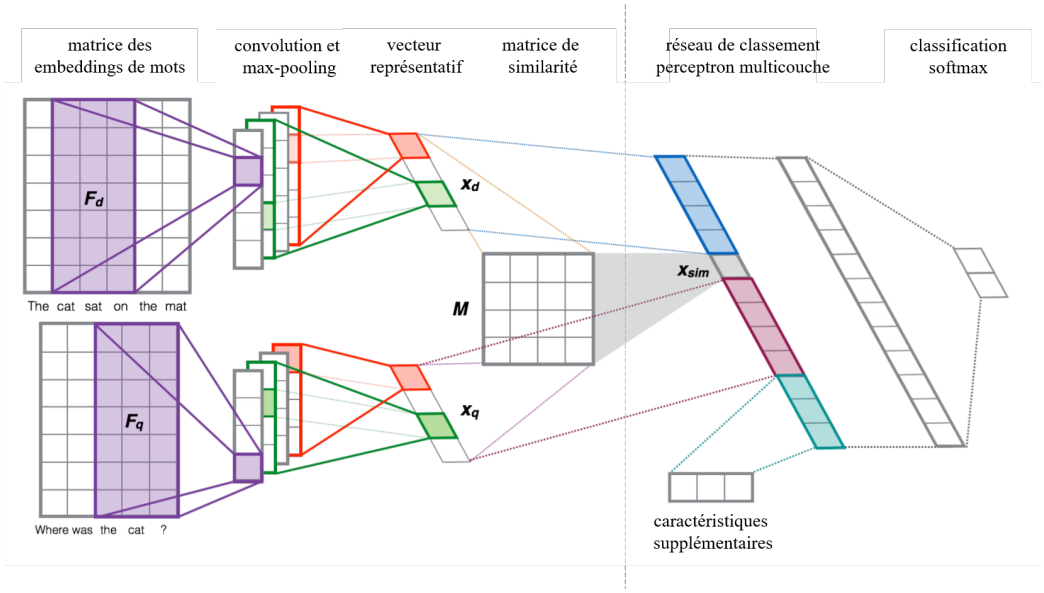


Figure 3.19 – Architecture de réseau d'appariement proposé dans Severyn and Moschitti (2015). Les deux parties sont séparées par la ligne pointillée.

lution pour classer des paires de textes courts. Leur architecture d'apprentissage profond comporte deux parties (Figure 3.19). La première partie est un modèle de représentation de phrases utilisant un CNN pour apprendre à projeter des phrases de questions et réponses dans des vecteurs représentatifs intermédiaires, qui sont utilisés pour calculer leur similarité. Particulièrement, dans ce travail, les auteurs proposent de calculer le score de similarité intermédiaire en utilisant la multiplication matricielle entre les vecteurs représentatifs intermédiaires et une matrice de poids à entraîner. La deuxième partie est un réseau de neurones de classification qui reçoit les caractéristiques de la partie précédente : deux vecteurs de phrases représentatifs intermédiaires, un score de similarité et quelques caractéristiques supplémentaires telles que le chevauchement de mots entre les phrases. Les résultats montrent une amélioration significative en termes de MAP par rapport aux résultats rapportés par Yu et al. (2014), dans lequel les auteurs utilisent un CNN suivi d'une régression logistique pour classer les paires de question-réponse.

3.2 *Modèles basés sur les interactions*

Les réseaux siamois cherchent à apprendre des représentations latentes à taille réduite pour la requête et le document en utilisant des transformations via les couches de neurones. Alternativement, on peut comparer individuellement différentes parties de la requête avec différentes parties du document, puis agréger ces

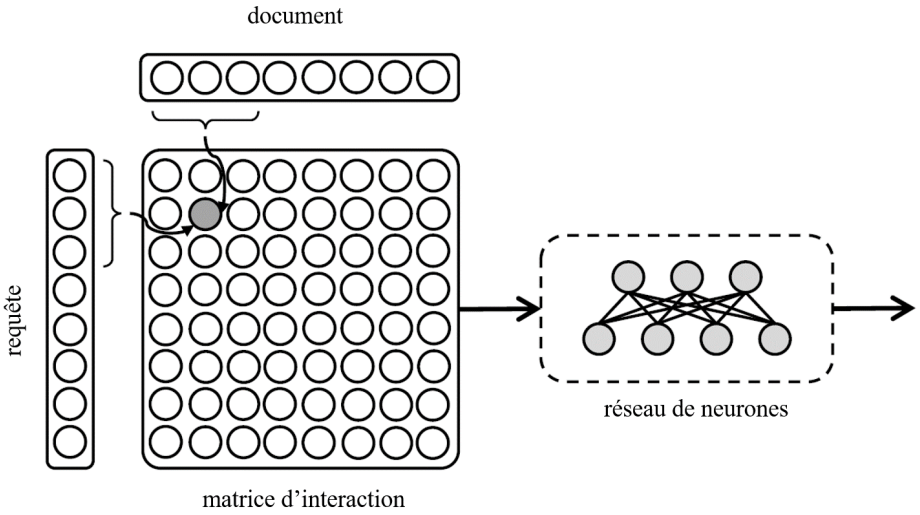


Figure 3.20 – Schéma d'une matrice d'interaction générée en comparant les fenêtres de texte de la requête et du document. Un réseau de neurones profond (e. g., CNN) appliqué sur la matrice d'interaction pour trouver des modèles d'appariement qui suggèrent la pertinence du document par rapport à la requête (Mitra and Craswell, 2018).

preuves partielles de pertinence. Au lieu de focaliser sur l'apprentissage complexe d'une représentation, une autre lignée de travaux vise plutôt à construire d'abord des appariements mot-à-mot entre la requête et le document et utilise ensuite des réseaux de neurones profonds pour apprendre des interactions hiérarchiques entre ces textes. En particulier, lorsqu'il s'agit de documents longs - qui peuvent contenir un ensemble de plusieurs sujets - une telle stratégie peut être plus efficace que d'essayer de représenter le document complet sous la forme d'un seul vecteur à faible dimension.

Généralement, dans cette approche basée sur les interactions, une fenêtre glissante est déplacée à la fois sur la requête et sur le texte du document. Les textes peuvent être représentés de différentes manières, y compris des vecteurs de termes classiques, des vecteurs de représentations pré-entraînées, ou des représentations qui sont mises à jour pendant l'entraînement du modèle. Chaque instance de la fenêtre sur la requête est comparée (ou "interagit") avec chaque instance de la fenêtre sur le texte du document (Figure 3.20). Par exemple, l'interaction peut être la cooccurrence entre deux termes dans la collection ou le produit scalaire entre deux vecteurs de représentations de mots. Un réseau de neurones (typiquement convolutif) est appliqué sur la matrice d'interactions et agrège les évidences dans toutes les paires de fenêtres comparées.

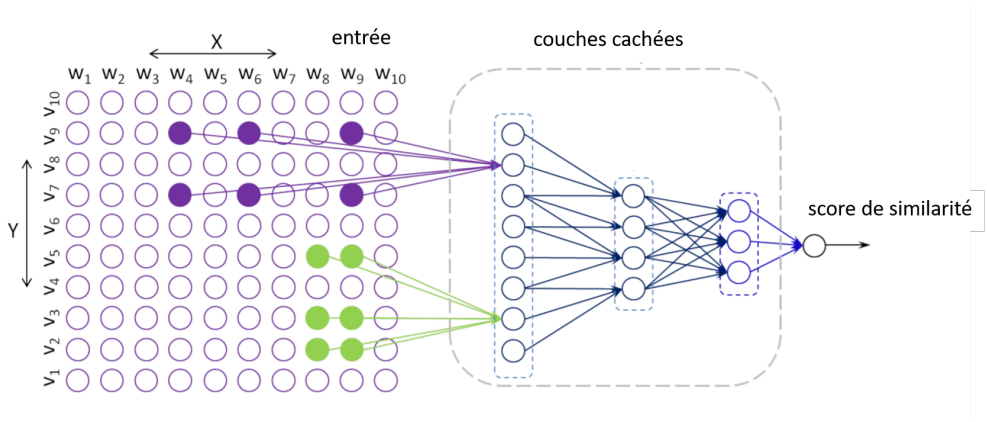


Figure 3.21 – Illustration de l'architecture profonde pour l'appariement DeepMatch (Lu and Li, 2013).

Par exemple, le modèle DeepMatch (Lu and Li, 2013) applique un réseau profond de type perceptron multicouches sur la matrice d'interactions (cooccurrences) des mots (Figure 3.21). Leur modèle est conduit par deux caractéristiques principales : (1) la localité et (2) la hiérarchie. La localité se traduit par le fait qu'il existe une structure locale importante dans l'espace sémantique des textes à apparier, qui peut être capturée par le modèle de cooccurrence des mots à travers les textes. Autrement dit, l'architecture à ce niveau ne modélise que des relations des paires "locales" à bas niveau, en laissant la comparaison des interactions entre des termes sémantiquement reliés aux niveaux supérieurs dans la hiérarchie. L'idée de la hiérarchie est que la prise de décision pour l'appariement a différents niveaux d'abstraction. Les décisions locales, qui capturent l'interaction entre les mots sémantiquement proches, seront combinées couche par couche, via le réseau de neurones, pour apprendre la décision finale et globale sur l'appariement. Pour entraîner le modèle, les auteurs utilisent des triplets de textes (x, y^+, y^-) dont le texte x est plus similaire à y^+ qu'à y^-. L'objectif d'apprentissage est de maximiser la similarité s (obtenue par le réseau) de la paire (x, y^+) et minimiser la similarité de la paire (x, y^-). Pour ce faire, les auteurs utilisent la fonction de coût de type *hinge-loss* suivante :

$$J(x, y^+, y^-) = max(O, 0, 1 + s(x, y^-) - s(x, y^+)) \tag{3.75}$$

Le modèle est entraîné sur des corpus de question-réponse et les résultats rapportés ont montré la performance supérieure par rapport aux modèles de références en termes de nDCG.

En adoptant la même structure générale, le travail de Hu et al. (2014) propose le modèle ARC-II dont l'architecture est illustrée dans la Figure 3.22. Ce dernier diffère du modèle DeepMatch en plusieurs points. La matrice des interactions n'est

plus des cooccurrences des mots mais il s'agit d'une matrice de multiplication entre deux matrices de représentations de la requête et du document. L'intuition est de capturer les interactions entre les mots au niveau de l'espace latent plutôt que les cooccurrences dans le corpus. Plus spécifiquement, chaque texte est

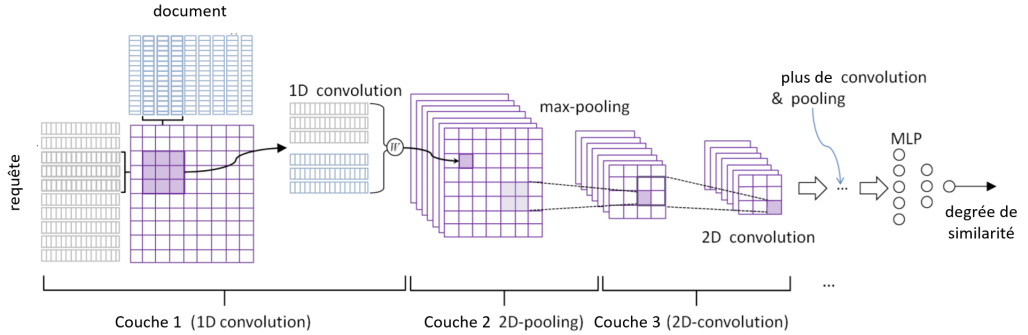


Figure 3.22 – Architecture du modèle ARC-II (Hu et al., 2014).

présenté par une séquence de représentations des mots (*embeddings*) composants, qui devient une matrice de taille $d \times m$, où d est la taille d'une représentation de mot (taille de l'espace latent) et m est la taille du texte (nombre de mots dans le texte). La matrice d'interaction est calculée par un opérateur matriciel (e. g., multiplication) entre les deux matrices de représentations des textes. Ensuite, pour apprendre les modèles d'appariement à partir de cette matrice, le réseau de neurones profond appliqué au dessus est un réseau à convolution avec des *max-pooling*, qui est capable de capturer et préserver l'ordre des caractéristiques locales dans la matrice d'interactions. Avec cette capacité de préserver l'ordre et la généralité du modèle, les auteurs suggèrent que ARC-II peut capturer la composition successive à l'intérieur de chaque phrase ainsi que l'extraction et la fusion des motifs d'appariement entre des phrases. Cette intuition est vérifiée par la performance supérieure de l'ARC-II lors d'expériences sur différentes tâches (complétion de phrase, question-réponses, identification de paraphrase).

Le modèle DRMM (*Deep Relevance Matching Model*) (Guo et al., 2016) est l'un des premiers modèles neuronaux à montrer une amélioration par rapport aux modèles de RI traditionnels sur des collections TREC et en considérant le texte intégral. Dans ce travail, les auteurs suggèrent et montrent que les méthodes d'apprentissage profond développées et appliquées en TALN pour un appariement sémantique ne seraient pas bien adaptées à la recherche ad-hoc, qui concerne surtout l'appariement de pertinence. Ils décrivent trois différences essentielles entre l'appariement sémantique et l'appariement de pertinence :

— L'appariement sémantique recherche la similarité sémantique entre les termes ; l'appariement de pertinence met davantage l'accent sur l'appariement exact.
— L'appariement sémantique s'intéresse souvent à la façon dont la composition et la grammaire aident à déterminer le sens ; dans l'appariement de pertinence, l'importance des termes de la requête est plus crucial que la grammaire car les requêtes sont généralement courtes et basées sur des mots-clés sans structures grammaticales complexes en RI ad-hoc.
— L'appariement sémantique compare deux textes entiers dans leur intégralité ; l'appariement de pertinence pourrait ne comparer que des parties d'un document à une requête.

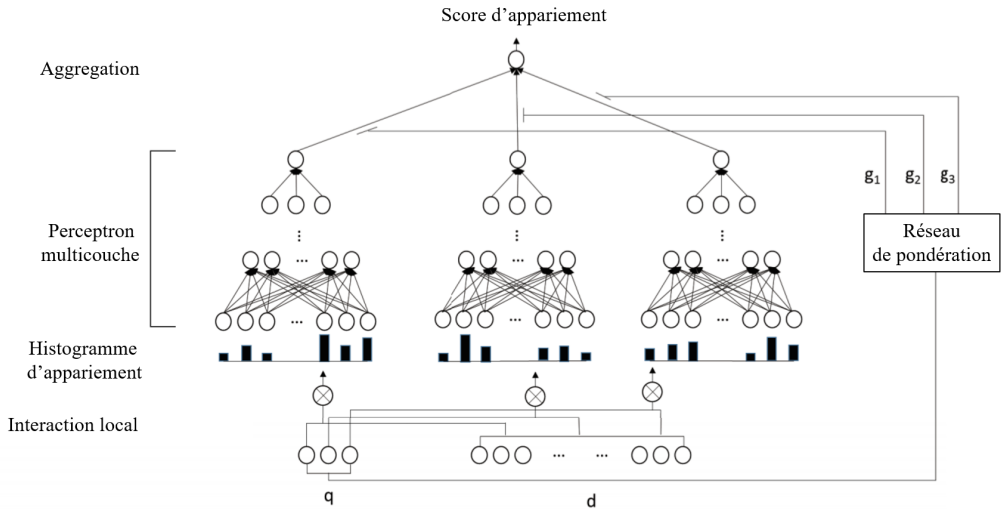


Figure 3.23 – Architecture du modèle DRMM (Guo et al., 2016).

Basés sur ces intuitions, les auteurs proposent un modèle neuronal profond basé sur les interactions pour la RI ad-hoc (avec le texte intégral). L'architecture du modèle est illustrée dans la Figure 3.23. L'entrée du réseau consiste en des histogrammes d'interactions qui représentent des niveaux de similarité cosinus entre les termes de la requête q avec chaque terme du document d. Un histogramme d'appariement regroupe les interactions locales en fonction de leurs niveaux de force des signaux plutôt que de leur position. Plus précisément, puisque l'interaction locale (i. e., la similarité du cosinus entre deux vecteurs de termes) se trouve dans l'intervalle $[-1\ ;\ 1]$, les auteurs discrétisent l'intervalle en un ensemble de sous-intervalles ordonnés et accumulent le nombre d'interactions locales dans chaque intervalle. Par exemple, supposons que la taille des intervalles soit fixée à $0,5$, on obtient cinq intervalles $\{[-1\ ;\ -0,5), [-0,5\ ;\ 0), [0\ ;\ 0,5), [0,5\ ;\ 1), [1\ ;\ 1]\}$. Étant donné un terme de requête "*voiture*", un document *(voiture, location, camion, bosse, injonction, piste)* et les interactions locales correspondantes basées sur la simi-

larité du cosinus sont $(1\ ;\ 0,2\ ;\ 0,7\ ;\ 0,3\ ;\ -0,1\ ;\ 0,1)$, on obtient un histogramme d'appariement de $[0\ ;\ 1\ ;\ 3\ ;\ 1\ ;\ 1]$. Le nombre d'histogrammes en entrée est égal au nombre de termes de la requête. Au-dessus de cette couche d'histogrammes se trouve un réseau de neurones (perceptron multicouche) qui sort un nœud unique pour chaque terme de requête. Ces sorties sont combinées avec la pondération de l'importance (des termes de la requête) calculée par un réseau de pondération (*gating network*). Puis le score pondéré final $s(q,d)$ est considéré comme le degré de pertinence de la paire requête/document. L'ensemble du réseau est entraîné à minimiser la fonction de coût de type *hinge-loss*, qui s'adapte le mieux au problème d'ordonnancement. Etant donné un triplet (q, d^+, d^-) où le document d^+ est a un rang plus élevé que le document d^- par rapport à la requête q, la fonction de coût vise à maximiser l'écart entre les scores des deux paires (q, d^+) et (q, d^-) :

$$J(q, d^+, d^-) = max(0, 1 - s(q, d^+) + s(q, d^-)) \tag{3.76}$$

Des expérimentations de recherche ad-hoc sont menées sur les corpus TREC Robusto4 et ClueWebo9, en comparant avec les modèles de référence basés sur la présentation (DSSM, C-DSSM, ARC-I) ainsi que les modèles basés sur les interactions (ARC-II, DeepMatch). Les résultats montrent que le DRMM surpasse significativement tous les modèles de référence, y compris les modèles de RI traditionnels (BM25, Modèle de langue).

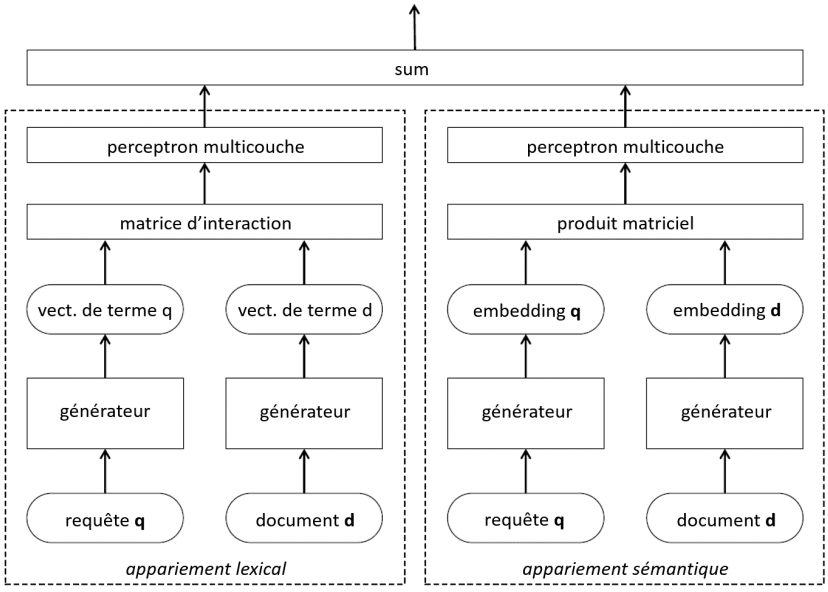


Figure 3.24 – Architecture du modèle Duet (Mitra et al., 2017).

Les auteurs dans Mitra et al. (2017) mettent l'accent sur l'importance de l'appariement lexical dans les modèles neuronaux profonds pour la RI. Ils ont montré

que les modèles basés sur la représentation ont tendance à ne pas être performants lorsqu'ils font face à des termes rares. Ils ont aussi argumenté que la recherche sur le Web nécessite à la fois l'appariement exact et inexact. Par exemple, pour la requête "entreprises salesforce", il est plus facile d'estimer la pertinence avec l'appariement exact du terme rare "salesforce". Un modèle neuronal qui fait l'appariement dans l'espace latent est peu susceptible d'avoir une bonne représentation pour ce terme rare. En revanche, pour la requête "quelle chaîne diffuse le match OM-PSG aujourd'hui", le document cible contient probablement "Bein Sport" ou "Sky Sports", pas le terme "chaîne". Basé sur cette motivation, Mitra et al. (2017) ont proposé un modèle de RI neuronal, appelé Duet architecture (Figure 3.24), qui incorpore à la fois des appariements lexicaux et sémantiques. Leur modèle consiste en deux parties parallèles : un réseau pour l'appariement exact qui cherche à apprendre les modèles interactions entre deux textes ; et un réseau pour l'appariement sémantique, basé sur la représentation latente des textes. Le réseau est entraîné avec une fonction de coût de type *hinge-loss* comme dans Lu and Li (2013) (ref. Equation 3.75). Les auteurs utilisent des grandes données étiquetées venues du moteur de recherche Bing pour entraîner le modèle Duet. Le modèle neuronal qui se concentre sur l'appariement lexical a généralement moins de paramètres et peut être entraîné sous de petits régimes de données - contrairement à celui d'appariement sémantique qui se concentre sur l'apprentissage des représentations du texte, qui nécessite beaucoup de données et de temps à entraîner.

Résumé

Nous avons présenté dans ce chapitre les concepts de base du réseau de neurones ainsi que les approches neuronales pour l'apprentissage de représentations de texte et leurs applications dans la RI. Les représentations distribuées de texte ont montré l'efficacité supérieure pour les tâches de TALN ainsi que celles de RI. Nous avons aussi abordé les modèles d'appariement par les réseaux de neurones profonds qui ont donné lieu aux progrès remarquables dans la RI.

Dans cette thèse, nous nous intéressons particulièrement à la combinaison de la sémantique relationnelle issue des ressources de connaissance avec la sémantique distributionnelle apprise par les modèles neuronaux. Dans cette optique, nous présentons dans ce manuscrit deux contributions principales :

— Un modèle d'appariement utilisant un réseau de neurones profond. Ce modèle exploite les représentations sémantiques latentes des documents et des requêtes en bénéficiant des concepts et des relations exprimés au sein d'une ressource externe.

— Deux méthodes d'apprentissage de représentations de texte qui ont pour but de réduire le fossé sémantique en RI en combinant les deux sources d'évidence (i. e., la sémantique relationnelle et la sémantique distributionnelle). Le premier modèle vise à corriger les vecteurs de représentations du document a posteriori en prenant compte des contraintes relationnelles. Le second modèle exploite des connaissances explicites pour régulariser conjointement l'apprentissage de représentations des mots, des concepts et des documents.

Partie II

PROPOSITIONS DES MODÈLES NEURONAUX EN RI

4

APPRENTISSAGE DES REPRÉSENTATIONS DU TEXTE

Introduction

Nous avons présenté dans le chapitre 3 un état-de-l'art sur les méthodes d'apprentissage des représentations distribuées de textes. Ces modèles exploitent la sémantique distributionnelle du texte via des architectures neuronales pour apprendre des représentations du texte (Bengio et al., 2003). Nous présentons dans ce chapitre nos contributions dans l'apprentissage des représentations distribuées de textes en combinant avec la sémantique relationnelle issue des ressources. Nos contributions se distinguent principalement des approches de l'état-de-l'art par la prise en compte de différents niveaux de ganularité dans le texte (mot, concept et document) dans l'étape d'apprentissage de représentations. Notre objectif est de réduire le fossé sémantique entre deux textes en exploitant la sémantique relationnelle issue des ressources terminologiques et l'introduire dans le processus d'apprentissage de représentations par un réseau de neurones. Pour cela, nous proposons deux méthodes, une de type "hors ligne" et une de type "en ligne"

Notre approche "hors ligne" consiste à apprendre une représentation de documents dite "optimale" qui combine deux types de représentations pré-entraînées. Pour cela, nous entraînons dans un premier temps deux modèles de représentations distribuées de documents, un basé sur le texte brut, un autre basé sur les concepts du document. Dans un second temps, nous modélisons une méthode d'apprentissage afin de rapprocher ces deux espaces de représentation préliminaires, à savoir l'espace de mots et l'espace de concepts, pour obtenir une représentation finale de document, qui combine ces deux sémantiques latentes.

Notre deuxième proposition, dite "en ligne", consiste en un apprentissage conjoint des représentations des textes à plusieurs niveaux (i. e., document, mot, concept). Nous l'appelons le modèle "tripartite" qui exploite synergie de l'apprentissage conjoint de mots, concepts et documents. Notre intuition est que l'apprentissage simultané de représentations dans un contexte à plusieurs niveaux permet d'affiner les représentations de mots pour mieux résoudre le problème de poly-

sémie, améliorant ainsi les représentations de documents apprises dans le même processus.

Nous proposons également l'intégration de la contrainte relationnelles entre les mots et/ou les concepts. Pour ce faire, nous modélisons deux approches pour l'intégration de la sémantique relationnelle dans l'apprentissage de représentations distribuées : 1) une basée sur la régularisation de la fonction objectif ; 2) une autre basée sur les instances d'apprentissage en entrée. Nous intégrons chaque approche dans chacun des deux modèles d'apprentissage "hors ligne" et "en ligne" pour évaluer l'apport de la sémantique relationnelle.

Nous menons également plusieurs expérimentations pour analyser et évaluer plusieurs aspects, à savoir la qualité des représentions entraînées, l'apport de la contrainte relationnelle, et l'efficacité en RI avec nos représentations de textes à plusieurs niveaux. Plus précisément, nous analysons d'abord la qualité de nos représentations sur les tâches similarité des phrases et similarité des documents. Ensuite, nous examinons l'apport de nos représentations en les exploitant dans des tâches de RI, à savoir l'expansion de la requête et le réordonnancement des documents. A travers toutes ces analyses, nous évaluons également l'apport des contraintes relationnelles en comparant avec les autres configurations ainsi qu'avec des modèles de référence.

Ce chapitre est organisé comme suit. La section 1 présente les problématiques et les principales motivations de nos contributions. La section 3 décrit notre premier modèle neuronal pour l'apprentissage "hors ligne" des représentations conceptuelles de documents. Notre deuxième contribution sur l'apprentissage conjoint ("en ligne") des représentations de textes est présentée dans la section 4. Les deux types d'intégration des connaissances relationnelles issues des ressources sémantiques sont détaillés dans la section 5. Des détails du protocole d'évaluation de nos contributions sont présentés dans la section 6. La section 7 présente les résultats expérimentaux ainsi que les discussions sur les résultats obtenus. Nous concluons ce chapitre dans la section 8.

1 Contexte et motivations

Du point de vue de la RI, l'accès à l'information implique la sélection d'informations pertinentes dans de grands corpus de documents. Ces informations sont sélectionnées par 1) l'appariement des requêtes de l'utilisateur, qui sont reformulées par un ensemble de mots-clés, avec des documents et 2) l'appariement des documents entre eux. Cependant, de nombreuses études ont montré qu'un tel appariement est généralement difficile, principalement en raison du *fossé sémantique*.

Ce dernier désigne la différence des représentations de bas niveau des documents et l'interprétation de haut niveau de leurs contenus telle que perçue par l'humain. Ce fossé est une des principales raisons du défaut d'appariement entre requête et document qui conduit à la dégradation des performances d'un système de RI (Crestani, 2000). Le fossé sémantique provient généralement de : 1) la *discordance de vocabulaire*, ce qui signifie que des mots de formes différentes partagent le même sens (e. g., **aperçu** est synonyme de **sommaire**) ; 2) la *discordance de granularité*, ce qui signifie que des mots de formes et de sens différents appartiennent au même concept général (e. g., **chat** et **chien** sont des **animaux**) ; 3) la *polysémie*, ce qui signifie qu'un mot peut couvrir différents sens en fonction des mots qui l'entourent dans le texte et qui représentent son contexte (e. g., **pêche** peut signifier un fruit ou l'action de pêcher).

De très nombreux travaux en RI ont œuvré dans le sens de la réduction du fossé sémantique à l'aide de la sémantique relationnelle et la sémantique distributionnelle. Une première lignée de travaux est basée sur l'exploitation des indices sémantiques explicites dérivés de ressources sémantiques (e. g., WordNet, UMLS) ou graphes de connaissances (e. g., DBpedia ou Freebase). L'idée sous-jacente à ces approches est d'injecter la connaissance portée par les concepts/entités et relations entre entités/concepts pour améliorer la représentation de la requête ou du document (Corcoglioniti et al., 2016a). Nous avons présenté dans le chapitre 2 un état-de-l'art sur ces approches, qui peuvent être classifiées en deux grandes catégories : l'expansion de la requête (Xiong and Callan, 2015b; Pal et al., 2014; Navigli and Velardi, 2003) et l'expansion de document (Agirre et al., 2010; Gobeill et al., 2008; L'Hadj et al., 2016; Gupta et al., 2017).

Une autre lignée de travaux, présentée dans le chapitre 3, exploite quant à elle des indices sémantiques implicites dérivés des corpus, à savoir la sémantique distributionnelle (Harris, 1954). Cette dernière est fondée sur le calcul de la proximité sémantique entre les mots sur la base des contextes partagés dans les corpus de textes. Ces modèles sont capables d'inférer les sens des mots par association à d'autres mots en analysant leurs co-occurrences dans le corpus de documents, comme l'apprentissage de la représentation distribuée des mots (Mikolov et al., 2013b; Pennington et al., 2014). Ces représentations vectorielles de dimension réduite sont alors utilisées dans la définition de nouveaux schémas d'appariement requête-document (Zuccon et al., 2015) ou alors pour une expansion de requêtes (Zamani and Croft, 2016b).

Cependant des travaux ont montré des limites des représentations distribuées de mots classiques : 1) elles ne sont pas capables de résoudre le problème de polysémie (Iacobacci et al., 2015) ; 2) l'apprentissage de ces représentations basées seulement sur le texte des corpus ne permet pas de capturer la sémantique relationnelle définie explicitement dans des ressources sémantiques ; 3) les vecteurs de repré-

sentation distribuée de mots peuvent s'avérer peu lisibles en ce sens qu'ils ne sont pas alignables avec des ressources externes. Afin d'améliorer la sémantique des représentations distribuées, plusieurs travaux utilisent des ressources sémantiques pour injecter la sémantique relationnelle dans l'apprentissage des représentations (Iacobacci et al., 2015; Yamada et al., 2016; Liu et al., 2016). L'idée sous-jacente à ces approches est d'injecter la connaissance portée par les concepts/entités et leur relations pour résoudre le problème de polysémie ou/et modifier les représentations pour capturer les relations explicites entre les mots/concepts. En effet, l'apprentissage de représentations avec la régularisation par la ressource sémantique améliore la qualité des représentations en combinant la *sémantique distributionnelle* dans le corpus et la *sémantique relationnelle* issue de la ressource sémantique. Pour rappel, nous avons présenté dans le chapitre 3 deux catégories de modèles selon l'étape d'incorporation de la connaissance externe dans les représentations de mots.
La première catégorie "en ligne" exploite la connaissance issue des ressources externes pendant la phase d'apprentissage des représentations distribuées des mots (Yu and Dredze, 2014; Iacobacci et al., 2015; Mancini et al., 2017; Nguyen et al., 2017d). Ces modèles modifient l'objectif original d'apprentissage distributionnel en y intégrant les contraintes issues des ressources externes. Par exemple, dans Cheng et al. (2015), les auteurs proposent d'étendre le modèle Skip-gram (Mikolov et al., 2013a) en identifiant les paires mot-concept vus comme des paires de mot-sens candidat dans un contexte donné en effectuant l'entraînement conjoint de leurs représentations latentes. Les alignements mot-concept sont établis soit avec des concepts explicites issus de ressources externes ou alors avec des concepts implicites dérivés du corpus. Dans la même perspective de résolution de la polysémie, Mancini et al. (Mancini et al., 2017) étendent le modèle CBOW pour apprendre des représentations distinctes des différents sens d'un mot. Pour cela, une architecture révisée du modèle CBOW est proposée en vue d'apprendre conjointement dans le même espace à la fois le mot et les différents sens candidats associés.
La deuxième catégorie se diffère par l'intégration des connaissances issues des ressources hors de l'apprentissage distributionnel (Faruqui et al., 2015; Mrkšić et al., 2016; Vulić et al., 2018; Vulić and Mrkšić, 2018). Cette approche de correction a posteriori (*hors ligne*) affine des représentations distribuées obtenues de n'importe quel modèle pour satisfaire aux contraintes des ressources externes. Par exemple, Faruqui et al. (2015) proposent une méthode pour affiner les représentations dans l'espace vectoriel à l'aide de la relation synonyme issue de la base lexicale WordNet en incitant les mots connectés (synonymes) à avoir des représentations vectorielles similaires. Cette méthode ne fait aucune hypothèse sur la façon dont les représentations d'entrée ont été construites.

Notre contribution s'inscrit dans la lignée de ces travaux dont l'objectif est d'améliorer la représentations de documents en vue de réduire le fossé séman-

tique, en nous intéressant spécifiquement à les intégrer dans un modèle de RI. De ce fait, nous nous intéressons à l'apprentissage de représentations de documents qui sont cruciaux en RI. Nos propositions se distinguent des approches dans l'état-de-l'art par la prise en compte des textes à plusieurs ganularités (i. e., le mot, le concept et le document) dans le but d'obtenir une meilleure représentation de documents. Nous abordons particulièrement les quatre questions de recherche suivantes :

• **RQ1** : Comment ajuster les représentations distribuées pré-entraînées des documents avec les connaissances de la ressource sémantique externe ?

• **RQ2** : Comment modéliser un apprentissage conjoint de document/mot/concept pour avoir de meilleures représentations distribuées ?

• **RQ3** : Comment intégrer les contraintes relationnelles dans ces processus d'apprentissage de représentations ?

• **RQ4** : Quelles sont les impacts de nos méthodes d'apprentissage sur les tâches de TALN et de RI ?

Pour répondre à ces questions de recherche, nous nous basons sur les deux hypothèses suivantes :

• *La prise en compte d'un contexte multi-niveaux (H1)* : chaque mot peut être assigné à un sens unique, c'est-à-dire à un concept pertinent identifié dans une ressource sémantique, au sein d'un même document alors qu'il pourrait relever de sens différents et être polysémique s'il est analysé sur un ensemble de documents. Ainsi, nous supposons que l'apprentissage simultané de représentations dans un contexte à plusieurs niveaux (à savoir, un niveau global pour des contextes de documents et un niveau local pour des contextes de mots et de concepts) permet d'affiner les représentations pour mieux résoudre le problème de polysémie.

• *La prise en compte d'un contexte basé sur les ressources sémantiques (H2)* : nous supposons que le problème lié à la discordance de granularité conceptuelle peut être partiellement ou complètement résolu en considérant la connaissance établie dans une ressource sémantique externe portant sur les relations entre mots. Selon le même principe que l'hypothèse distributionnelle, notre hypothèse repose sur le fait que des mots reliés au même concept devraient avoir des représentations latentes proches.

Ainsi nous proposons d'abord une méthode de correction de représentations de documents et intégrant la sémantique issue d'une base de connaissances. Ensuite, nous présentons un modèle tripartite qui permet d'apprendre les représentations de documents en dérivant conjointement les représentations de mots et concepts

et en considérant la contrainte de relations établies dans une ressource sémantique externe. Plus précisément, nous présentons dans ce chapitre les contributions suivantes :

• Le modèle hors ligne $SD2V_{off}$ (Nguyen et al., 2017c) apprend une représentation de documents dite "optimale" qui combine deux types de représentations pré-entraînées. L'intégration de la connaissance conceptuelle des documents se dit "hors-ligne" car elle s'est faite dans une étape hors de l'apprentissage de représentations distribuées.

• Le modèle en ligne neuronal tripartite $SD2V_{on}$ (Nguyen et al., 2018b,a) apprend conjointement les représentations de documents et les représentations de mots et concepts. Ce modèle s'inscrit dans la lignée des méthodes d'apprentissage "en ligne" car les représentations de différents niveaux de texte (document, mot, concept) sont construites pendant l'étape d'apprentissage de représentations distribuées.

• Deux approches pour intégrer des connaissances de relation issues des ressources sémantiques externes, une basée sur la régularisation de la fonction objectif et une autre basée sur les instances d'apprentissage en entrée. Chaque approche est intégrée dans les deux modèles d'apprentissage "hors ligne" et "en ligne" pour évaluer l'apport de la sémantique relationnelle.

• Une évaluation expérimentale comparative utilisant des jeux de données génériques ainsi que spécifique au domaine (médecine). Nous analysons d'abord la qualité de nos représentations sur les tâches similarité des phrases et similarité des documents. Ensuite, nous examinons l'apport de nos représentations en les appliquant sur les tâches de RI, à savoir l'expansion de la requête et le réordonnancement. A travers toutes ces analyses, nous évaluons aussi l'apport des contraintes relationnelles en comparant avec les autres configurations ainsi qu'avec les modèles de référence.

2 Notation

Nous introduisons dans cette section les notations utilisés pour nos deux modèles. Plus particulièrement, nous définissons une collection des documents avec l'annotation des concepts associés aux mots dans le texte. Ces concepts sont issus d'une ressource sémantique externe, qui fournit également des relations entre les mots et les concepts. Nous détaillerons la notion de relation dans la section 5.1.

Formellement, un document d dans la collection D est modélisé sous la forme d'une séquence de mots ordonnés w_i. Chaque document d est annoté avec les concepts c_j issus d'une ressource sémantique $\mathcal{R}$. A partir de cette annotation, nous créons un document conceptuel $\mathcal{C}_d$ qui ne contient que les concepts annotés, en gardant leur ordre d'apparition dans le texte. Pour mieux comparer, nous distinguons deux types de contenus pour un document d :

— $\mathcal{W}_d$, document représenté par une séquence de mots w_i (contenu original)
— $\mathcal{C}_d$, document représenté par une séquence de concepts c_j (document conceptuel)

Avec ces deux types de documents, la collection D a un vocabulaire de mot V et un vocabulaire de concepts C qui contiennent tous les mots w (et tous les concepts c, respectivement) distincts dans tous les documents.

3 Apprentissage hors ligne de représentations de documents basée sur deux espaces latents

Nous présentons dans cette section notre contribution qui consiste en une méthode d'amélioration des représentations distribuées des documents. Pour répondre à la question de recherche **RQ1**, nous proposons d'intégrer les représentations conceptuelles de documents dans les représentations distribuées basées sur le texte original. Notre modèle est guidé par les intuitions suivantes : 1) la prise en compte des concepts dans le processus d'apprentissage de représentations des documents, en plus des mots, devrait permettre de construire des représentations des documents sémantiques qui remédient aux limites du fossé sémantique ; 2) la représentation optimale d'un même document dans un espace latent de faible dimension nécessite la proximité de la représentation issue de façon indépendante des sources d'évidence basées sur la ressource et celles basées sur les corpus de documents. De ce fait, nous fixons les deux objectifs suivants :

• **(O1)** : Construire un espace de représentation conceptuelle des documents, qui est capable de capturer la sémantique exprimée dans une ressource externe.
• **(O2)** : Construire une représentation optimale des documents qui permet de rapprocher deux espaces disjoints de représentations distribuées, un espace conceptuel créé dans (O1) et un espace de représentation basé sur le texte brut.

Etant donné un document, l'objectif final de notre approche est d'améliorer sa représentation pour les tâches de RI en combinant les connaissances sémantiques et les informations textuelles brutes. Dans ce modèle, nous proposons de combiner

les informations textuelles et les connaissances conceptuelles pour un apprentissage de représentations combinées des documents. Cette combinaison est effectuée dans une étape d'apprentissage hors ligne qui rapproche deux espaces latents, obtenus par deux modèles d'apprentissage de représentations, un basé sur les mots, un basé sur les concepts identifiés dans le document. L'apprentissage est dit "hors ligne" parce qu'il est effectué hors du processus d'apprentissage de représentations distribuées. La nouvelle représentation est entraînée pourqu'elle garde au maximum la sémantique contenue dans chaque espace latent.

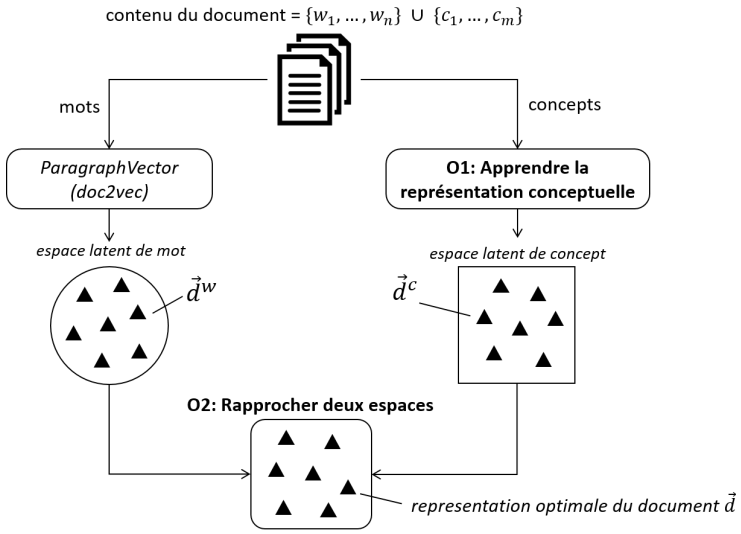


Figure 4.1 – Intuition du modèle hors ligne $SD2V_{off}$

L'intuition de notre approche est illustrée dans la Figure 4.1. Pour un document d, dans un premier temps, nous exploitons le modèle *Paragraph Vector* (PV-DM) (Le and Mikolov, 2014) pour obtenir un vecteur $\vec{d^w}$ dans l'espace latent de mot (branche gauche de la figure). Dans un second temps, nous réalisons notre premier objectif qui consiste à construire la représentation conceptuelle $\vec{d^c}$ du document dans l'espace des concepts (branche droite de la figure). Pour cela, nous proposons le modèle *cd2vec (conceptualDoc2vec)* (Section 3.2) qui apprend une représentation distribuée en utilisant les concepts identifiés dans le document. Enfin, notre second objectif est effectué par l'apprentissage d'un vecteur latent $\vec{d}$ permettant de rapprocher les représentations basées sur les concepts $\vec{d^c}$ et celles sur le texte brut $\vec{d^w}$.

Nous présentons dans ce qui suit les détails de ces étapes : (1) apprendre la représentation distribuée basée sur le texte des documents ; (2) apprendre la représentation conceptuelle des document ; (3) obtenir la représentation *optimale* qui rapproche les deux espaces latents disjoints (conceptuelle et textuelle).

3.1 *Apprentissage de représentations basées sur le texte des documents*

Pour obtenir les représentations distribuées basées sur le texte des documents, nous exploitons le modèle *Paragraph Vector* (Le and Mikolov, 2014) (cf. Chapitre 3, Section 2.1.2). Plus précisément, nous utilisons la version *Distributed Memory* (PV-DM). Pour rappel, le modèle PV-DM est entraîné pour prédire un mot en tenant compte du contexte d'entrée qui se compose par les mots voisins et le document qui les contient (cf. Figure 4.2).

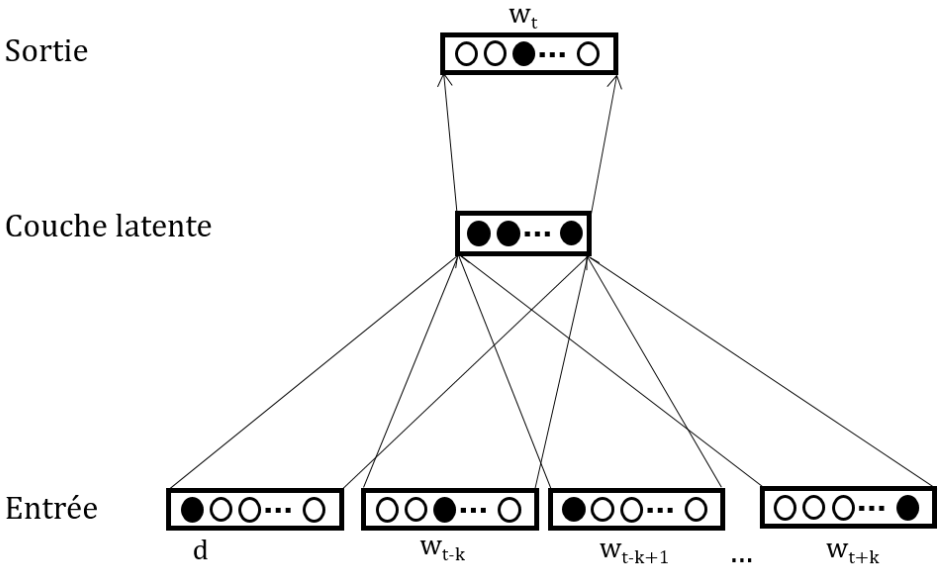


Figure 4.2 – Architecture du modèle PV-DM (Le and Mikolov, 2014).

Etant donné un document d, pour un mot $w_t \in d$ et ses mots voisins $w_{t\pm k}$ dans la fenêtre k, le modèle PV-DM apprend les représentations de mots et de documents en maximisant la probabilité d'obtenir le mot w_t sachant le contexte compris des mots voisins $w_{t\pm k}$ et le document d. La fonction objectif pour un instance de document d est calculée comme suit :

$$J_{PV-DM} = \sum_{w_t \in d} \log P(w_t | w_{t\pm k}, d) = \frac{\exp(\vec{w}_t{}^{\top} \cdot \overline{h}_{w_t})}{\sum_{w' \in V} \vec{w'}{}^{\top} \cdot \overline{h}_{w'})} \tag{4.1}$$

où V est le vocabulaire de la collection ; $\vec{x}$ est le vecteur de représentation distribuée de l'objet x (mot/document) ; $\overline{h}_{w_t}$ est le vecteur de la couche cachée qui est combiné des vecteurs des objets dans le contexte du mots w_t, à savoir les mots voisins $w_{t\pm k}$ et le document d.

Ai et al. (2016b) ont montré que les modèles *Paragraph Vector* a un problème de sur-apprentissage au cours de l'entraînement, et ce problème est plus grave pour les documents courts. Dans notre modèle, nous suivons leur approche pour résoudre les problèmes de sur-apprentissage en utilisant une régularisation dans l'objectif d'apprentissage de PV-DM. Ils ont sugéré que le problème de sur-apprentissage est principalement causé par les vecteurs de document sans restriction, nous ajoutons une L2-régularisation sur les vecteurs de document. Ainsi la fonction objectif du modèle PV-DM est modifiée comme suit :

$$J_{PV-DM} = \sum_{w_t \in d} \log P(w_t | w_{t\pm k}, d) - \frac{\gamma}{|d|} ||\vec{d}||^2 \tag{4.2}$$

où $|d|$ est la longueur du document d (nombre de mots dans le document) ; $||\vec{d}||$ dénote la norme du vecteur $\vec{d}$ et γ est l'hyperparamètre qui contrôle la force de régularisation.

Une fois appliqué le PV-DM modifié sur les documents pour obtenir les représentations distribuées basées sur le texte $\vec{d}^w$, nous résolvons notre objectif (O1) en proposant un modèle qui apprend les représentations conceptuelles des documents, appelé *conceptualDoc2vec*. Nous détaillons ce modèle dans la section qui suit.

3.2 *Apprentissage de représentations conceptuelles des documents*

Guidé par le modèle *Paragraph Vector* (PV-DM) (Le and Mikolov, 2014) qui apprend la représentation de documents à partir de leur texte brut, nous proposons le modèle *conceptualDoc2vec (cd2v)* qui produit la représentation sémantique distributionnelle des concepts sous-jacents au texte. Le modèle PV-DM repose sur l'intuition qu'un mot peut être prédit en fonction de son contexte et du paragraphe auquel il est associé, permettant ainsi d'apprendre conjointement la représentation des mots et du paragraphe.

De façon similaire, notre modèle d'apprentissage de représentations conceptuelles des documents *conceptualDoc2vec* repose sur un objectif de prédiction de concept à partir d'un contexte, permettant ainsi d'apprendre la représentation des concepts et du document $\vec{d}^c$. L'architecture de notre modèle incluant une illustration *conceptualDoc2vec* est illustrée dans la Figure 4.3. Dans ce modèle, nous obtenons d'abord les concepts dans le document en utilisant un outil d'annotation conceptuelle. Puis, un document conceptuel $\mathcal{C}_d$ est reproduit en utilisant les

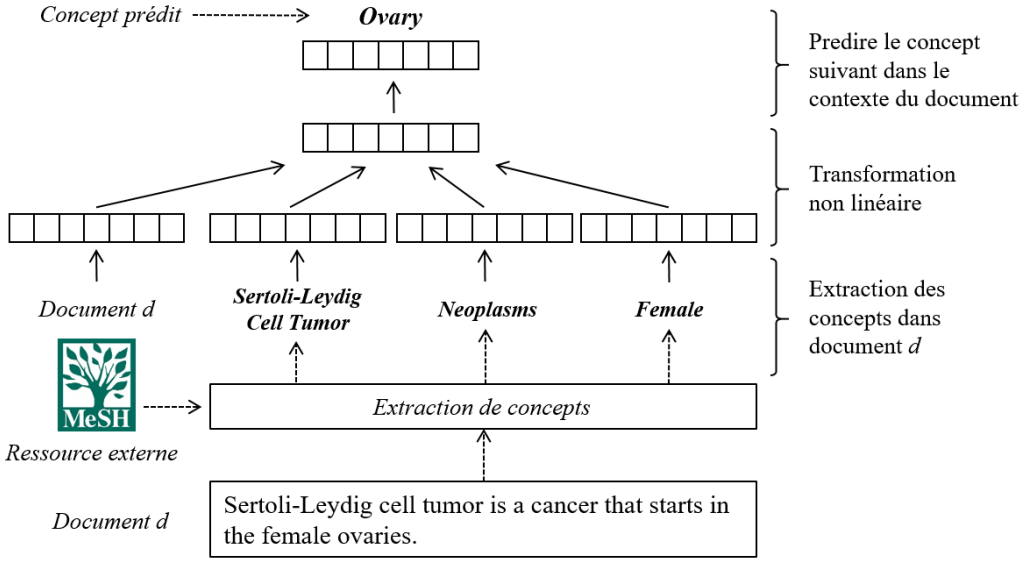


Figure 4.3 – Architecture du modèle *conceptualDoc2vec*

concepts identifiés, en gardant l'ordre d'apparition dans le texte. Puis, nous appliquons le modèle PV-DM sur ces documents conceptuels. Nous soulignons que dans ce travail, nous considérons seulement les associations mot-concept pour des mots simples (uni-grammes) et laissons les associations entre concepts et mots composés pour de futurs travaux.

Etant donné un document d, pour un concept $c_t \in d$ et ses concepts voisins $c_{t\pm k}$ dans la fenêtre k, le modèle *conceptualDoc2vec* a pour objectif de maximiser la probabilité d'obtenir le concept c_t sachant le contexte compris des concepts voisins $c_{t\pm k}$ et le document d. Nous appliquons aussi la régularisation par la longueur du document conceptuel $\mathcal{C}_d$. Ainsi, la fonction objectif pour une instance de document $\mathcal{C}_d$ est calculée comme suit :

$$J_{cd2v} = \sum_{c_t \in \mathcal{C}_d} \left[\log P(c_t \mid c_{t\pm k}, d) - \frac{\gamma}{|\mathcal{C}_d|} ||\vec{d}||^2 \right] \tag{4.3}$$

Similaire au modèle PV-DM, la probabilité $P(c_t \mid c_{t\pm k}, d)$ est définie par une fonction soft-max comme suit :

$$P(c_t \mid c_{t\pm k}, d) = \frac{\exp(\vec{c_t}^{\top} \cdot \overline{h}_{c_t})}{\sum_{c' \in C} \exp(\vec{c'}^{\top} \cdot \overline{h}_{c'})} \tag{4.4}$$

où $\vec{c_t}$ est la représentation du concept c_t ; $\overline{h}_{c_t}$ correspond à la moyenne des représentations des concepts $c_{t\pm k}$ dans la fenêtre de contexte du concept c_t, incluant le document d ; et C est l'ensemble de concepts dans la collection.

Étant donné la taille importante du vocabulaire V et l'ensemble des concepts C, les probabilités décrites dans les formules (4.1) et (4.3) sont difficiles à estimer. Guidé par des précédents travaux (Mikolov et al., 2013a), nous exploitons les stratégies d'échantillonnage négatif ("*negative sampling*") pour définir des fonctions objectif alternatives pour chaque élément $e_t \in \{w_t; c_t\}$:

$$p(e_t|w_{t\pm k}, c_{t\pm k}, d) = \log \sigma(\vec{e}_t'^{\top} \cdot \overline{h}_{e_t}) + \sum_{i=1}^{n} \mathbb{E}_{e_i \sim P_n(e)} \left[\log \sigma(-\vec{e}_i'^{\top} \cdot \overline{h}_{e_t}) \right] \tag{4.5}$$

où $\sigma(x)$ correspond à la fonction sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ et $\mathbb{E}_{e_i \sim P_n(e)}$ est la valeur attendue de $\log \sigma(-\vec{e}_i'^{\top} \cdot \overline{h}_{e_t})$ quand e_i est tiré de la distribution uniforme pondérée $P_n(e)$, comme réalisé par Ai et al. (2016b).

3.3 *Rapprocher deux espaces de représentations latentes*

Notre second objectif (O2) est d'optimiser la représentation du document d afin d'obtenir un vecteur latent $\vec{d}$ permettant de rapprocher les représentations basées sur les concepts $\vec{d}^c$ et celles sur le texte brut $\vec{d}^w$. Reprenons la Figure 4.1, notre objectif est de construire un troisième espace latent qui rappoche les deux espaces des représentations distribuées préapprises, une basée sur les mots, une basée sur les concepts. Notre idée est de calculer un nouveau vecteur de représentation $\vec{d}$ qui est à la fois proche des deux espaces des vecteurs $\vec{d}^w$ et $\vec{d}^c$. Ce problème peut être formulé par une optimisation qui vise à miniser la fonction objectif suivante :

$$\psi(D) = \sum_{d \in D} \psi(d) = \sum_{d \in D} \left[(1-\beta) \times \|\vec{d} - \vec{d}^c\|^2 + \beta \times \|\vec{d} - \vec{d}^w\|^2 \right] \tag{4.6}$$

où D est la collection de documents, $||x - y||$ la distance euclidienne entre les vecteurs de représentation x et y, et β correspond au coefficient de pondération, défini expérimentalement.

Nous utilisons la méthode de descente de gradient stochastique (SGD) pour résoudre le problème d'optimisation (Équation 4.6) qui infère la représentation optimale des documents $\vec{d}$ pour rapprocher deux espaces latents, un sur les mots et un sur les concepts. Plus particulièrement, cette méthode met à jour, pour chaque document d, sa représentation en utilisant la première dérivée $\Delta = \frac{\partial \psi(\vec{d})}{\partial \vec{d}}$ de la fonction ψ par rapport à $\vec{d}$ avec un pas de α, comme illustré dans Algorithme 1.

Algorithme 1 Apprentissage de la représentation de documents par SGD

Entrée : $\vec{d_i}^{w}, \vec{d_i}^{c}$
Sortie : $\vec{d}$
  $\vec{d} = randomVector()$
  $\psi(\vec{d}) = (1-\beta)\|d - \vec{d^c}\|^2 + \beta\|d - \vec{d^w}\|^2$
  tant que $\psi(d) > \epsilon$ **faire**
    $\Delta = 2 \times (1-\beta) \times (d - \vec{d^c}) + 2 \times \beta \times (\vec{d} - \vec{d^w})$
    $d = \vec{d} - \alpha \times \Delta$
  fin tant que
  retourner $\vec{d}$

4 Apprentissage en ligne tripartite pour la représentation de documents

Dans la section précédente, nous avons proposé une méthode d'apprentissage des représentations combinées de deux espaces sémantiques disjointes, un basé sur le texte brut et un basé sur les concepts du document. Un inconvénient de cette méthode est qu'elle ne permet pas d'obtenir les représentations de mots et de concepts dans le même espace latent final des documents. Pour répondre à la question de recherche **RQ2**, nous proposons ici un modèle d'apprentissage conjoint de représentations de document, mot et concept. Nous rappelons notre hypothèse (H1), selon laquelle l'apprentissage simultané de représentations dans un contexte à plusieurs niveaux (à savoir, un niveau global pour des contextes de documents et un niveau local pour des contextes de mots et de concepts) permet d'affiner les représentations pour mieux résoudre le problème de polysémie. Inspiré par cette hypothèse et le modèle *ParagraphVector* (Le and Mikolov, 2014), nous supposons que l'apprentissage simultané de représentations de plusieurs niveaux (à savoir, un niveau global pour des contextes de documents et un niveau local pour des contextes de mots et de concepts) permet d'affiner les représentations des composants. En plus, nous obtenons des représentations distribuées de documents, de mots et de concepts dans un même espace latent. Cela permet de tirer des bénéfices pour plusieurs tâche de TALN ou RI qui requirent des représentations de mots/concepts (e. g., l'expansion de la requête). Nous détaillons ensuite l'architecture de notre modèle en ligne, appelé $SD2V_{on}$.

4.1 *Architecture du réseau de neurones*

Notre modèle en ligne tripartite consiste en un réseau de neurones qui apprend des représentations de documents augmentées par une sémantique issue des concepts d'une ressource externe, permettant conjointement de dériver la représentation des mots et des concepts sous-jacents. Notre modèle augmente le processus d'apprentissage par la prise en compte des concepts dans la prédiction ainsi que dans le contexte. L'entrée du modèle est le texte annoté avec des concepts. Un exemple est illustré dans la Figure 4.5, où les mots soulignés sont associés à un concept sous-jacent. Par exemple, le mot "Apple" est associé à l'entité "`Apple_Inc`" dans la ressource DBpedia.

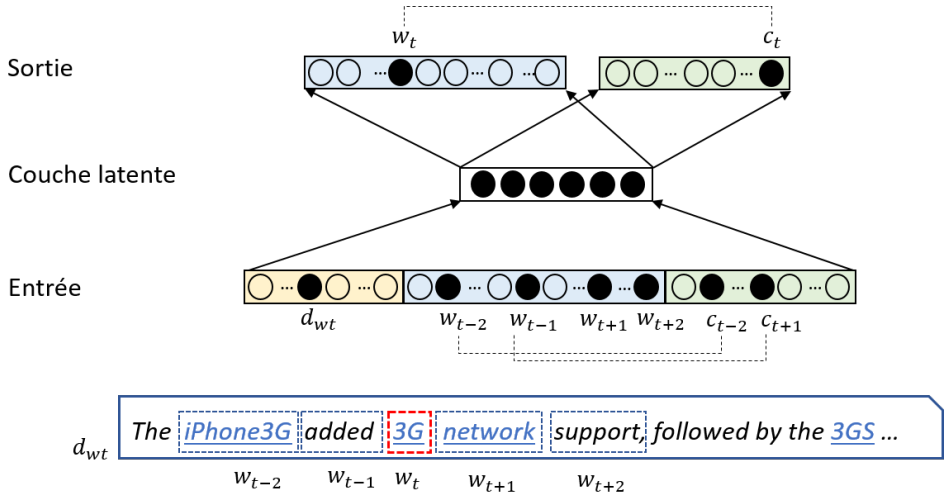


Figure 4.4 – Architecture du modèle neuronal tripartite

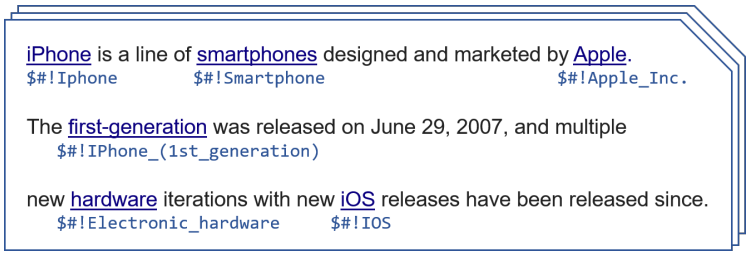


Figure 4.5 – Exemple d'un document annoté avec les concepts (entités de DBpedia)

En ce qui concerne l'apprentissage de représentations distribuées, similaire au modèle *ParagraphVector*, notre réseau apprend à prédire un mot en utilisant le contexte, y compris le document. La Figure 4.4 illustre l'architecture du modèle neuronal sur une instance d'apprentissage. A la différence du *ParagraphVector*, nous ajoutons les concepts associés aux mots dans le contexte de prédiction d'un mot. Autrement dit, la fenêtre glissante du contexte est déplacée au niveau mot, si

un mot dans la fenêtre a un concept associé sous-jacent, ce concept est ajouté au contexte de prédiction. De plus, si le mot à prédire est associé à un concept, notre modèle prédit aussi ce concept en utisant le même contexte de ce mot.

Formellement, l'apprentissage repose sur un ensemble D de documents d ; chaque document d est modélisé individuellement comme une séquence de mots ordonnés $\mathcal{W}_d$; un mot $w_i \in \mathcal{W}_d$ dans le document peut être associé à un concept c_i ; V représente le vocabulaire (c'est-à-dire les mots) des documents de la collection D et C correspond à l'ensemble des concepts identifiés dans la collection ; ces concepts sont issus d'une ressource sémantique externe $\mathcal{R}$. Cette dernière fournit des connaissances au travers de concepts et de relations entre concepts. Nous rappelons que dans ce travail, comme avec le modèle hors ligne, nous considérons seulement les associations mot-concept pour des mots simples (uni-grammes).

4.2 *Mécanismes d'apprentissage du réseau*

4.2.1 *Apprentissage de représentations de documents, de mots et de concepts*

Afin d'apprendre la représentation de documents de façon conjointe à l'apprentissage de représentations des mots et concepts identifiés dans le contexte du document, nous proposons d'étendre le modèle de représentations de documents *Paragraph Vector* (Le and Mikolov, 2014). Plus particulièrement, les représentations de documents (également appelés vecteurs de documents) $\vec{d}$ sont apprises en fonction de leurs mots et concepts en maximisant la prédiction des vecteurs de mots $\vec{w}$ et de concepts $\vec{c}$ en fonction de leur contexte. La fonction objectif répond à l'hypothèse (H1) énoncée précédemment : apprendre la représentation de plusieurs niveaux de granularité (documents, mots et concepts) en fonction de la prédiction des mots et concepts qui occurrent dans une fenêtre de contexte multi-niveaux. Ainsi, la fonction objectif de l'apprentissage conjoint document-mot-concept maximise la log-vraisemblance suivante :

$$J_{SD2V} = \sum_{d \in D} \sum_{w_t \in \mathcal{W}_d} [\log p(w_t | w_{t\pm k}, c_{t\pm k}, d) + \log p(c_t | w_{t\pm k}, c_{t\pm k}, d) - \frac{\gamma}{|d|} ||\vec{d}||^2] \tag{4.7}$$

où l'ensemble des mots du document d est noté $\mathcal{W}_d$; k correspond à la taille de la fenêtre de contexte liée à un mot cible w_t ; c_t est le concept associé au mot w_t en fonction de son context ; $\frac{\gamma}{|d|} ||\vec{d}||^2$ est la régularisation qui permet de limiter le sur-apprentissage lié à l'apprentissage des textes (Ai et al., 2016b) avec $|d|$ correspondant à la longueur du document et γ est le coefficient de régularisation. La

probabilité $p(w_t|w_{t\pm k}, c_{t\pm k}, d)$ du mot w_t étant donné son contexte est définie par une fonction soft-max :

$$p(w_t|w_{t\pm k}, c_{t\pm k}, d) = \frac{\exp(\vec{w}_t{}^\top \cdot \overline{h}_{w_t})}{\sum_{w' \in V} \exp(\vec{w'}^\top \cdot \overline{h}_{w_t})} \tag{4.8}$$

où V correspond au vocabulaire de la collection ; $\overline{h}_{w_t}$ représente la représentation du contexte moyennant les vecteurs v des mots dans le contexte $w_{t\pm k}$ et des concepts dans le contexte $c_{t\pm k}$ et incluant le vecteur document $\vec{d}$. Cette représentation $\overline{h}_{w_t}$ est estimée ainsi :

$$\overline{h}_{w_t} = \frac{1}{m}\left(\vec{d} + \sum_{\substack{-k \leq j \leq k \\ j \neq 0}} (\vec{w}_{t+j} + \vec{c}_{t+j})\right) \tag{4.9}$$

où m est le nombre de vecteurs dans le contexte de prédiction, y compris le document, les mots voisins dans la fenêtre k et les concepts associés à ces mots.

De façon similaire, la probabilité $p(c_t|w_{t\pm k}, c_{t\pm k}, d_{w_t})$ du concept c_t en fonction de son contexte est estimée comme suit :

$$p(c_t|w_{t\pm k}, c_{t\pm k}, d) = \frac{\exp(\vec{c}_t{}^\top \cdot \overline{h}_{c_t})}{\sum_{c' \in C} \exp(\vec{c'}^\top \cdot \overline{h}_{c_t})} \tag{4.10}$$

où C correspond au ensemble des concepts de la collection ; $\overline{h}_{c_t}$ est la représentation du vecteur de contexte lié au concept c_t, qui est aussi le contexte lié au mot w_t, autrement dit, $\overline{h}_{c_t} = \overline{h}_{w_t}$.

Similaire à la formule 4.1 du modèle PV-DM, les probabilités décrites dans les formules (4.8) et (4.10) sont également difficiles à estimer en raison de la taille importante du vocabulaire V et l'ensemble des concepts C. Nous adoptons la même stratégie d'échantillonnage négatif comme avec le modèle hors ligne $SD2V_{off}$ pour définir des fonctions objectif alternatives pour chaque élément $e_t \in \{w_t; c_t\}$:

$$p(e_t|w_{t\pm k}, c_{t\pm k}, d) = \log \sigma(\vec{e}_t{}'^\top \cdot \overline{h}_{e_t}) + \sum_{i=1}^{n} \mathbb{E}_{e_i \sim P_n(e)} \left[\log \sigma(-\vec{e}_i{}'^\top \cdot \overline{h}_{e_t})\right] \tag{4.11}$$

où $\sigma(x)$ correspond à la fonction sigmoid $\sigma(x) = \frac{1}{1+e^{-x}}$ et $\mathbb{E}_{e_i \sim P_n(e)}$ est la valeur attendue de $\log \sigma(-\vec{e}_i{}'^\top \cdot \overline{h}_{e_t})$ quand e_i est tiré de la distribution uniforme pondérée $P_n(e)$, comme réalisé par Ai et al. (2016b).

5 Intégration des contraintes relationnelles

Nous avons présenté deux modèles d'apprentissage de représentations qui exploitent les sémantiques des ressources externes via l'association de concepts dans le texte des documents. Dans cette section, nous détaillons notre amélioration de ces deux modèles avec l'hypothèse H2 - la prise en compte des relations entres les mots et les concepts, définies dans la ressource, augmente les représentations distribuées de texte à plusieurs niveaux. L'intuition sous-jacente est d'intégrer dans le processus d'apprentissage, des contraintes de relations entre des mots qui peuvent ne pas être (suffisamment) mises en évidence dans les contextes des documents utilisés pour l'apprentissage basé sur l'analyse distributionnelle ; c'est particulièrement le cas lorsque des mots, pourtant sémantiquement reliés, occurrent peu fréquemment dans les mêmes contextes en partie en raison de la diversité du vocabulaire. Plus précisément, pour répondre à la question de recherche **RQ3**, nous proposons deux approches pour intégrer les contraintes relationnelles :

• *Contrainte intégrée par régularisation de la fonction objectif*. Cette approche consiste à modifier la fonction objectif par un terme de régularisation. Ce dernier vise à ajuster les représentations des composants (mots/concepts) de façon à ce que les mots/concepts reliés dans la ressource aient les représentations proches dans l'espace latent.
• *Contrainte exprimée dans les instances d'apprentissage*. A la différence de l'approche précédente, cette approche consiste à intégrer les relations entre mots et concepts de façon explicite dans les contextes des données d'apprentissage. Le principe consiste à ajouter au contexte de prédiction de chaque objet (mot/concept) les objets reliés aux objets du contexte. Notre intuition est que des mots similaires aux mots dans le contexte de prédiction d'un mot, peuvent aussi aider à prédire ce mot. Ainsi, en apprenant à prédire un objet sachant les objets voisins étendus avec leurs objets reliés, les représentations des objets seront ajustées avec la sémantique relationnelle intégrée.

Avant de détailler ces deux approches, nous présentons d'abord les notions de "relation" utilisées dans ces deux approches.

5.1 *Relation entre les mots et les concepts*

Dans ce modèle, une ressource sémantique $\mathcal{R}$ est modélisée par un graphe $\mathcal{G} = (\mathcal{V}, L)$ où $\mathcal{V}$ est un ensemble de nœuds et L est un ensemble de liens. Chaque nœud $v_i =< c_i, T_i >$ représente un concept c_i et ses termes associés T_i. Les termes associés T_i à un concept c_i sont les mots qui permettent d'identifier ce concept (cf.

Chapitre 2, Section 1.1). Chaque lien $l_{i,i'}$ exprime une relation sémantique entre les concepts c_i et $c_{i'}$.

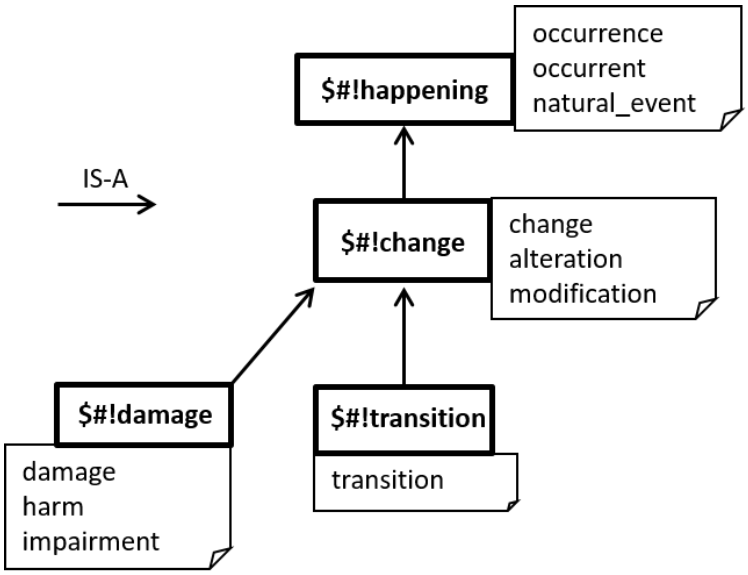


Figure 4.6 – Illustration de relation entre des concepts

La Figure 4.6 illustre cette modélisation avec l'exemple dans WordNet. Chaque nœud se compose d'un concept et d'une liste des termes. Un concept est présenté par un terme préféré avec le préfixe *$#!*. Les termes appartenant à un concept dans WordNet sont des synonymes, pour cette raison, un concept dans WordNet est appelé Synset. Le lien entre les concepts représente la relation hyponyme (*IS-A*). Dans cet exemple, le concept *$#!damage* a trois termes associés (*damage, harm, impairment*). Cela signifie que si on trouve un de ces termes dans le texte, on peut associé au concept *$#!damage*.

Dans notre modélisation de la contrainte relationnelle, nous distinguons deux types de relation :

— relation entre les concepts r_C
— relation entre les mots r_W

• La relation entre les concepts r_C est exprimée explicitement par les liens L entre les nœuds. On dit que deux concepts c_i et $c_{i'}$ sont reliés s'il existe un lien $l_{i,i'}$ connecté ces deux concepts.
Par exemple, dans la Figure 4.6, il y a trois paires de concepts reliés sont : (*$#!damage*, *$#!change*), (*$#!transition*, *$#!change*), (*$#!change*, *$#!happening*).

Pour un concept c_i donné, nous notons $\mathcal{R}_{c_i}$ l'ensemble des concepts reliés au concept c_i. Particulièrement, nous définissons $\mathcal{R}_C$ comme l'ensemble de toutes les paires de concepts reliés dans la ressource.

• La relation entre les mots r_W est déduite à partir des termes appartenant à un concept. Autrement dit, deux mots sont reliés s'ils appartiennent à la liste des termes associés à un concept. Par exemple, quelques paires de mots reliés dans l'exemple ci-dessus sont : (damage, harm), (harm, impairment), (change, alteration). Pour un mot w_i donné, nous notons $\mathcal{R}_{w_i}$ l'ensemble des mots reliés au mot w_i. Nous définissons aussi $\mathcal{R}_W$ comme l'ensemble de toutes les paires de concepts reliés dans la ressource.

Avec cette modélisation des relations, nous définissons ensuite deux approches pour intégrer les contraintes relationnelles aux modèles d'apprentissage de représentations présentés, à savoir le modèle hors ligne $SD2V_{off}$ et le modèle en ligne $SD2V_{on}$.

5.2 *Contrainte intégrée par régularisation de la fonction objectif*

Dans cette section, nous présentons notre première méthode pour intégrer dans les représentations distribuées les contraintes relationnelles issues d'une ressource externe. Inspirés par les travaux précédents (Yu and Dredze, 2014; Xu et al., 2014), nous proposons de régulariser la fonction objectif afin d'intégrer les contraintes relationnelles dans les représentations de mots. L'intuition est de régulariser le processus d'apprentissage en ajoutant une contrainte dans la fonction objectif qui conserve la similarité vectorielle entre les mots ou les concepts connectés dans la ressource externe. La régularisation ajustera les représentations des mots (ou des concepts), et simultanément l'apprentissage de représentations de documents, de sorte que les mots (ou les concepts) connectés aient des représentations proches. L'apprentissage de représentations est réalisé en maximisant un objectif défini, en y ajoutant un terme de régularisation par relation, les vecteurs de représentations sont ainsi appris avec une contrainte qui fait maximiser la similarité des objets (mot/concept) connectés dans la ressource externe. Par souci de simplicité, nous notons un objet o pour représenter soit un mot w, soit un concept c.

De façon formelle, notre objectif est de maximiser la similarité entre les objets (mots ou concepts) (o_i, o_j) qui sont reliés dans la ressource sémantique au travers le terme de régularisation suivant :

$$J_{Reg} = \sum_{(o_i,o_j)\in\mathcal{R}_O} sim(\vec{o_i}, \vec{o_j}) \tag{4.12}$$

où o dénote un objet, soit le mot, soit les concept ; $\mathcal{R}_O$ est l'ensemble de toutes les paires d'objets connectés définies dans la ressource $\mathcal{R}$; $sim()$ est une fonction de similarité vectorielle (e. g., similarité cosinus).

Nous détaillons ensuite l'intégration de ce terme de régularisation dans nos deux modèles d'apprentissage hors ligne et en ligne.

5.2.1 *Intégration dans le modèle hors ligne*

Pour rappel, dans le modèle hors ligne, nous avons deux modèles d'apprentissage de représentations distribuées, à savoir PV-DM et *conceptualDoc2vec*. Pour intégrer la contrainte relationnelle via la régularisation de l'objectif d'appentissage, nous ajoutons le terme de régularisation (Equation 4.12) dans la fonction objectif de chaque modèle, en gardant le type de relations (mot ou concept) selon le modèle d'apprentissage.

Plus précisément, pour intégrer la contrainte relationnelle au modèle *Paragraph-Vector* (PV-DM), nous ajoutons la régularisation sur les mots. La régularisation est appliquée sur les mêmes vecteurs de présentations appris avec le modèle PV-DM. Cependant, cette régularisation est déroulée de façon indépendante au processus d'apprentissage de représentations distribuées. Autrement dit, à chaque itération d'apprentissage du modèle PV-DM qui prédit un mot w_t, la régularisation est appliquée sur une paire de mots (w_i, w_j) tirée par hasard dans l'ensemble des paires de mots connectés $\mathcal{R}_W$. Le modèle régularisé PV_{Reg} est entraîné pour maximiser la fonction objectif suivante, sur chaque document d de la collection :

$$J_{PV_{Reg}} = \sum_{w_t \in d} \left[logP(w_t \mid w_{t\pm k}, d) - \frac{\gamma}{|d|} ||\vec{d}||^2 \right] + \alpha_W \sum_{(w_i, w_j) \in \mathcal{R}_W} sim(\vec{w}_i, \vec{w}_j) \quad (4.13)$$

où w_t est t^e mot dans le texte du document d ; k exprime la taille de la fenêtre de contexte ; $w_{t\pm k}$ représente l'ensemble des mots entre $t-k$ et $t+k$, sans inclure le mot w_t ; $sim(\bullet, \bullet)$ est la mesure de similarité cosinus entre deux vecteurs ; et α_W est le coefficient de force de régularisation.

De façon similaire, pour le modèle *conceptualDoc2vec* (*cd2v*), où l'apprentissage est basé sur les concepts, nous préparons d'abord l'ensemble des paires de mots reliés dans la ressource externe $\mathcal{R}_C$. L'objectif du modèle est ajouté par le terme de régularisation sur les concepts connectés. Ainsi, le modèle régularisé $cd2v_{Reg}$ a pour objectif de maximiser la log-vraisemblance suivante, sur chaque document d de la collection

$$J_{cd2v_{Reg}} = \sum_{c_t \in \mathcal{C}_d} \left[logP(c_t \mid c_{t\pm k}, d) - \frac{\gamma}{|\mathcal{C}_d|} ||\vec{d}||^2 \right] + \alpha_C \sum_{(c_k, c_l) \in \mathcal{R}_C} sim(\vec{c_k}, \vec{c_l}) \quad (4.14)$$

où c_t est t^e concept de l'ensemble des concepts ordonnés $\mathcal{C}_d$ dans le document d. k exprime la taille de la fenêtre de contexte, $c_{t\pm k}$ représente l'ensemble des concepts de positions comprises entre $t-k$ et $t+k$, sans inclure le concept c_t ; $sim(\bullet, \bullet)$ est la mesure de similarité cosinus entre deux vecteurs ; α_C est le coefficient de force de régularisation.

5.2.2 *Intégration dans le modèle en ligne*

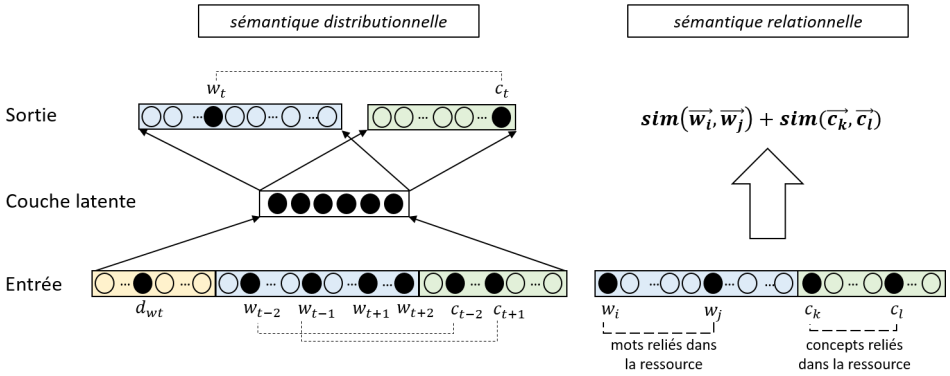


Figure 4.7 – Architecture du modèle SD2V avec la régularisation de sémantique relationnelle.

Comme le modèle en ligne tripartite $SD2V_{on}$ est entraîné en prenant en compte les mots et les concepts dans le contexte, nous ajoutons à ce modèle une régularisation sur les représentations de mots ainsi que les représentations de concepts. Avec la même intuition que dans le modèle offline, cette contrainte ajouté dans la fonction objectif sert à conserver la similarité vectorielle entre les mots ou les concepts connectés dans la ressource externe. Avec le terme de régularisation ajouté, le modèle en ligne apprend des représentations distribuées avec un objectif secondaire de corriger les représentations de mots (ou de concepts) à être proches si ces mots (ou concepts) sont reliés par une relation définie dans la ressource sémantique. Pour ce faire, nous préparons un ensemble de paires de mots connectés $\mathcal{R}_W$ et un ensemble de paires de concepts connectés $\mathcal{R}_C$ issues de la ressource externe. Puis, basé sur cette base de connaissance, nous ajoutons les composants de régularisation J_{Reg_W} et/ou J_{Reg_C} à la fonction objectif du modèle, qui maximise la similarité vectorielle de chaque paires de mots et/ou concepts connectés. Cette méthode est illustrée dans la Figure 4.7.

La fonction objectif finale répond aux deux hypothèses énoncées précédemment : 1) la composante J_{SD2V} (Hypothèse *H1*) qui apprend la représentation de plusieurs niveaux de granularité (documents, mots et concepts) en fonction de la prédiction des mots et concepts qui occurrent dans une fenêtre de contexte

multi-niveaux ; 2) le composante J_{Reg_W} et/ou J_{Reg_C} (Hypothèse *H2*) qui régularise l'apprentissage de relations en prenant en compte les contraintes de sémantique relationnelle. Ainsi, la fonction objectif est formalisée comme suit :

$$J_{SD2V_{Reg}} = J_{SD2V} + \alpha_W * J_{Reg_W} + \alpha_C * J_{Reg_C} \tag{4.15}$$

où J_{SD2V} est la fonction objectif originale de notre modèle en ligne, définie dans l'Equation 4.7 ; α_W et α_C sont les coefficients de combinaison pour les composants J_{Reg_W} et J_{Reg_C} respectivement, qui sont les termes de régularisation sur les mots et/ou les concepts. Ces termes de régularisation sont définis comme suit :

$$J_{Reg_W} = \sum_{(w_i,w_j)\in\mathcal{R}_W} sim(\vec{w}_i, \vec{w}_j) \tag{4.16}$$

$$J_{Reg_C} = \sum_{(c_k,c_l)\in\mathcal{R}_C} sim(\vec{c}_k, \vec{c}_l) \tag{4.17}$$

où $\mathcal{R}_W$ et $\mathcal{R}_C$ sont l'ensemble de paires de mots et de concepts, respectivement, connectés issues de la ressource externe ; $sim(\bullet, \bullet)$ est la similarité cosinus entre deux vecteurs.

5.3 *Contrainte exprimée dans les instances d'apprentissage*

Dans cette section, nous présentons notre deuxième méthode pour intégrer dans les représentations distribuées les contraintes relationnelles issues d'une ressource externe. L'objectif ici est d'enrichir les instances d'apprentissage avec les rélations établies dans la ressource externe. Notre intuition est basée sur l'idée qu'un mot peut être prédit en fonction des mots dans le contexte local. Pour aller plus loin, nous supposons que des mots similaires aux mots dans le contexte de prédiction d'un mot, peuvent aussi aider à prédire ce mot.

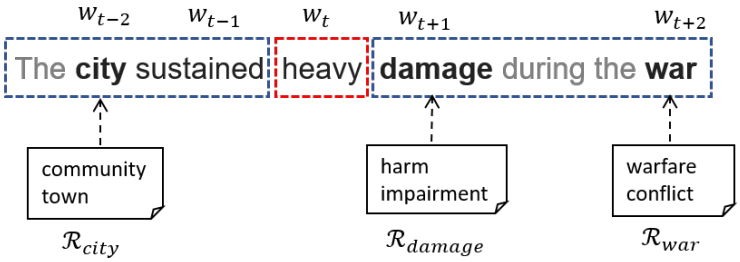


Figure 4.8 – Illustration de contrainte exprimée par les instances d'apprentissage

La Figure 4.8 illustre un exemple d'intégration de contrainte dans les instance d'apprentissage. Dans cet exemple, pour apprendre à prédire le mot *heavy*, le

modèle PV-DM est entraîné à maximer la probabilité $P(heavy|context_{heavy}$, où $context_{heavy}$ sont les mots voisins dans la fenêtre de taille k, et le document. Plus précisément, avec une fenêtre de contexte de taille 2, les mots dans le contexte $context_{heavy}$ de la prédiction pour le mot *heavy* sont {*city*, *sustained*, *damage*, *war*} (en ignorant les mots vides).
Avec notre contrainte de relation, le contexte de prédiction est étendu avec les mots reliés $\mathcal{R}_{w_i}$ au mot w_i dans la fenêtre k. Dans cet exemple, le contexte de prédiction sera étendu avec les mots synonymes de *city*, *damage*, *war*, respectivement (*community*, *town*), (*harm*, *impairment*), (*warfare*, *conflict*).

Nous détaillons ensuite l'intégration de contraintes relationnelles dans nos modèles d'apprentissage de représentations présentés, à savoir le modèle hors ligne $SD2V_{off}$ et le modèle en ligne $SD2V_{on}$.

5.3.1 *Intégration dans le modèle hors ligne*

Pour rappel, nous avons appliqué deux modèles d'apprentissage de représentations distribuées, à savoir PV-DM et *conceptualDoc2vec*. Pour intégrer la contrainte relationnelle via les instances d'appentissage, nous enrichissons le contexte de prédiction par les éléments (respectivement mots, concepts) réliés selon le modèle (respectivement PV-DM, *conceptualDoc2vec*).

Plus précisément, pour intégrer la connaissance relationnelle au modèle *ParagraphVector*, nous ajoutons les mots reliés aux mots du contexte de prédiction. Pour un mot w_i dans la fenêtre de contexte, nous définissons l'ensemble $\mathcal{R}_{w_i}$ des mots reliés au mot w_i via la ressource externe. Ces mots reliés sont ajoutés dans le contexte de prédiction lors de l'apprentissage du modèle PV-DM. Ainsi, l'apprentissage est fait par la prédiction d'un mot w_t en utilisant le contexte compris du document actuel d, des mots voisins $w_{t\pm k}$ (dans la fenêtre k) et des mots reliés $\mathcal{R}_{w_{t\pm k}}$ à chaque mot voisin. Le modèle étendu *PVr* est entraîné pour maximiser la fonction objectif suivante, sur chaque document d de la collection :

$$J_{PVr} = \sum_{w_t \in d} \left[logP(w_t \mid w_{t\pm k}, \mathcal{R}_{w_{t\pm k}}, d) - \frac{\gamma}{|d|} ||\vec{d}||^2 \right] \tag{4.18}$$

où w_t est t^e mot dans le texte du document d ; k exprime la taille de la fenêtre de contexte ; $w_{t\pm k}$ représente l'ensemble de mots entre $t-k$ et $t+k$, sans inclure le mot w_t ; $\frac{\gamma}{|d|}||\vec{d}||^2$ est la normalisation par la longueur du document d.

De façon similaire, pour le modèle *conceptualDoc2vec* (*cd2v*), où l'apprentissage est basé sur les concepts, nous préparons d'abord une base des concepts reliés par une relation (e. g., IS-A). Pour chaque concept c_i dans la fenêtre de contexte,

nous identifions l'ensemble $\mathcal{R}_{c_i}$ des concepts directement connectés au concept c_i. Puis, nous définissons le modèle étendu *cd2vr* par la prédiction d'un concept c_t en utilisant le contexte compris du document actuel d, des concepts voisins $c_{t\pm k}$ (dans la fenêtre k) et des concepts reliés $\mathcal{R}_{c_{t\pm k}}$ à chaque concept voisin. Plus formellement, le modèle étendu *cd2vr* a pour objectif de maximiser la log-vraisemblance suivante, sur chaque document d de la collection :

$$J_{cd2vr} = \sum_{c_t \in \mathcal{C}_d} \left[logP(c_t \mid c_{t\pm k}, \mathcal{R}_{c_{t\pm k}}, d) - \frac{\gamma}{|\mathcal{C}_d|} ||\vec{d}||^2 \right] \tag{4.19}$$

où c_t est t^e concept de l'ensemble de concepts ordonnés $\mathcal{C}_d$ dans le document d. k exprime la taille de la fenêtre de contexte, $c_{t\pm k}$ représente l'ensemble de concepts de positions comprises entre $t-k$ et $t+k$, sans inclure le concept c_t.

5.3.2 *Intégration dans le modèle en ligne*

Pour rappel, la fonction objectif (Equation 4.7) de l'apprentissage conjoint document-mot-concept maximise la log-vraisemblance suivante :

$$J_{SD2V} = \sum_{d \in D} \sum_{w_t \in \mathcal{W}_d} [\log p(w_t | w_{t\pm k}, c_{t\pm k}, d) + \log p(c_t | w_{t\pm k}, c_{t\pm k}, d) - \frac{\gamma}{|d|} ||\vec{d}||^2] \tag{4.20}$$

où $w_{t\pm k}$ et $c_{t\pm k}$ sont respectivement les mots et les concepts dans la fenêtre de voisinage k. De la façon similaire que nous avons utilisé avec le modèle hors ligne, la contrainte relationnelle est intégrée dans le modèle en ligne via les instances d'apprentissage, plus particulièrement le contexte de prédiction. Comme le contexte de prédiction dans ce modèle en ligne peut contenir à la fois les mots et les concepts, nous appliquons l'expansion par les deux types de relation : des mots reliés au mots du contexte, des concepts reliés aus concepts du contexte. Plus précisément, pour chaque mot w_i et c_i (si il existe un concept associé à ce mot), nous identifions l'ensemble $\mathcal{R}_{w_i}$ des mots reliés au mot w_i, et aussi l'ensemble $\mathcal{R}_{c_i}$ des concepts reliés au concept c_i. Notons $context_{w_t}$ de prédiction du mot w_t. Selon l'Equation 4.20 le contexte $context_{w_t}$ du mot w_t contient des mots voisins $w_{t\pm k}$, des concepts voisins $c_{t\pm k}$ et le document d, qui est aussi le contexte $context_{c_t}$ du concept c_t est le suivant :

$$context_{w_t} = context_{c_t} = (w_{t\pm k}, c_{t\pm k}, d) \tag{4.21}$$

Pour intégrer la contrainte relationnelle par les instances, nous étendons ce contexte de prédiction. Le contexte étendu de prédiction $contextR_{w_t}$ est ajouté par les mots reliés au mots voisins et par les concepts reliés aux concepts voisins :

$$contextR_{w_t} = (w_{t\pm k}, \mathcal{R}_{w_{t\pm k}}, c_{t\pm k}, \mathcal{R}_{c_{t\pm k}}, d) \tag{4.22}$$

où $w_{t\pm k}$ sont les mots dans la fenêtre k ; $c_{t\pm k}$ sont les concepts reliés aux mots dans la fenêtre k ; $\mathcal{R}_{w_{t\pm k}}$ (respectivement $\mathcal{R}_{c_{t\pm k}}$) est l'ensemble des mots (respectivement concepts) reliés à chaque mot $w_{t\pm k}$ (respectivement concept $c_{t\pm k}$) dans le contexte.

Ainsi, la fonction objectif du modèle en ligne étendu est de maximiser la probabilité d'obtenir le mot w_t et le concepts $\vec{c_t}$ en fonction de leur contexte étendu $contextR_{w_t}$:

$$J_{SD2V_{Ins}} = \sum_{d \in D} \sum_{w_t \in \mathcal{W}_d} \left[\log p(w_t | contextR_{w_t}) + \log p(c_t | contextR_{w_t}) - \frac{\gamma}{|d|} ||\vec{d}||^2 \right] \tag{4.23}$$

Nous notons $SD2V_{on}^{Ins}$, ce modèle d'apprentissage en ligne intégré des contraintes par les instances.

6 Cadre expérimental

Afin de valider nos deux modèles d'apprentissage de représentations (hors ligne et en ligne), nous avons mis en place un protocole d'évaluation comparative basé sur les tâches de TALN et de RI. L'objectif de notre évaluation est double :

1. (EV1) Evaluation comparative de nos deux modèles en testant l'effet du pré-entraînement, sans considérer les relations. Nous menons plusieurs analyses pour comparer la qualité des représentations apprises par chacun des deux modèles. Nous précisons aussi que dans le processus d'apprentissage des représentations, les vecteurs de représentations peuvent être initialisés soit aléatoirement soit par les représentations pré-entraînées sur un autre corpus (pour porter plus de sémantique).
2. (EV2) Evaluation comparative de nos deux modèles en testant les variantes de prise en compte des relations. Cet objectif nous aide à analyser les différents apports entre deux façons d'intégrer les contraintes relationnelles, à savoir intégration par régularisation de la fonction objectif (*reg*) et intégration par les instances d'apprentissage (*ins*).

6.1 *Jeux de données et ressources sémantiques*

Pour mesurer la robustesse de notre approche d'apprentissage de représentation, nous menons des expérimentations sur des domaines d'application génériques et domaines d'application spécifiques. Nous considérons le domaine médical comme le domaine d'application spécifique en raison de sa grande difficulté

à résoudre le problème du fossé sémantique (grande variabilité du langage et de l'orthographe, utilisation fréquente d'acronymes et d'abréviations, ambiguïté inhérente aux processus automatisés pour interpréter les concepts selon les contextes). Plus précisément, les approches en ligne et hors ligne apprennent à intégrer des documents sur trois jeux de données principaux dont les statistiques sont présentées dans le tableau 4.1 :

• La collection Robust04 [1] qui est le jeu de données de "nouvelles" utilisé dans le campagne d'évaluation standard TREC Robust Track 2004 comprenant 528 155 documents et 250 sujets. Nous avons utilisé les titres des sujets comme requêtes, par exemple "Best Retirement Country".

• Le jeu de données OHSUMED (Hersh et al., 1994) composé d'un ensemble de 348 566 des références de MEDLINE et 63 requêtes. Ce jeu de données est connu sous le nom de collection standard à grande échelle pour la RI médical ad hoc (Stokes et al., 2009). La tâche de récupération consiste en une recherche liée à la santé se référant à la situation dans laquelle un médecin cherche des articles scientifiques pertinents fournissant une assistance enrichissante pour obtenir un diagnostic/prognostic précis et/ou proposer un traitement considérant le cas médical. Un exemple de requête est "*adult respiratory distress syndrome*".

• Le jeu de données TREC Med [2] comprend plus de 17 000 rapports de visites médicales anonymes et 35 requêtes. La tâche consiste en une recherche clinique de cohortes. Cette dernière vise à identifier les cohortes dans les études cliniques pour la recherche comparative sur l'efficacité de traitement. Nous utilisons la collection standard TREC Med, dans laquelle les requêtes spécifient des ensembles particuliers de maladies ou d'affections et des traitements ou interventions particuliers, exprimés par les médecins en langage naturel. Un exemple de requête est "*find patients with gastroesophageal reflux disease who had an upper endoscopy*".

Pour enrichir les représentations avec des concepts et de la sémantique relationnelle, nous exploitons deux ressources sémantiques, respectivement dans le domaine d'application des ensembles de données :

• **DBpedia** avec sa large couverture. Les requêtes et les documents sont annotés par *TagMe* (Ferragina and Scaiella, 2010), un outil d'annotation de pointe pour lier le texte aux entités DBpedia. Nous utilisons les noms des entités de DBpedia (aussi appelées concepts dans notre modèle) pour annoter les requêtes et les documents et exploiter la relation `gold:hypernym`. Par souci de simplicité dans la description du modèle, nous faisons référence aux entités par concepts.

1. http ://trec.nist.gov/data/robust/04.guidelines.html
2. https ://trec.nist.gov/data/medical2012.html

	Robust	OHSUMED	TREC Med
Nombre de documents	528 155	348 566	17 000
Nombre de requêtes	250	63	35
Nb. moyen de mots d'un doc.	488	301	400
Nb. moyen de concepts d'un doc.	31	58	68
Nb. moyen de relations d'un doc.	164	148	200
Nb. moyen de mots d'une req.	3	5	7
Nb. moyen de concepts d'une req.	1	2	3
Nb. moyen de relations d'une req.	2	4	5

Tableau 4.1 – Statistiques des jeux de données

• **MeSH** qui est la terminologie la plus utilisée dans le domaine biomédical (Stokes et al., 2009). Cette ressource comprend 27 000 concepts, organisés en 16 catégories et structurées hiérarchiquement du plus général au plus spécifique. Nous utilisons l'outil Cxtractor [3] pour extraire des concepts. La relation "IS-A" dans la hiérarchie des concepts est exploitée pour la contrainte relationnelle.

6.2 *Tâche d'évaluation TALN*

L'objectif de ces tâches est de mesurer la qualité des représentations de textes à plusieurs niveaux par rapport à leur capacité à saisir la sémantique sous-jacente du texte. Pour cela, nous adoptons deux tâches d'analyse de qualité : similarité des mots, similarité des phrases et similarité des documents.

6.2.1 *Similarité des mots*

Nous évaluons les représentations de mots sur trois jeux de données différents et standards qui ont été largement utilisés pour mesurer la similarité des mots. Le premier est le jeu de données WS-353 (noté WS) (Finkelstein et al., 2001) qui contient 353 paires de mots anglais. Le deuxième est le jeu de données RG-65 (Rubenstein and Goodenough, 1965) qui contient 65 paires de noms. Le troisième est SimLex-999 (Hill et al., 2015), qui se compose de 666 paires de noms, 222 paires de verbes et 111 paires d'adjectifs. Nous utilisons également le jeu de données MEN (Bruni et al., 2012) formé de 3 000 paires de mots qui apparaissent au moins 700 fois dans un grand corpus web. Dans chaque jeu de données, toutes les paires de mots ont une note de similarité attribuée par des assesseurs humains. Nous

3. https ://sourceforge.net/projects/cxtractor

calculons la similarité cosinus entre les vecteurs des deux représentations associées aux paires de mots puis reportons le coefficient de corrélation de Spearman entre : 1) le classement obtenu des paires de représentations de mots issues de notre modèle et 2) le classement de ces mêmes paires de mots obtenu grâce aux scores attribués par les experts humains.

6.2.2 *Similarité des phrases (SentEval)*

Nous utilisons les tâches proposées dans un cadre d'évaluation de la qualité des phrases SentEval[4] (Conneau and Kiela, 2018). Ces tâches sont divisées en deux catégories :

• *La proximité sémantique des phrases*. Cette tâche vise à mesurer dans quelle mesure la similarité entre deux phrases représentées dans l'espace latent peut se rapprocher des scores de similarité notés par l'humain. La qualité des représentations est mesurée par la corrélation de Spearman entre les scores notés par l'humain et la similarité cosinus des représentations apprises par nos modèles. Dans nos expériences, nous présentons les résultats de l'indice de référence de la mission STS 2014 (Cer et al., 2017).

• *La classification des phrases*. Cette tâche vise à étiqueter la représentation des phrases selon une tâche spécifique : classification subjectivité/objectivité (SUBJ) (Wang and Manning, 2012), polarité d'opinion (MPQA) (Wiebe and Cardie, 2005), classification de type question (TREC) (Voorhees, 2001), et identification paraphrase (MSRP) (Dolan et al., 2004). Pour chaque jeu de données, un classifieur de régression logistique est appris à partir des représentations de phrases extraites et son efficacité est mesurée à l'aide de la précision de la classification.

Pour toutes ces tâches, nos modèles de représentations distribuées sont appliqués sur les phrases fournies pour inférer les représentations de phrases associées.

6.2.3 *Similarité des documents*

Cette tâche est décrite dans le travail de Le and Mikolov (2014). Elle consiste à discriminer la similarité des documents par rapport à une requête cible. Plus précisément, pour chaque requête du jeu de données, nous créons un pool de triplets de documents récupérés par un modèle de RI classique. Dans chaque triplet d'une requête, les deux premiers sont récupérés par les deux premiers résultats renvoyés pour cette requête et le troisième document est échantillonné au hasard à partir de l'ordonnancement des documents par rapport aux autres requêtes. L'objectif sous-jacent est de mesurer dans quelle mesure l'extension de la mesure de similarité du document (à savoir la similarité cosinus standard) estimée en utilisant des repré-

4. https://github.com/facebookresearch/SentEval

sentations de documents appris permet de fournir une similarité plus importante pour les documents issus de la même requête cible et une similarité plus faible pour les documents issus d'autres requêtes. Comme pour Le and Mikolov (2014), nous utilisons le taux d'erreur sur toutes les requêtes mesurant quand les représentations donnent une plus petite similarité pour les deux premiers documents que pour le troisième.

6.3 *Tâche d'évaluation RI*

L'objectif de ces tâches est de mesurer l'efficacité des représentations de documents dans les tâches RI et, par conséquent, leur capacité à saisir les signaux de pertinence. Nous adoptons deux techniques d'évaluation en RI qui impliquent l'utilisation des représentations de texte : *Réordonnancement de document* et *Expansion de requête*. La première technique vise à évaluer la qualité des représentations de documents apprises par nos modèles. La deuxième technique, exploitant les représentations de textes à plusieurs niveaux (mot/concept/document), permet de mieux analyser la qualité ainsi que la relation entre les représentations de textes à plusieurs niveaux.

6.3.1 *Réordonnancement de document*

Cette tâche vise à améliorer le score de pertinence d'un document avec un score supplémentaire issu de nos représentations apprises. Pour ce faire, nous utilisons le modèle proposé dans Liu et al. (2016) qui combine un score traditionnel de pertinence de document $IRScore(q, d)$ avec un score de similarité cosinus $NeuralScore_{KB}(q, d)$ calculé entre les représentations de la requête et du document :

$$RSV(q, d) = \alpha \cdot IRScore(q, d) + (1 - \alpha) \cdot NeuralScore_{KB}(q, d) \tag{4.24}$$

où α est un paramètre de combinaison ajusté en utilisant une validation croisée selon la métrique MAP ; $IRScore$ est obtenu en utilisant un modèle de RI traditionnel, à savoir BM25 ; et $NeuralScore_{KB}$ est la similarité cosinus entre les représentations de la requête et du document apprises à l'aide de notre modèle.

6.3.2 *Expansion de requête*

Cette tâche consiste à réécrire la requête initiale en exploitant la proximité des éléments (mots et/ou concepts) dans l'espace des représentations distribuées. Dans notre contexte, nous nous basons sur l'hypothèse que la pertinence pourrait

être saisie en calculant les similitudes entre les représentations de requêtes d'un côté et les représentations de mots/concepts de l'autre côté. Pour ce faire, nous nous appuyons sur le modèle de pointe proposé dans Zamani and Croft (2016b) qui suppose que les éléments candidats m sont identifiés à l'aide d'un critère de pertinence estimé par la similarité entre la requête d'un côté et le mot/concept de l'autre côté. Cette similarité est modélisée sous forme d'interpolation linéaire entre l'estimation du maximum de vraisemblance $p_{mle}(m|q)$ de la requête originale (à savoir, la probabilité basée sur le comptage de terme) et un score de similarité neuronale $p_{emb}(m|q)$:

$$p(m|q*) = \alpha p_{mle}(m|q) + (1-\alpha) p_{emb}(m|q) \tag{4.25}$$
$$p_{emb}(m|q) = \frac{\sigma(\vec{e}_m, \vec{q})}{\sum_{m' \in V} \sigma(\vec{e}_{m'}, \vec{q})}$$

où $\vec{q}$ et $\vec{e}_m$ sont respectivement les représentations de la requête q et de l'élément word/concept candidat m ; V est le vocabulaire. $sigma(\cdot, \cdot)$ indique l'exponentielle du cosinus de deux vecteurs et Z est le facteur de normalisation calculé en additionnant $\sigma(\vec{e}_{m'}, \vec{q})$ sur tous les termes m' dans le vocabulaire (à savoir tous les mots sur tous les documents ou tous les concepts extraits de tous les mots).

Pour obtenir le vecteur de représentation de la requête, qui n'est pas entraîné dans l'étape d'apprentissage, nous appliquons le technique d'inférence qui est une caractéristique originale du modèle ParagraphVector (Le and Mikolov, 2014). L'inférence de représentation du nouveau texte est faite en appliquant le modèle ParagraphVector entraîné sur le texte pour effectuer une nouvelle phase d'apprentissage avec les représentations de mots entraînées et fixées. De même façon, nos modèles d'apprentissage hors ligne et en ligne peuvent effectuer cette inférence pour obtenir le vecteur de la requête.

Soulignons cependant, l'expansion de la requête effectué avec notre modèle hors ligne est peu robuste. Comme l'espace de représentation finale de documents est rapprochée de deux espaces latents de documents (un sur les mots, un sur les concepts), la représentation finale des documents (ainsi de la requête) n'est pas dans le même espace des mots ou des concepts.

6.4 *Modèles de référence*

Pour évaluer la qualité de nos représentations de documents, nous comparons nos scénarios aux modèles de référence suivants :

• *Un modèle basé sur des statistiques d'occurrences simples.* Pour les tâches TALN, nous utilisons la modélisation traditionnelle des documents $TF - IDF$ dans laquelle les documents sont représentés par un vecteur mot pondéré en utilisant le

schéma TF-IDF. Bien que le TF-IDF soit bien adapté au cadre RI, nous utilisons plutôt le modèle d'ordonnancement traditionnel $BM25$ qui est un modèle de référence solide en RI. Ce référentiel vise à évaluer l'impact de la représentation des documents d'apprentissage.

• *Un modèle basé sur la représentation de mots*, noté AWE, qui construit des représentations de documents en faisant la moyenne des représentations de ses mots (Le and Mikolov, 2014; Vulić and Moens, 2015). Le but de la comparaison avec cette représentation est d'évaluer l'impact de la prise en compte d'un contexte à plusieurs niveaux (à savoir des concepts et des documents en plus des mots) dans l'apprentissage de la représentation de documents.

• *Un modèle basé sur la représentation de mot renforcée par une ressource sémantique*, noté AWE_R, qui prend la moyenne des représentations de mots construites à l'aide d'une ressource sémantique externe comme proposé dans Faruqui et al. (2015). Ce modèle de référence vise à évaluer l'impact de la prise en compte d'un contexte multi-niveaux (à savoir des documents en plus des mots et des concepts) dans l'apprentissage de la représentation de documents.

• *Un modèle basé sur la représentation de document*, noté $D2V$, qui fait référence au modèle Paragraph-Vectoriel Le and Mikolov (2014) à partir duquel nous construisons notre modèle neuronal étendu. Ce scénario permettrait d'évaluer l'impact de la prise en compte des concepts et des relations dans l'apprentissage de la représentation puisque ce scénario ne comprend que des contextes de mots et de documents.

Pour évaluer avec les tâches de RI des modèles de référence basés sur la représentation distribuée, ainsi que les scénarios de nos modèles, nous utilisons leurs représentations de mots et/ou de documents pour les injecter dans les Equations 4.24 et 4.25 (pour réordonnancement et expansion de requête, respectivement).

6.5 *Scénarios d'évaluation*

Pour analyser plusieurs aspects entre nos différentes configurations des modèles hors ligne et en ligne, nous adoptons les scénarios présentés dans le Tableau 4.2 pour nos propositions.

6.6 *Détails d'implémentation*

Pour les configurations de modèles basées sur la représentation distribuée (AWE, AWE_R, $D2V$, $SD2V$, $SD2VR$), nous fixons la dimension des vecteurs de représentations à 300 et sélectionnons empiriquement la taille de fenêtre $k = 8$. Après avoir supprimé les mots non alphanumériques, nous ne gardons que les

Acronyme	Référence	Objectif
$SD2V_{off}$	Méthode d'apprentissage hors ligne (section 3)	Evaluer l'impact de l'intégration des concepts dans les représentations de documents
$SD2V_{on}$	Méthode d'apprentissage en ligne (section 4)	
$SD2V_{Ins_{off}}$	Méthode d'apprentissage hors ligne intégrée de la contrainte relationnelle par instance d'entraînement (section 5.3.1)	Evaluer l'impact de la contrainte relationnelle apprise par l'instance d'entraînement
$SD2V_{Ins_{on}}$	Méthode d'apprentissage en ligne intégrée de la contrainte relationnelle par instance d'entraînement (section 5.3.2)	
$SD2V_{Reg_{off}}$	Méthode d'apprentissage hors ligne intégrée de la contrainte relationnelle par régularisation d'objectif (section 5.2.1)	Evaluer l'impact de la contrainte relationnelle apprise par la régularisation de la fonction objectif
$SD2V_{Reg_{on}}$	Méthode d'apprentissage en ligne intégrée de la contrainte relationnelle par régularisation d'objectif (section 5.2.2)	

Tableau 4.2 – Différents scénarios de nos modèles d'apprentissage de représentations

mots dont la fréquence dans le corpus est supérieure à 5. Le taux d'apprentissage initial est fixé à $0,02$ et diminue linéairement pendant l'optimisation par *SGD*. Nous utilisons la technique d'échantillonnage négatif où l'échantillon négatif est fixé à 5.
Le paramètre β de l'Equation 4.6 est réglé sur $0,75$; $0,8$ et $0,85$ pour les ensembles de données respectivement Robust, OHSUMED et TREC Med. Ces valeurs soulignent qu'il est utile de combiner à la fois des mots et des concepts pour représenter les documents, avec une prévalence plus élevée de mots dans le jeu de données TREC Med. Cela pourrait s'expliquer par le fait que les requêtes dans cette collection sont plus volumineuses. Concernant la régularisation de longueur γ dans l'Equation 4.7, nous testons $0,1$; 1 et 10 comme suggéré dans Ai et al. (2016b) ; la meilleure performance est obtenue avec $\gamma = 0.1$. Les coefficients α_W et α_C dans l'Equation 4.15 sont fixés à 1. Pour la tâche de réordonnancement, la combinaison (Equation 4.24) est effectuée avec $\alpha = 0,85$. Pour la tâche d'extension de la requête, nous avons fait varier le nombre d'éléments m étendant la requête d'origine de 1 à 10. Ces éléments incluent des mots et des concepts en fonction de la probabilité $p(m|q*)$ et leur nombre a été défini à 2. Toutes ces valeurs ont été optimisées par validation croisée via la métrique MAP. Pour les deux tâches de RI, les scores sont calculés à l'aide du moteur de recherche Indri [5].

5. https ://www.lemurproject.org/indri.php

7 Résultats d'évaluation

Nous présentons dans cette section l'évaluation comparative de nos modèles d'apprentissage de la représentation de documents selon deux critères : le processus d'apprentissage (EV1) et méthode d'intégration des relations les relations dans la représentation de documents (EV2).

7.1 *Evaluation des modèles sans contrainte de relations*

Notre objectif est de comparer deux approches d'apprentissage en ligne et hors ligne à travers diverses tâches d'évaluation TALN et RI.

7.1.1 *Efficacité par rapport aux modèles de référence*

Nous présentons les résultats des tâches de RI (réordonnancement et expansion de requête) dans le Tableau 4.3 (la colonne *%Chg* indique le taux d'accroissement de chaque configuration par rapport au modèle BM25). Dans ce tableau, nous pouvons observer que nos deux modèles (hors ligne et en ligne) sont capables de saisir des signaux de pertinence qui permettent d'améliorer l'ordonnancement des documents, par rapport aux modèles de référence. Il y des améliorations importantes et significatives selon les scénarios. Nous faisons les deux principaux constats suivants :

• En comparant avec le modèle BM25, nos deux modèles obtiennent des améliorations plus importantes sur le domaine médical (OHSUMED, TREC Med) que sur le domaine générique (Robust). Le modèle de référence de le BM25 est difficile à surpasser pour le jeu de données Robust, même pour les autres modèles de référence qui sont également basées sur l'apprentissage de la représentation. En effet, les scénarios basés sur la représentation distribuées (modèles de référence et nos deux modèles) permettent d'obtenir des améliorations généralement de 11% à 40% sur les jeux de données médicales. Tandis que cette amélioration reste négligeable sur Robust (%Chg varie entre $-0,40\%$ et $0,16\%$)

Cela suggère que les domaines spécifiques tels que le domaine médical nécessitent un scénario permettant de capturer davantage d'inférences textuelles qu'un simple modèle statistique basé sur l'indépendance des termes (à savoir le BM25). En effet, les statistiques des jeux de données (cf. Tableau 4.1) indiquent que les documents des jeux de données médicales sont caractérisés par un plus grand nombre de concepts (60 concepts en moyenne) que le jeu de données génériques Robust (31 concepts en moyenne).

		Réordonnancement				Expansion de requête			
		Robust		OHSUMED		TREC Med		Robust		OHSUMED		TREC Med	
	Modèles	MAP	%Chg	MAP	%Chg	MAP	%Chg	MAP	%Chg	MAP	%Chg	MAP	%Chg
	BM25	0,2510		0,2147		0,3120		0,2510		0,2147		0,3120	
	AWE	0,2500	-0,40%	0,2010	-2,24%	0,349	11,83%	0,250	-0,36%	0,2520	17,51%	0,2890	-7,08%
	AWE_R	0,2510	0,04%	0,3010	40,20%	0,3500	12,24%	0,2510	0,00%	0,2540	18,30%	0,2901	-7,02%
	D2V	0,2505	-0,20%	0,3000	39,78%	0,3560	14,07%	0,2511	0,04%	0,2550	19,19%	0,2910	-6,67%
Non pré-entraîné	$SD2V_{off}$	0,2510	0,00%	0,3018	40,57%	0,3591	15,10%	0,2464	-1,83%	0,2580	20,17%	0,3205	2,72%
	$SD2V_{on}$	0,2507	-0,12%	0,3020	40,66%	0,3554	13,91%	0,2443	-2,67%	0,2599	21,05%	0,2889	-7,40%
Pré-entraîné w2v	$SD2V_{off}$	0,2510	0,16%	0,3020	40,66%	0,3581	14,78%	0,2458	-2,07%	0,2579	20,12%	0,3227	3,43%
	$SD2V_{on}$	0,2510	-0,08%	0,3019	40,61%	0,3582	14,81%	0,2440	-2,79%	0,2592	20,73%	0,2890	-7,37%

Tableau 4.3 – Comparaison des approches d'apprentissage hors ligne/en ligne sur les tâches de RI.

Cette tendance est similaire avec les requêtes qui sont plus verbeuses et qui incluent plus de concepts. Cela confirme que la recherche médicale exige des techniques efficaces (comme des modèles d'apprentissage de la représentation) pour capturer la sémantique du texte et de la requête afin de déduire les signaux de pertinence.

Nous notons aussi que nos deux modèles obtiennent généralement des scores plus élevés que d'autres modèles de référence de l'apprentissage de représentation, à savoir AWE, AWE_R, $D2V$, quand ces derniers sont injectés dans une tâche de réordonnancement ou une tâche d'expansion de requête. Par exemple, pour la tâche de réordonnancement sur TREC Med, notre scénario $SD2V_{off}$ (non pré-entraîné) obtient la valeur de MAP à $0,3591$ qui est plus élevée que celles des modèles AWE, AWE_R et $D2V$ ($0,349$, $0,350$ et $0,356$ respectivement). Cela suggère l'avantage de combiner la sémantique distributionnelle capturée par l'apprentissage de représentations et la sémantique relationnelle exprimée dans les ressources externes pour effectuer des tâches orientées domaine.

Cette affirmation est contrastée par l'observation des résultats de notre modèle sur les tâches de TALN (cf. Tableau 4.4). En effet, en ce qui concerne les différents modèles de référence, nos modèles obtiennent généralement de meilleurs résultats pour le jeu de données génériques Robust tandis qu'ils obtiennent des résultats inférieurs ou équivalents pour les jeux de données médicales.
Par exemple, pour la tâche MRPC sur Robust, nos scénarios obtiennent des précisions de $71,90\%$ à $74,26\%$, tandis que les précisions des modèles de référence varient de $68,05\%$ à $70,81\%$. Pourtant, pour la même tâche sur OHSUMED, notre meilleur scénario ($SD2V_{on}$) obtient seulement $29,96\%$ de précision, par rapport

Modèles		Robust					OHSUMED					TREC Med				
		SUBJ	MPQA	TREC	MRPC	STS2014	SUBJ	MPQA	TREC	MRPC	STS2014	SUBJ	MPQA	TREC	MRPC	STS2014
	$TF-IDF$	72,13	68,45	79,98	69,12	42,84	33,13	25,35	31,48	30,32	37,84	22,55	21,99	21,48	19,75	26,92
	AWE	73,10	68,04	79,52	68,05	44,77	32,50	25,74	32,12	29,35	34,77	21,92	21,87	20,51	19,25	25,11
	AWE_R	75,71	69,08	81,91	68,75	45,17	35,61	26,63	34,18	31,20	36,66	22,63	22,45	21,24	20,60	26,03
	$D2V$	73,52	69,35	79,30	70,81	42,56	31,56	25,01	32,07	28,23	33,76	21,55	21,58	20,68	18,56	25,60
Non pre.	$SD2V_{off}$	76,15	72,11	79,50	71,90	44,76	32,65	25,40	32,15	27,88	33,32	22,00	21,18	20,92	18,21	26,06
	$SD2V_{on}$	75,44	70,89	79,56	72,04	44,69	32,99	25,53	32,57	28,80	33,70	21,81	21,38	21,00	18,15	25,83
Pre. w2v	$SD2V_{off}$	76,36	71,81	79,89	72,44	45,36	33,03	25,59	32,00	28,66	34,06	22,04	21,23	20,28	18,45	26,28
	$SD2V_{on}$	76,66	70,68	81,60	74,26	45,30	34,17	26,67	33,64	29,96	34,64	23,11	22,20	21,14	19,32	25,64

Tableau 4.4 – Comparaison des approches d'apprentissage hors ligne/en ligne sur les tâches de similarité et de classification (SentEval)

aux valeurs qui varient de 28,23% à 31,32% des modèles de référence. Ces énoncés soulignent que :

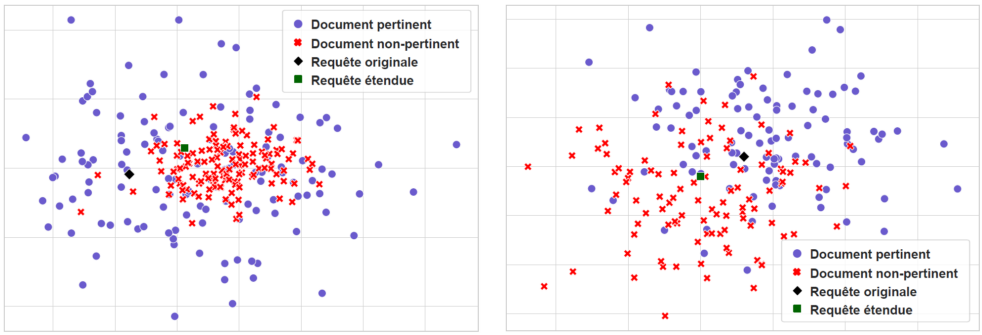
1. Les jeux de données génériques sont mieux adaptés aux tâches qui raisonnent à un niveau global comme les tâches d'appariement sémantique de TALN. D'ailleurs, la thématique des jeux de test dans les tâches de TALN sont souvent dans le domaine générique. Cela est cohérent avec les meilleures performances des représentations distribuées entraînées sur Robust.
2. Le détail technique du domaine médical exige des modèles de représentations qui sont capables de capturer la sémantique à plusieurs niveaux de granularité (comme suggéré dans notre modèle en ligne), qui sont plus efficaces pour les tâches de RI.

Il convient de mentionner que nous explorons également l'impact de l'utilisation des représentations de mots pré-entraînées pour l'initialisation de notre modèle. Bien qu'il n'y ait pas de différences significatives pour les tâches de RI, la représentation de mots pré-entraînée semble aider notre modèle à apprendre de meilleures représentations de documents concernant les tâches de TALN. En effet, dans les tâche de RI (cf. Tableau 4.3), pour un même modèle (hors ligne ou en ligne), le scénario pré-entraîné n'apporte pas en général des améliorations par rapport au scénario non pré-entraîné, la plus grande amélioration est de 0,9% pour le modèle $SD2V_{on}$ dans la tâche de réordonnancement sur TREC Med. Tandis que sur les tâches de TALN, on peut observer dans la plupart des cas, les scénarios pré-entraînés de nos modèles obtiennent de meilleurs résultats par rapport à ceux non pré-entraîné et la plus grande amélioration est à 6% pour le modèle $SD2V_{on}$ dans la tâche SUBJ sur TREC Med. En comparant l'appariement de similarité (TALN) et l'appariement de pertinence (RI), on voit ainsi que l'injection de représentation de

mots pré-entraînée, principalement conçue pour capturer les similarités de mots, dans nos modèles permet d'améliorer les signaux de similarité qui sont traités dans les tâches TALN.

• En observant les taux d'accroissement sur les jeux de données médicales, les résultats indiquent que les taux d'accroissement des approches basées sur la représentation (modèles de référence et nos modèles) sont plus élevés pour la tâche de réordonnancement que pour l'expansion de la requête (cf. Tableau 4.3). Par exemple, sur le jeu de donnée Ohsumed, ce taux est autour de 40% pour la tâche de réordonnancement, alors qu'il est inférieur à 21,05% sur la tâche d'expansion de requête. Comme nous avons déjà mentionné, la tâche d'expansion de requête consiste à étendre la requête en fonction de la similarité requête-mots et requête-concepts qui ne sont pas directement corrélées avec des signaux de pertinence. En revanche, la tâche de réordonnancement est davantage orientée vers les signaux de pertinence. Pour illustrer cette affirmation, nous proposons dans la Figure 4.9 de visualiser les représentations de documents et de requêtes à travers une analyse t-SNE. Étant donné un sujet TREC particulier du jeu de données OHSUMED, nous distinguons : (1) les documents pertinents (points violets) par rapport aux documents non pertinents (croix rouges) ; et (2) la requête originale (diamant noir) utilisée dans la tâche de réordonnancement par rapport à la requête étendue (carré vert) utilisée dans la tâche d'expansion de requête.
La requête étendue est obtenue par notre modèle en ligne à l'aide de la représentation de mots pré-entraînée.

Les figures illustrent que 1) notre modèle est capable de construire des représentations de documents discriminant les documents pertinents et non pertinents ; et 2) que la requête originale est plus proche dans la projection des documents pertinents que la requête étendue.



(a) QID 7 (b) QID 35

Figure 4.9 – Représentations TSNE de documents pertinents et non pertinents pour une requête originale et sa version étendue.

7.1.2 *Evaluation comparative des modèles hors ligne vs. en ligne*

	Modèles	Robust	OHSUMED	TREC Med
	$TF - IDF$	8,80	**9,52**	**8,57**
	AWE	15,20	61,90	44,00
	AWE_R	14,00	55,56	36,00
	$D2V$	6,80	31,75	20,00
Non pré-entraîné	$SD2V_{off}$	**5,60**	30,16	22,71
	$SD2V_{on}$	10,40	30,16	25,71
Pré-entraîné w2v	$SD2V_{off}$	8,40	28,57	17,14
	$SD2V_{on}$	10,40	30,16	22,86

Tableau 4.5 – Comparaison des modèles sur la tâche de similarité des documents (taux d'erreur de classification).

Nous présentons ici les résultats obtenus sur le tâche similarité des documents (cf. Tableau 4.5). Pour les tâches de TALN présentées (cf. Tableaux 4.4 et 4.5), on peut distinguer deux grandes catégories : 1) les tâches de similarité mesurant la proximité entre les représentations (à savoir la similarité des documents et la tâche STS 2014 dans SentEval) et 2) les tâches de classification apprenant un modèle prédictif supplémentaire sur la base des représentations de documents (à savoir SUBJ, MPQA, TREC, et MRPC dans SentEval).

Si nous comparons les modèles hors ligne et en ligne selon les catégories de tâches, nous pouvons observer que le modèle hors ligne obtient généralement de meilleurs résultats pour les tâches de similarité. En effet, pour la tâche similarité des documents (cf. Tableau 4.5), le modèle hors ligne obtient les taux d'erreur les plus faibles (e. g., pour le jeu de données Robust, 5.60% et 8.40% selon que le modèle est pré-entraîné ou non) en comparant au modèle en ligne qui atteint un taux d'erreur plus élévé (e. g., 10.40% et plus pour le jeu de données Robust). Cette tendance est également observée sur la tâche STS 2014 (cf. Tableau 4.4) pour les jeux de données Robust et TREC Med.

En ce qui concerne les tâches de classification (cf. Tableau 4.4), le modèle d'apprentissage en ligne de la représentation de documents semble plus efficace puisqu'il surpasse le modèle hors ligne pour 3/5 tâches pour le jeu de données Robust, 5/5 pour OHSUMED, et 3/5 pour TREC Med. Cette différence pourrait s'expliquer par le fait que les tâches de similarité utilisent directement l'espace de représentations appris pour estimer les similarités des documents alors que les tâches de classification nécessitent une couche supplémentaire basée sur un modèle de classification. Cela suggère que la méthode hors ligne est suffisante pour détermi-

ner si deux documents traitent du même sujet, mais ne permet pas de saisir des indicateurs de granularité plus fins de la sémantique des documents individuels, au moins pas assez pour permettre un raisonnement sur les textes. En effet, la méthode d'apprentissage hors ligne apprend séparément deux représentations de documents (l'une basée sur des mots et l'autre sur des concepts) avant de rapprocher ces représentations de documents pour construire une nouvelle représentation de documents sans tenir compte de son contexte mot/concept.

En revanche, la méthode en ligne intègre des mots, des concepts et des documents dans le même espace de représentations et, par conséquent, apprend à intégrer des documents en exploitant leur contexte de mot/concept.
Cette différence constitue une première raison qui explique pourquoi le modèle en ligne est plus efficace pour les tâches de classification. En effet, ces dernières sont plus complexes et exigent plus d'inférence sur la sémantique du document que la simple comparaison du sujet des deux documents.
La deuxième raison tient au fait que l'apprentissage conjoint des représentations de mots, de concepts et de documents dans les modèles en ligne permet d'obtenir un avantage mutuel pour la qualité de la représentation de ces différents niveaux de granularité. En effet, ces différents niveaux de granularité sont projetés dans le même espace de représentation qui est réajusté tout au long du processus d'apprentissage. Cela aide à affiner les représentations des mots et des concepts qui peuvent directement améliorer la qualité de la représentation du document ; et inversement, le niveau du document fournit un niveau plus élevé de sémantique qui peut améliorer les représentations des mots et des concepts. En revanche, dans la méthode hors ligne, les représentations de mots et de concepts sont ignorées lors de la construction de la représentation finale du document dans un espace séparé.

Pour mieux comprendre ce phénomène, nous évaluons les représentations des mots et des concepts par les tâches similarité des mots et les tâches de similarité des concepts. Les résultats présentés dans le Tableau 4.6 montrent que le modèle en ligne surpasse généralement le modèle hors ligne pour la similarité des mots et obtient toujours de meilleurs résultats pour la tâche de similarité des concepts. Par exemple, le modèle en ligne obtient toujours le meilleur résultat sur la tâche SIMLEX999 par rapport au modèle hors ligne pour tous les jeux de données (sur Robust, OHSUMED, TREC Med avec les valeurs respectivement : $0,35$ vs. $0,31$; $0,21$ vs. $0,19$ et $0,13$ vs. $0,11$).
Ceci confirme notre intuition que les représentations conjointes des différents niveaux de granularité (mots, concepts et documents) dans le même espace de représentations conduit à un bénéfice mutuel dans le processus d'apprentissage et permet également de construire de meilleures représentations pour les mots et concepts.

	Similarité des mots												Similarité des conceps	
	Robust				OHSUMED				TREC Med				OHSUMED	TREC Med
	MEN	RG65	SIMLEX	WS353	MEN	RG65	SIMLEX	WS353	MEN	RG65	SIMLEX	WS353		
$SD2V_{off}$	**0,48**	0,42	0,31	**0,35**	**0,35**	0,29	0,19	**0,41**	**0,24**	0,20	0,11	0,30	0,386	0,223
$SD2V_{on}$	0,45	**0,43**	**0,35**	0,34	**0,35**	0,28	**0,21**	**0,41**	0,23	**0,24**	**0,13**	**0,37**	**0,413**	**0,301**

Tableau 4.6 – Comparaison de l'efficacité des approches en ligne et hors ligne sur les tâches de similarité de mots et de concepts (corrélation de Spearman ρ).

En ce qui concerne les tâches de RI (cf. Tableau 4.3), les différences entre les modèles hors ligne et en ligne sont généralement très faibles. Le taux d'accroissement moyen en terme de MAP des scénarios hors ligne par rapport aux scénarios en ligne est à $0,18\%$ sur la tâche de réordonnancement et à $3,83\%$ sur la tâche d'expansion de requête (calculé sur tous les jeux de données). Cette absence de différences pour les tâches de RI par rapport aux énoncés précédents pour les tâches de TALN suggère que nos deux modèles se comportent de la même façon pour ce qui est de saisir la notion de signaux de pertinence. En effet, comme suggéré par Guo et al. (2016), les tâches de RI impliquent la prise en compte des signaux d'appariement exacts avec la requête relative aux informations à grain fin (appariement exact des termes, pondération des termes, etc.) alors que les tâches TALN sont plus orientées vers un appariement sémantique à un niveau global.

La seule différence entre les modèles hors ligne et en ligne qui pourrait être perçue concerne le jeu de données TREC Med dans la tâche expansion de la requête. Nous pouvons observer que les bases de référence d'apprentissage de la représentation AWE, AWE_R et $D2V$ ne parviennent pas à dépasser la référence de le BM25. En effet, les valeurs de MAP obtenues par ces modèles de référence sont inférieures à $0,291$ tandis que le modèle BM25 obtient un résultat à $0,312$.
La tâche de RI sur ce jeu de données liée au jeu de données consiste en une recherche clinique de cohortes. Cette tâche est particulièrement complexe en raison de la spécificité du domaine d'application et de la technicité de la tâche médicale. Les requêtes sont généralement verbeuses avec des groupes verbaux avec des dépendances. Par conséquent, la tâche d'expansion de requête conduirait à construire des requêtes plus verbeuses, ce qui augmenterait la probabilité d'introduire du bruit et d'éventuelles interactions de mots inutiles dans la requête.
Le Tableau 4.3 montre que notre modèle hors ligne (qu'il soit pré-entraîné ou non) est le seul scénario qui permet d'améliorer le modèle de référence BM25 ($0,3205$ et $0,3227$ pour nos modèles hors ligne vs. $0,312$ pour BM25). Notre hypothèse est que le fait de séparer l'espace de représentations basé sur les mots et l'espace de représentations basé sur le concept dans le modèle hors ligne évite certaines interactions bruyantes de mots-concepts qui pourraient être introduites dans le modèle en ligne qui apprennent conjointement le représentations de mots, de concepts et de documents dans le même espace.

Pour illustrer cette affirmation, nous présentons dans le Tableau 4.7 des exemples de requêtes étendues pour les modèles en ligne et hors ligne. La tendance étant similaire entre les scénarios pré-entraînés ou non, nous ne considérons que le scénario pré-entraîné où l'écart entre les deux modèles est plus grand. Par exemple, la deuxième requête portant sur le cancer est étendue avec les termes "*eval*" et "*mesiotemporal*" dans le modèle hors ligne. Ce dernier terme fait référence aux symptômes de l'épilepsie qui sont corrélés aux métastases. Ces dernières sont des groupes de cellules cancéreuses ayant migré à partir d'un foyer d'origine. En revanche, le modèle en ligne étend la requête plutôt avec des mots/concepts basés sur les médicaments, qui sont liés au cancer, mais pas particulièrement aux examens médicaux mentionnés dans la requête. Cet exemple montre que des interactions entre le cancer et les médicaments ont été détectées dans le modèle en ligne, mais qu'elles ne sont pas nécessairement pertinentes pour identifier les bons termes/concepts candidats pour l'expansion de la requête.

Requête originale	Modèles	Termes ajoutés
patients with hearing loss	$SD2V_{off}$	vbk hyperchol
	$SD2V_{on}$	frontotemporal care
patients who had positron emission tomography (pet), magnetic resonance imaging (mri), or computed tomography (ct) for staging or monitoring of cancer	$SD2V_{off}$	eval mesiotemporal
	$SD2V_{on}$	nitropaste medication

Tableau 4.7 – Exemples de requêtes étendues pour les modèles hors ligne et en ligne

7.2 *Evaluation des modèles avec contrainte de relations*

Dans la section précédente, nous avons comparé l'efficacité de nos deux approches d'apprentissage. Ces dernières apprennent la représentation de documents en utilisant les mots et/ou les concepts comme l'information contextuelle. Nous proposons d'évaluer à ce niveau l'intérêt d'intégrer les contraintes relationnelles dans nos modèles d'apprentissage de la représentation de documents, et plus particulièrement d'évaluer comment elles devraient être intégrées. Pour ce faire, nous comparons les deux façons d'intégrer les contraintes de relation décrites à la section 5 :

1. en enrichissant les instances d'apprentissage (noté $SD2V_{Ins_{off}}$ et $SD2V_{Ins_{on}}$ dans nos expériences)
2. en utilisant une fonction de régularisation dans l'objectif d'apprentissage (notée $SD2V_{Reg_{off}}$ et $SD2V_{Reg_{on}}$ dans nos expériences)

Nous présentons dans les tableaux 4.8 , 4.9 et 4.10 les résultats de diverses tâches TALN et RI obtenus à partir des scénarios dans lesquels nos modèles sont initia-

lisés de façon aléatoire (non pré-entraînés). Ces scénarios évitent toute prise en compte sémantique préalable dans le processus d'apprentissage ; cependant, des expériences préliminaires ont montré que des tendances similaires pourraient être obtenues avec le scénarios pré-entraînés.

Modèles	Robust	OHSUMED	TREC Med
$TF-IDF$	8,80	**9,52**	**8,57**
AWE	15,20	61,90	44,00
AWE_R	14,00	55,56	36,00
$D2V$	6,80	31,75	20,00
$SD2V_{off}$	**5,60**	30,16	22,71
$SD2V_{Reg_{off}}$	6,80	18,57	17,14
$SD2V_{Ins_{off}}$	**5,60**	28,57	17,14
$SD2V_{on}$	10,40	30,16	25,71
$SD2V_{Reg_{on}}$	7,60	31,74	20,00
$SD2V_{Ins_{on}}$	10,00	28,57	22,86

Tableau 4.8 – Comparaison des approches d'intégration des relations sur la tâche de similarité des documents.

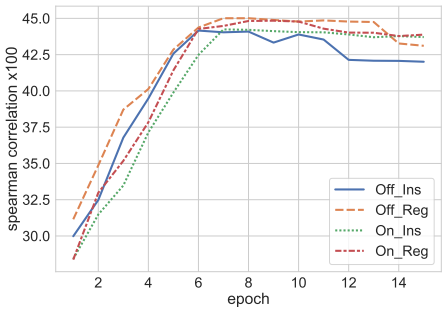
Modèles	Robust					OHSUMED					TREC Med				
	SUBJ	MPQA	TREC	MRPC	STS 2014	SUBJ	MPQA	TREC	MRPC	STS 2014	SUBJ	MPQA	TREC	MRPC	STS 2014
$SD2V_{off}$	76,15	72,11	79,50	71,90	44,76	32,65	25,40	32,15	27,88	33,32	22,00	21,18	20,92	18,21	26,06
$SD2V_{Reg_{off}}$	77,02	72,78	80,89	73,37	45,01	33,83	26,34	33,12	28,77	34,22	23,02	21,99	22,48	18,93	27,70
$SD2V_{Ins_{off}}$	76,41	72,00	80,20	73,77	44,16	34,37	27,42	34,18	29,31	34,49	22,94	20,94	21,32	19,44	27,09
$SD2V_{on}$	75,44	70,89	79,56	72,04	44,69	32,99	25,53	32,57	28,80	33,70	21,81	21,38	21,00	18,15	25,83
$SD2V_{Reg_{on}}$	75,69	71,09	80,01	73,62	44,84	34,33	27,28	33,21	29,44	34,25	22,78	22,74	22,03	19,61	26,24
$SD2V_{Ins_{on}}$	76,78	71,13	79,43	72,27	44,24	33,31	26,08	32,76	28,96	33,95	22,40	22,05	20,34	19,28	27,29

Tableau 4.9 – Comparaison des approches d'intégration des relations sur les tâches de similarité et de classification (benchmark SentEval)

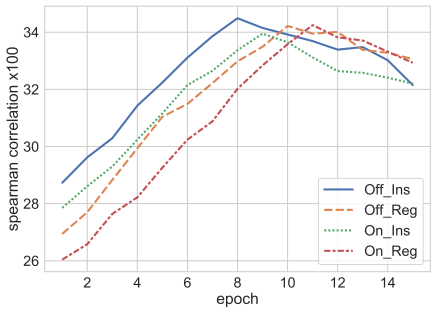
Pour les tâches de TALN (cf. Tableaux 4.8 et 4.9), la comparaison des deux approches intégrant les contraintes relationnelles souligne que la méthode basée sur la régularisation donne généralement de meilleurs résultats pour les modèles d'apprentissage hors ligne et en ligne. Par exemple, dans la tâche de similarité des documents (cf. Tableau 4.8), le scénario $SD2V_{Reg_{on}}$ obtient des taux d'erreur plus petits que le scénario $SD2V_{Reg_{off}}$ sur les jeux de donnée Robust et TREC Med : 7,6% vs. 10% et 20% vs. 22,86%, respectivement. En particulier, sur tous les tâches/jeux de données/modèles d'apprentissage, les scénarios de régularisation dépassent ceux d'instance 20/30 tâches d'évaluation du TALN (cf. Tableau 4.9).
Cela pourrait s'expliquer par le fait que la méthode de régularisation ajuste directement les représentations des mots/concepts reliés pour qu'elles soient proches dans l'espace latent alors que la mise en place des instances d'apprentissage four-

nit simplement une information contextuelle augmentée dans le processus d'apprentissage.

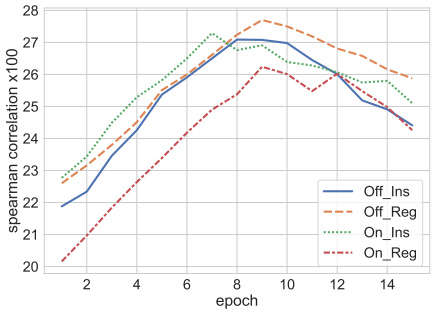
Par ailleurs, nous analysons la relation entre l'efficacité de la tâche et le nombre d'époques du processus d'apprentissage. La Figure 4.10 illustre les scores de corrélation pour la tâche STS 2014 estimés à différentes époques du processus d'apprentissage.
Si l'on compare le scénario de régularisation avec le scénario d'instance d'apprentissage, on constate que le scénario s'appuyant sur les instances d'apprentissage, bien que moins efficace que la régularisation, atteint plus rapidement leur efficacité maximale. Cela suggère que l'information contextuelle améliorée fournie par les instances d'apprentissage stimule les interactions entre les différents niveaux de granularité (mots, concepts et documents) dans le processus d'apprentissage. Cependant, ces interactions ne sont pas suffisantes pour contraindre l'espace de représentation.



(a) Collection Robust



(b) Collection OHSUMED



(c) Collection TREC Med

Figure 4.10 – Représentations TSNE de documents pertinents et non pertinents pour une requête originale et sa version étendue.

Modèle	Réordonnancement			Expansion de requête		
	Robust	OHSUMED	TREC Med	Robust	OHSUMED	TREC Med
$SD2V_{off}$	0,2510	0,3018	0,3591	0,2464	0,2580	0,3205
$SD2V_{Reg_{off}}$	0,2509	0,3026	0,3558	0,2458	0,2573	0,3224
$SD2V_{Ins_{off}}$	0,2511	0,3018	0,3591	0,2464	0,2580	0,3205
$SD2V_{on}$	0,2507	0,3020	0,3554	0,2443	0,2599	0,2889
$SD2V_{Reg_{on}}$	0,2517	0,3010	0,3582	0,2449	0,2598	0,2908
$SD2V_{Ins_{on}}$	0,2511	0,3036	0,3580	0,2402	0,2489	0,2823

Tableau 4.10 – Comparaison des approches d'intégration de relations sur les tâches d'évaluation de RI : réordonnancement et expansion de la requête.

Comme pour l'analyse précédente concernant les tâches de TALN, le tableau 4.10 souligne que les différences entre les deux façons d'intégrer les relations sont très faibles pour les tâches de RI. Le taux d'accroissement moyen en terme de MAP des scénarios en ligne par rapport aux scénarios hors ligne est à $-0,2\%$ sur la tâche de réordonnancement et à $1,6\%$ sur la tâche d'expansion de requête (calculé sur tous les jeux de données). Cependant, en accord avec les observations précédentes sur les tâches de TALN, nous pouvons observer que dans $7/12$ cas, les scénarios de régularisation conduisent à de légères améliorations, avec un taux d'accroissement maximal à $4,38\%$ (pour le modèle en ligne dans la tâche d'expansion de requête sur OHSUMED).

Par contre, le scénario d'instance d'apprentissage obtient les mêmes résultats que le scénario de base (sans contrainte) dans $5/12$ cas ; ce qui suggère que les instances d'apprentissage, même si elles permettent d'incorporer en avance les signaux correspondants comme indiqué dans les TALN (Figure 4.10), ne parviennent pas à créer des représentations efficaces pour capturer des signaux de pertinence supplémentaires dans les tâches de RI.

Comme le scénario de régularisation semble plus favorable, nous proposons dans ce qui suit d'explorer les différentes composantes intégrées à la fonction de régularisation. En effet, dans le scénario présenté, nous considérons de manière équivalente la régularisation sur le terme et le concept (α_W et α_C sont tous deux réglés à 1, cf. Equation 4.15).

Pour mieux comprendre l'impact de chaque composante de granularité, nous présentons dans le Tableau 4.11 différents scénarios de régularisation, à savoir contrainte sur les termes ($\alpha_W = 1$ et $\alpha_C = 0$, conduisant à ne considérer que Reg_W) ou contrainte sur les concepts ($\alpha_W = 0$ et $\alpha_C = 1$, conduisant à considérer uniquement Reg_C), ou contrainte sur les deux ($\alpha_W = 1$ et $\alpha_C = 1$, menant au scénario Reg_W, Reg_C). Les résultats montrent que la régularisation des représentations sur

les termes n'est pas puissante et que la régularisation sur les concepts est nécessaire. Dans la plupart des cas (11/12), les scénarios Reg_W ont une valeur MAP inférieure à celle des scénarios Reg_C ou Reg_W, Reg_C. Cela suggère qu'au-delà de l'utilisation des ressources sémantiques pour contraindre les représentations textuelles comme dans Faruqui et al. (2015), elles semblent être plus efficaces lorsque leurs sources d'évidence (termes, concepts, relations) sont tous intégrées pour la contrainte des représentations de documents.

	Modèles	Robust	OHSUMED	TREC Med
Réordonnancement	$SD2V_{Reg_{off}}(Reg_W)$	0,2501	0,3025	0,355
	$SD2V_{Reg_{off}}(Reg_C)$	0,2510	0,3029	0,3557
	$SD2V_{Reg_{off}}(Reg_W, Reg_C)$	0,2509	0,3026	0,3558
	$SD2V_{Reg_{on}}(Reg_W)$	0,2516	0,3012	0,3588
	$SD2V_{Reg_{on}}(Reg_C)$	0,2517	0,3015	0,3580
	$SD2V_{Reg_{on}}(Reg_W, Reg_C)$	0,2517	0,3010	0,3591
Expansion de requête	$SD2V_{Reg_{off}}(Reg_W)$	0,2457	0,2570	0,3220
	$SD2V_{Reg_{off}}(Reg_C)$	0,2460	0,2580	0,3219
	$SD2V_{Reg_{off}}(Reg_W, Reg_C)$	0,2458	0,2573	0,3224
	$SD2V_{Reg_{on}}(Reg_W)$	0,2433	0,2590	0,291
	$SD2V_{Reg_{on}}(Reg_C)$	0,2449	0,2599	0,2909
	$SD2V_{Reg_{on}}(Reg_W, Reg_C)$	0,2449	0,2598	0,2908

Tableau 4.11 – Impact des niveaux de granularité (mots vs. concepts) intégrés dans la contrainte relationnelle modélisée dans l'objectif d'apprentissage.

D'un point de vue général, si les résultats de la tâche RI ne mettent pas en évidence de différences significatives entre le scénario de base de nos approches (sans contrainte) et la version intégrée des contraintes relationnelles, on peut trouver que les relations linguistiques permettent de mieux intégrer les tâches de TALN (cf. Tableaux 4.8 et 4.9).

Concernant la tâche de similarité des documents (cf. Tableau 4.8), on observe généralement une diminution du taux d'erreur en tenant compte des contraintes relationnelles. Par exemple, pour le jeu de donnée TREC Med, le taux d'erreur est diminué de 22, 17% à 17, 14% et de 25, 71% à 20% quand on applique la contrainte par régularisation *Reg* sur les modèle hors ligne et en ligne, respectivement. Cet avantage est particulièrement remarqué pour les jeux de données médicales OHSUMED et TREC Med, qui impliquent la nécessité de disposer d'un vocabulaire technique spécifique et de connaissances supplémentaires pour l'intégration précise de documents dans des domaines d'application.
Bien que cette source d'évidence permette d'améliorer l'écart entre les méthodes

standards de similarité des documents (AWE et $D2V$), il n'est pas suffisant d'atteindre la représentation du document basée sur le modèle de pondération statistique de terme $TF - IDF$. Sur les jeux de données médicaux, le modèle de référence $TF - IDF$ obtient le taux d'erreur plus faible à $9,52\%$ et $8,57\%$ (sur OHSUMED et TREC Med respectivement).

Il convient de mentionner que cette mesure de similarité des documents est basée sur les ordonnancement RI pour construire des triplets de documents à comparés, par opposition à d'autres tâches du TALN (similarité et classification). Ainsi, cette similarité est basée sur l'appariement exact des termes, expliquant pourquoi la méthode $TF - IDF$ donne de bons résultats dans cette tâche de similarité pilotée RI. Pour le jeu de données Robust, l'amélioration est moins évidente car la version sans relations fonctionne relativement bien.

Pour les tâches de proximité relationnelle et de classification, on peut observer que les deux méthodes d'intégration des contraintes relationnelles basées soit sur la régularisation, soit sur des instances d'apprentissage (respectivement notées $SD2V_{Reg.}$ et $SD2V_{Ins.}$) dépassent généralement le scénario correspondant en intégrant seulement les concepts (notées $SD2V.$). Par exemple, le modèle hors ligne $SD2V_{off}$ entraîné sur le jeu de données Robust obtient une précision de $79,50$ pour le benchmark MRPC tandis que les modèles intégrés de relation $SD2V_{Reg_{off}}$ et $SD2V_{Ins_{off}}$ atteignent respectivement $80,89$ et $80,20$ de précision.

8 Bilan

Nous avons présenté dans ce chapitre les contributions pour l'apprentissage des représentations distribuées de textes en intégrant la sémantique relationnelle issue des ressources. Dans un premier temps, deux méthodes d'apprentissage de représentations de documents sont proposées, une de type "hors ligne" et une de type "en ligne". Notre approche "hors ligne" consiste à apprendre une représentation de documents dite "optimale" qui combine deux espaces de représentations pré-entraînées, un basé sur les mots, un basé sur les concepts. La deuxième approche dite "en ligne", consiste en un apprentissage conjoint des représentations des textes à plusieurs niveaux (i. e., document, mot, concept). Dans un second temps nous avons intégré dans l'apprentissage de ces modèles des contraintes relationnelles afin d'améliorer les représentations de documents. Cette intégration s'est faite grâce à deux méthodes différentes, une basée sur la régularisation de la fonction objectif, une autre basée sur les instances d'apprentissage en entrée. Pour chaque approche, nous intégrons dans chacun des deux modèles d'apprentis-

sage "hors ligne" et "en ligne" pour évaluer l'apport de la sémantique relationnelle.

Plusieurs analyses expérimentales sont menées pour comparer nos propositions sur plusieurs aspects, à savoir la qualité des représentions entraînées, l'apport de la contrainte relationnelle, et l'efficacité en RI avec nos représentations de textes à plusieurs niveaux. Plus précisément, nous avons analysé la qualité de nos représentations de documents sur les tâches TALN. Nous examinons également l'apport de nos représentations sur les tâches de RI, à savoir l'expansion de la requête et le réordonnancement. Les résultats sur les tâches montrent que nos propositions permettent en général d'améliorer les performances sur les tâches d'évaluation.
En effet, nos modèles de base qui intègrent les concepts dans l'apprentissage (sans contrainte de relation) donnent des meilleurs résultats par rapport aux modèles de référence sur les tâches de TALN et RI. Nos modèles obtiennent particulièrement des résultats plus élevés pour les jeux de données médicales que pour le domaine générique en RI.
En analysant l'apport de la contrainte de relation, les deux modèles avec l'intégration des contraintes relationnelles dépassent généralement le scénario correspondant en intégrant seulement les concepts. Cette amélioration est plus importante sur les tâches de TALN que sur la RI. Entre les deux méthodes d'intégration de contrainte relationnelle, la méthode basée sur la régularisation donne généralement de meilleurs résultats, par rapport à la méthode basée sur l'instance, pour nos deux modèles d'apprentissage hors ligne et en ligne.

5

MODÈLE NEURONAL POUR LA RI

Introduction

Dans le chapitre 3, nous avons présenté un état-de-l'art sur les réseaux de neurones profonds pour la recherche d'information (RI). Ces modèles exploitent la sémantique distributionnelle du texte via des architectures neuronales pour apprendre à apparier deux textes. Il existe deux familles de modèles : 1) les modèles basés sur la représentation qui apprennent la fonction d'appariement entre la requête et le document dans un réseau siamois ; et 2) les modèles basés sur les interactions qui modélisent dès la couche d'entrée les indicateurs d'appariement qui sont ensuite exploités dans un réseau à une seule branche.

Nous présentons dans ce chapitre notre contribution d'un modèle neuronal pour la RI. Notre modèle se situe dans la famille des modèles basés sur la représentation des textes. Notre objectif est de réduire le fossé sémantique entre deux textes en exploitant la sémantique relationnelle issue des ressources terminologiques afin de l'introduire dans le processus d'apprentissage d'un réseau de neurones. Plus précisément, nous cherchons à modéliser une *représentation symbolique* du texte à travers ses objets (entités/concepts) identifiés grâce à une ressource sémantique. Cette représentation prend en compte également les relations entre les objets, prédéfinies par la ressource, pour rapprocher les représentations de deux textes similaires.

Ce chapitre est organisé comme suit. La section 1 présente les problématiques et les motivations principales de notre contribution. La section 2 décrit notre modèle neuronal pour l'appariement de texte ainsi que la modélisation de la représentation symbolique. Une évaluation expérimentale est présentée et discutée dans la section 3. La section 4 conclut ce chapitre.

1 Contexte et motivations

Les approches traditionnelles de RI s'appuient fondamentalement sur le principe de "sac de mots" où l'appariement entre une requête et un document est basé sur un rapprochement lexical entre les mots qui les composent. L'une des limites de ces approches repose sur la difficulté de capter la sémantique des mots en raison de leur variation lexicale (e.g. acronymes, homonymes, synonymes, etc.) qui s'ajoute à l'ambiguïté du besoin en information caché derrière une requête, généralement composée de peu de mots (Xu and Croft, 2000; Koopman et al., 2016). Ce problème, connu sous le nom de *fossé sémantique*, empêche intrinsèquement l'appariement entre une requête et un document, qui est l'étape cruciale pour sélectionner les documents pertinents candidats en réponse à la requête d'un utilisateur (Edinger et al., 2012). Pour rappel, le fossé sémantique provient généralement de : 1) la *discordance de vocabulaire*, ce qui signifie que des mots de formes différentes partagent le même sens (e. g., **aperçu** est synonyme de **sommaire**) ; 2) la *discordance de granularité*, ce qui signifie que des mots de formes et de sens différents appartiennent au même concept général (e. g., **chat** et **chien** sont des **animaux**) ; 3) la *polysémie*, ce qui signifie qu'un mot peut couvrir différents sens en fonction des mots qui l'entourent dans le texte et qui représentent son contexte (e. g., **pêche** peut signifier un fruit ou l'action de pêcher).

Pour réduire le fossé sémantique, les approches principales en RI se concentrent sur l'amélioration de la représentation des requêtes et/ou des documents en utilisant des connaissances explicites fournies par des sources de connaissances externes ou des connaissances implicites déduites de corpus de textes. En ce qui concerne les connaissances explicites, nous avons présenté dans le chapitre 2 une lignée de travaux des modèles de RI sémantique basée sur le principe de la *sémantique relationnelle* en exploitant des ressources sémantiques. Ces ressources fournissent des informations sur les sens des mots sous la forme d'entités (ou concepts) et de relations sémantiques entre ces entités. Ces dernières peuvent être classées en deux grandes catégories : (1) celles qui représentent des connaissances linguistiques, soit générales (e.g. WordNet) (Pal et al., 2014; Navigli and Velardi, 2003; Baziz et al., 2003), soit basées sur un domaine particulier (e.g. UMLS) (Hersh et al., 2000) et (2) celles structurées comme des graphes de connaissances (e.g. DBpedia, Freebase), représentant des informations factuelles sur les entités et les relations sémantiques entre ces entités (Fu et al., 2005; Xiong and Callan, 2015b). En RI, ces ressources fournissent des sources d'évidence supplémentaires sur les objets et leurs relations (e.g. les relations de synonymie ou de hyponymie) permettant de réduire la discordance de vocabulaire entre les requêtes et les documents grâce à diverses techniques : l'expansion de la requête (Xiong and Callan, 2015b; Pal et al., 2014; Navigli and Velardi, 2003), l'expansion de document (Agirre et al.,

2010; Gobeill et al., 2008; Chahine et al., 2011; Gupta et al., 2017), ou plus récemment, l'utilisation des facteurs enrichis décrivant la relation entre les documents et les requêtes (Xiong and Callan, 2015a).

Récemment, des travaux se sont intéressés aux connaissances implicites qui pourraient être extraites du texte brut. Ces approches, présentées dans le chapitre 3 portent sur l'apprentissage de représentations des documents et des requêtes pour en faire émerger les facteurs d'associations latentes. Ces modèles sont capables d'inférer les sens des mots par association à d'autres mots en analysant leurs co-occurrences dans le corpus de documents (Mikolov et al., 2013b; Pennington et al., 2014). Egalement, l'apprentissage de représentation a été exploité dans les réseaux de neurones profonds guidés par une tâche de RI (approches de bouts-en-bouts) permettant d'apprendre de façon conjointe la fonction d'appariement ainsi que la sémantique latente du texte brut (Guo et al., 2016; Huang et al., 2013; Severyn and Moschitti, 2015). L'une des premières contributions dans ce domaine s'appuie sur des architectures siamoises, opposant la requête et le document dans un réseau à deux branches (Huang et al., 2013; Severyn and Moschitti, 2015). Cependant, ces précédents travaux présentent certaines limites :

1. L'apprentissage de représentation est généralement réalisé sur le titre du document soulignant la difficulté liée au traitement du texte intégral pouvant générer des signaux bruités lors de l'apprentissage (Guo et al., 2016).
2. L'apprentissage est réalisé sur des grands jeux de données, pour la plupart, issus de moteurs de recherche commerciaux/propriétaires, incluant un très grand nombre de requêtes. Ainsi, l'utilisation de collections d'évaluation standard (e.g., collections TREC) où le nombre de requêtes est limité présente un réel enjeu pour la communauté.
3. Certains travaux (Iacobacci et al., 2015) démontrent que les approches neuronales permettant de capturer la sémantique latente des textes ne sont pas capables de résoudre le problème de polysémie.

Ces différentes problématiques expliquent ainsi un constat récurrent dans les précédent travaux : il est difficile d'atteindre une performance équivalente ou nettement supérieure aux modèles de RI traditionnels (e. g., BM25, modèles de langue) (Guo et al., 2016; Huang et al., 2013; Hu et al., 2014).

Guidés par les mérites et limites des deux grandes lignées de travaux énoncées précédemment, nous proposons dans cette contribution d'exploiter à la fois des approches d'apprentissage profond (Huang et al., 2013; Severyn and Moschitti, 2015) et des connaissances exprimées dans les ressources sémantiques (Xiong and Callan, 2015b; Pal et al., 2014). Notre intuition est de combiner la sémantique relationnelle issue des ressources externes et la sémantique distributionnelle apprise via l'apprentissage de représentation. Contrairement aux travaux antérieurs des modèles neuronaux profonds pour la RI (Huang et al., 2013; Hu et al., 2014; Pang

et al., 2016) qui s'appuient uniquement sur la sémantique distributionnelle des textes et aux travaux qui apprennent la représentation sémantique des objets (entités/concepts) en s'appuyant uniquement sur les ressources sémantiques (Bordes et al., 2013; Faruqui et al., 2015; Xu et al., 2014), notre enjeu principal est d'estimer une fonction de pertinence s'appuyant sur une représentation sémantique des documents qui prend simultanément en compte les objets et leurs relations exprimés dans une ressource externe. Dans cette optique, nous modélisons d'abord dans un vecteur à faible dimension la sémantique relationnelle du texte en considérant conjointement les objets et les relations. Ensuite, nous examinons l'hypothèse que la combinaison de la sémantique relationnelle et de la sémantique distributionnelle permettrait d'améliorer la représentation de texte obtenue dans l'espace latent et à terme, améliorer l'appariement requête-document.

À notre connaissance, il s'agit d'une des premières approches combinant la sémantique distributionnelle et relationnelle dans une architecture neuronale pour améliorer l'appariement de requête-document, comme attesté dans Zhang et al. (2016). Plus précisément, nous présentons dans ce chapitre le modèle DSRIM (*Deep Semantic Resource Inference Model*) (Nguyen et al., 2017a,b) qui repose sur les contributions suivantes :

• Une représentation des requêtes et des documents, au niveau de l'entrée du réseau de neurones, s'appuyant sur les connaissances d'une ressource sémantique. Plus particulièrement, cette représentation repose sur l'hypothèse qu'un texte est un sac d'objets provenant d'une ressource sémantique et que des textes sémantiquement similaires contiennent des objets similaires ou connexes. Pour traiter un grand nombre de relations objet-à-objet, nous proposons la méthode *relation mapping* qui vise à projeter des paires dans un espace à faible dimension de groupes d'objets. Notre méthode est flexible puisqu'elle peut être utilisée avec n'importe quelle ressource fournissant des objets et des relations entre objets.
• Un réseau de neurones siamois "de bout en bout" qui apprend une fonction d'appariement des documents en utilisant des vecteurs d'entrée combinant à la fois les représentations distributionnelles et les représentations de document/requête basées sur les ressources de la connaissance.

Une évaluation expérimentale est menée afin de valider notre approche. Pour cela, nous utilisons deux jeux de données TREC, à savoir TREC PubMed CDS et TREC GOV2 Terabyte et deux ressources sémantiques, respectivement MeSH [1] et WordNet [2]. Nous soulignons également que, à la différence des travaux précédents qui utilisent les textes courts (e.g. titre du document) (Huang et al., 2013; Severyn and Moschitti, 2015), nos expérimentations utilisent les textes intégraux des documents.

1. https ://www.nlm.nih.gov/mesh/
2. http ://wordnet.princeton.edu

2 Modèle neuronal d'appariement augmenté par une ressource sémantique

Nous décrivons dans cette partie notre modèle neuronal conçu pour une tâche de RI *ad-hoc* et exploitant la sémantique relationnelle fournie par une ressource sémantique externe. Nous pensons en effet que cette sémantique relationnelle combinée à la sémantique distributionnelle inférée à travers le corpus permettrait d'améliorer la qualité de l'appariement requête-document. Notre contribution repose sur les deux questions de recherche suivantes, illustrées à la Figure 5.1 :

• **RQ1** : Comment modéliser la sémantique relationnelle des textes en s'appuyant conjointement sur les objets et leurs relations exprimés par les ressources sémantiques ?

• **RQ2** : Comment apprendre la fonction de pertinence d'une paire de document-requête en combinant la sémantique relationnelle et distributionnelle du texte dans une architecture neurale siamoise ?

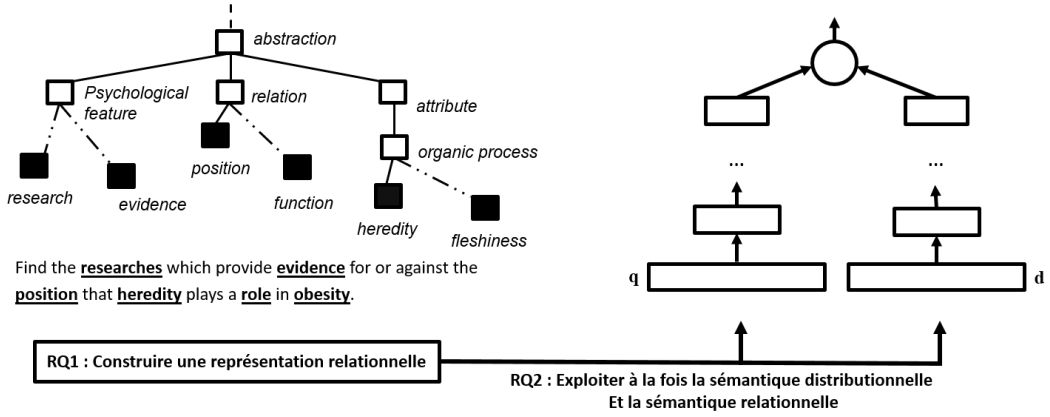


Figure 5.1 – Les enjeux principaux de notre contribution et les questions de recherche.

Pour répondre aux questions de recherche posées ci-dessous, nous modélisons d'abord une représentation vectorielle qui encode la sémantique relationnelle du texte à travers des objets et des relations issus d'une ressource externe. Puis un réseau de neurones profond de type siamois est utilisé pour apprendre une fonction d'appariement du texte en utilisant des vecteurs de représentations. Nous détaillons dans ce qui suit ces deux phases du modèle.

2.1 *Représentation vectorielle de la sémantique relationnelle*

Cette section consiste en notre contribution pour répondre à la question de recherche **RQ1** ayant pour objectif de modéliser la sémantique relationnelle des textes en s'appuyant conjointement sur les objets et leurs relations exprimés par les ressources sémantiques. Notre but est de construire un vecteur d'entrée incluant la sémantique relationnelle afin de l'injecter dans un réseau de neurones profond dédié à une tâche de RI (Section 2.2). Nous rappelons dans un premier temps le formalisme d'une ressource de connaissance et présentons ensuite les hypothèses et la modélisation de ce vecteur de représentation.

2.1.1 *Notations*

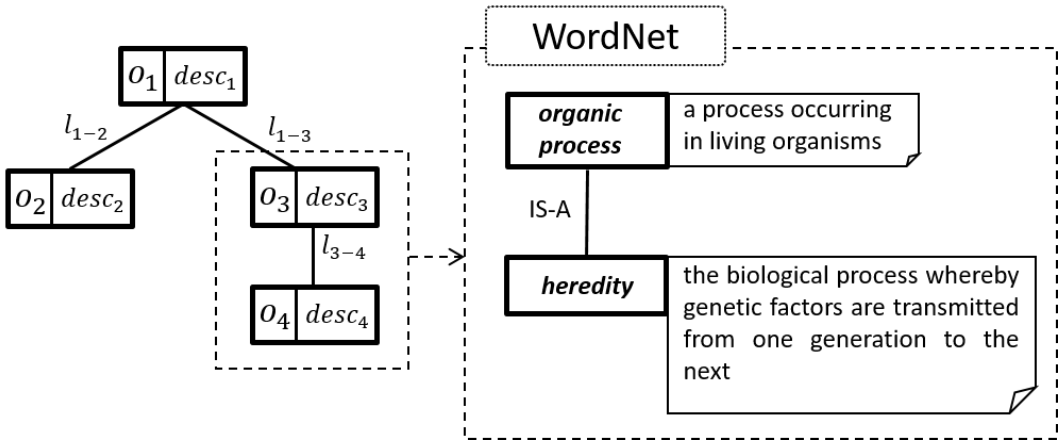


Figure 5.2 – Illustration d'une structure de ressource avec l'exemple de WordNet.

Dans ce modèle, une ressource sémantique $\mathcal{R}$ est modélisée par un graphe $\mathcal{G} = (\mathcal{V}, L)$ où $\mathcal{V}$ est un ensemble de nœuds et L est un ensemble de liens. Chaque nœud $v_i =< o_i, desc_i >$ représente un objet o_i (e. g., mot, entité, concept) et son étiquette textuelle $desc_i$ (e. g., définition du mot). Chaque lien $l_{i,i'}$ exprime une relation sémantique entre les objets o_i et $o_{i'}$.
La Figure 5.2 illustre une structure de ressource sémantique avec quatre nœuds et leur relation. La figure montre aussi un exemple avec deux objets (synsets) dans WordNet, l'objet *organic process* prend le texte "*a process occurring in living organisms*" comme son l'étiquette textuelle. La relation *IS-A* qui relie l'objet *heredity* et *organic process* peut se traduire par "*heredity* est un *organic process*".

A partir de l'ensemble O des objets dans le graphe $\mathcal{G}$ de la ressource sémantique $\mathcal{R}$, il est possible d'identifier, pour chaque texte T, un ensemble des objets $O(T) \subset O$ trouvé dans le texte T. Cette identification peut se faire manuellement

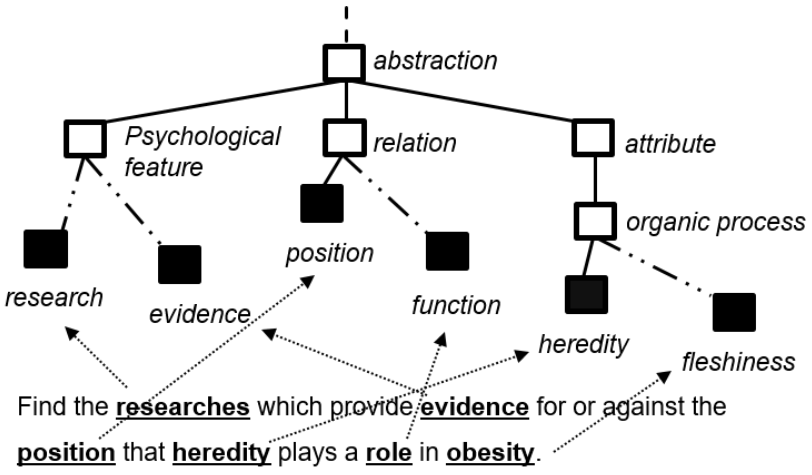


Figure 5.3 – Exemple de l'identification des objets dans le texte

ou automatiquement par un outil d'extraction des concepts/entités. La Figure 5.3 donne un exemple de l'identification des objets de WordNet (synsets) dans une phrase. Dans cet exemple, chaque terme en gras dans le texte est associé à un objet dans la ressource (e. g., le terme **obesity** est identifié avec le synset *fleshiness* dans WordNet).

2.1.2 *Hypothèses de modélisation*

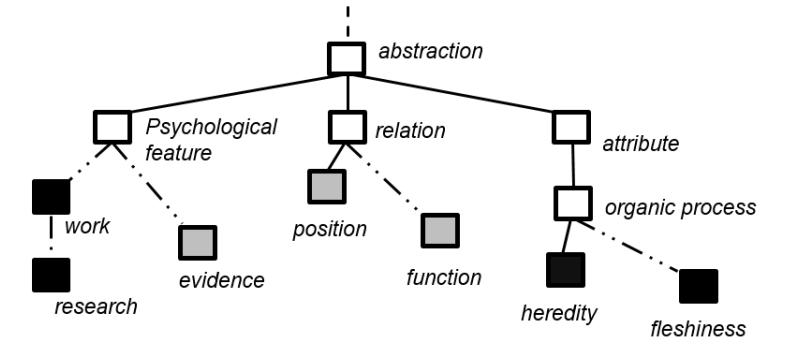


Figure 5.4 – Illustration de deux hypothèses. H1 : les concepts de WordNet sont identifiés dans le texte (mots soulignés). H2 : deux textes sémantiquement similaires contiennent des concepts similaires ou connexes (mot en gras).

Notre représentation sémantique d'un texte repose sur deux hypothèses :

— **(H1)** un texte est un sac d'objets provenant d'une ressource sémantique.
— **(H2)** des textes sémantiquement similaires contiennent des objets similaires ou connexes, en termes de relation définie dans la ressource.

La Figure 5.4 illustre nos deux hypothèses, chaque texte est annoté avec des objets de la ressource (mots soulignés), et les objets qui appartiennent à deux textes sont mis en évidence (mots en gras et objets noircis).

Afin de formaliser l'hypothèse H1 qui considère qu'un texte T est un sac d'objets provenant d'une ressource sémantique, il est possible d'utiliser un vecteur de représentation binaire où chaque élément fait référence à un objet $o_i \in V$ de la ressource et encore sa présence/absence dans le texte T. Egalement, une autre solution serait de construire un vecteur combinant les représentations distributionnelles x_i^d des objets $O(T)$ identifiés pour un texte T. La Figure 5.5 illustre des vecteurs binaires de deux textes T_1 et T_2. Les objets identifiés dans le texte sont mis en évidents (en noir) dans la ressource. Chaque vecteur représente la présence/absence des objets dans le texte via l'activation (valeur) de chaque dimension. Dans cet exemple, le vecteur de représentation a la taille de 3239, qui correspond au nombre d'objets dans la ressource. Chaque cercle blanc/noir signifie une valeur binaire 0/1 pour chaque dimension.

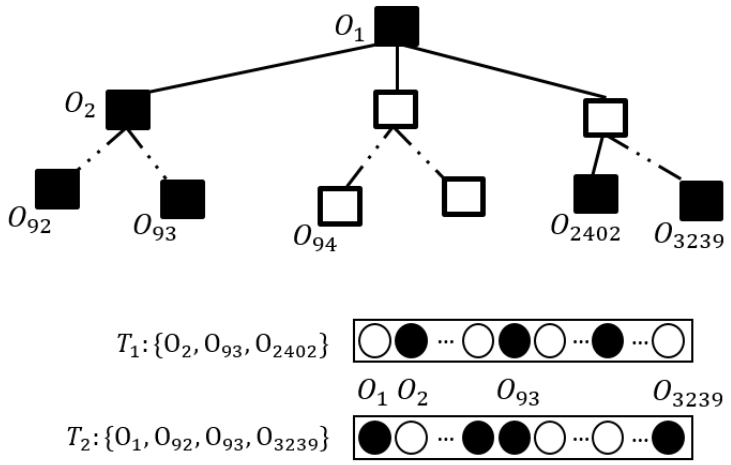


Figure 5.5 – Illustration de la représentation binaire des objets dans le texte.

Néanmoins, cette modélisation ne permet pas de satisfaire l'hypothèse H2. En effet, un vecteur binaire des objets o_i représente seulement l'occurrence des objets dans le texte, il n'est pas capable de donner les informations sur les relations entre différents objets. Il en est de même pour la représentation basée sur la concaténation des représentations distribuées des objets, ces dernières peuvent capturer

la sémantique distributionnelle à travers le texte mais ils ne contiennent aucune information sur la sémantique relationnelle définie par la ressource sémantique.

Pour faire face à ce problème, une solution naïve est d'utiliser un vecteur binaire représentant des paires "objet-objet" qui peut capturer simultanément : 1) les objets appartenant à un texte et 2) leur liaison ainsi que leur similarité. La Figure 5.6 illustre la représentation binaire des relations "objet-objet". Dans cet exemple, chaque dimension de l'espace vectoriel correspond à un lien direct entre deux objets dans la ressource, chaque cercle noir signifie la valeur 1 pour chaque lien si ce lien contient un objet identifié dans le texte.

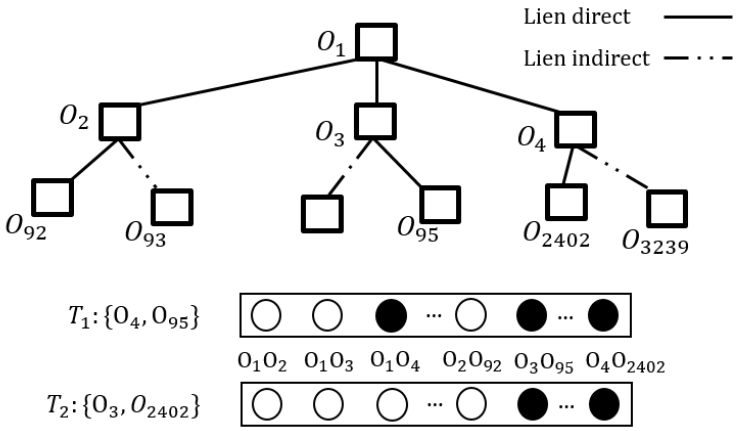


Figure 5.6 – Illustration de la représentation binaire des relations directes "objet-objet".

Cependant, le grand nombre de paires potentielles d'objets, plus précisément le nombre important de relations entre les objets d'une ressource sémantique conduirait à un vecteur de grande dimension et de faible densité. Pour pallier ce problème, nous proposons une méthode de *projection de relations* qui vise à ces deux sous-objectifs :

— Sous-objectif 1 : réduire la dimensionnalité de la représentation vectorielle pour la rendre plus dense
— Sous-objectif 2 : construire des représentations des objets appartenant au texte T ainsi que leurs relations selon l'hypothèse H2

Nous décrivons dans ce qui suit notre approche pour atteindre ces deux sous-objectifs, qui se décompose en deux étapes : 1) la construction de l'espace des représentations pour les objets de la ressource sémantique et 2) la représentation symbolique des documents dans cet espace.

2.1.3 *Espace de représentation des objets*

Tout d'abord, notre objectif est de modéliser un espace vectoriel représentant les objets de la ressource sémantique. Une approche naïve consiste à considérer les objets de la ressource comme des vecteurs binaires dans l'espace de dimension $|\mathcal{V}|$. Même si le nombre d'objets $|\mathcal{V}|$ dans la ressource est significativement inférieur au nombre de relations objet-à-objet $|L|$, il reste un problème pour l'efficacité de calcul. En effet, le nombre d'entités dans une ontologie est souvent grand, par exemple DBpedia contient les informations de plus de quatre millions entités. Ce nombre reste problématique pour un vecteur d'entrée du réseau de neurones profond. Il nécessite à la fois une ressource de calcul performante et un temps d'exécution important. De plus, la quantité d'objets dans un document est souvent négligeable par rapport au nombre total d'objets dans une ressource. Cela engendre des vecteurs de document très parsimonieux, qui peuvent affecter la qualité de l'apprentissage.

Pour répondre au sous-objectif 1 (réduire la dimensionnalité de l'espace vectoriel de la représentation), nous considérons plutôt les groupes d'objets comme représentants de chaque dimension de l'espace vectoriel. Nous proposons de construire k groupes thématiques g_j des objets $o_i \in O$, qui sont supposés être mutuellement indépendants. Ces k groupes d'objets constituent un référentiel $\mathcal{R}' = \{g_1, \ldots, g_k\}$ de la ressource $\mathcal{R}$. En pratique, nous utilisons l'algorithme de partitionnement *k-means* afin de regrouper les objets sur la base de leur représentation thématique. Le nombre de groupes thématiques k est ajusté expérimentalement (Section 3.3.1).

Plus spécifiquement, pour chaque objet o_i, nous considérons son étiquette textuelle $desc_i$ comme le contenu thématique. Pour préparer les représentations thématiques des objets, nous utilisons le modèle *ParagraphVector* entraîné sur l'étiquette textuelle $desc_i$ de chaque objet o_i. De ce fait, chaque objet o_i est associé à une représentation distribuée x_i^{desc} de son contenu thématique. Ainsi, nous utilisons cette représentation distribuée x_i^{desc} comme la représentation thématique pour appliquer le partitionnement avec *k-means*. Nous obtiendrons un espace de représentation des objets de dimension k, dans lequel k est le nombre de groupes thématiques.

2.1.4 *Représentation symbolique de texte guidée par les ressources sémantiques*

Etant donné l'espace de représentation des objets défini ci-dessus, l'objectif suivant est de modéliser une représentation vectorielle de texte (document/requête) qui capture la sémantique relationnelle issue de la ressource sémantique. Autre-

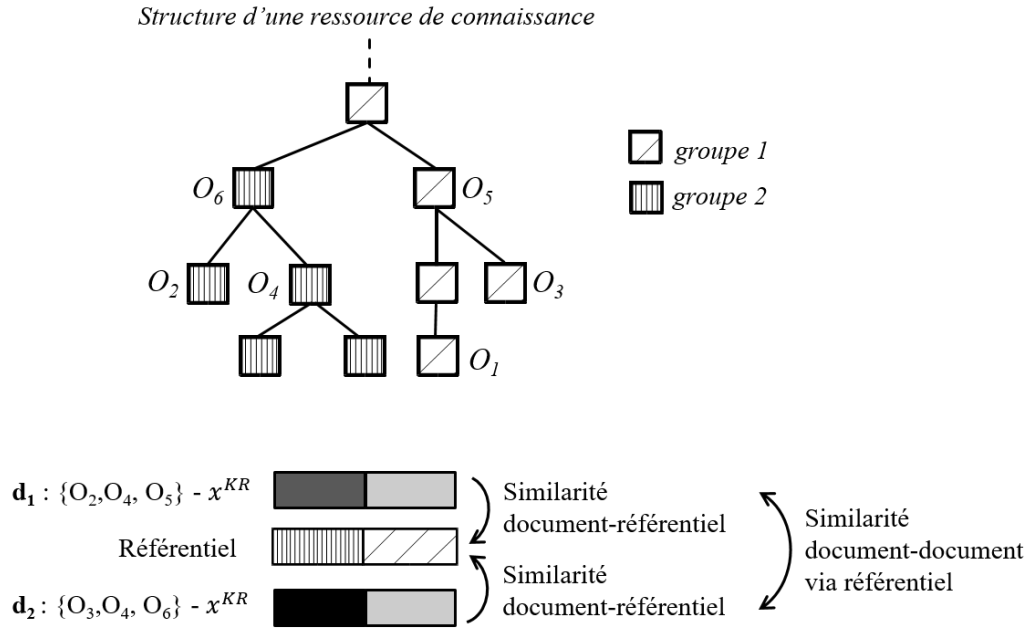


Figure 5.7 – Intuition de la propriété transitive dans la représentation des documents guidée par les ressources de la connaissance

ment dit, nous proposons de pondérer les vecteurs de documents sur cet espace de représentation en tenant compte de la relation sémantique entre ces documents.

Notre intuition est que deux documents sont susceptibles d'être similaires s'ils mentionnent des objets qui sont rassemblés autour des mêmes groupes thématiques. Pour ce faire, le degré de similitude entre ces documents est estimé à travers les groupes thématiques. C'est ce que nous appelons une *propriété transitive*, illustrée dans la figure 5.7.
Dans cet exemple, chaque document d_1 et d_2 est modélisé par un vecteur bidimensionnel dans lequel chaque élément représente un groupe thématique. Les niveaux de gris dans la représentation du document expriment le degré de similarité des objets du document par rapport aux groupes thématiques, une valeur plus forte a une couleur plus foncée. Bien que les documents d_1 et d_2 ne soient pas caractérisés par les mêmes objets, ils sont aussi proches du référentiel et ont des représentations similaires.

Intuitivement, pour qu'un document soit "*similaire*" à un groupe thématique g_j, il faut qu'il soit proche en termes de "similarité thématique" et de "proximité relationnelle" dans la ressource. Notons x^{KR} le vecteur symbolique de dimension k du texte T, qui représente la sémantique relationnelle à modéliser $x^{KR} = (x_1^{KR}, ..., x_k^{KR})$. Nous calculons chaque élément x_j^{KR}, qui correspond à la similarité du texte T par

rapport au groupe thématique g_j, comme une combinaison de deux éléments suivants :

— L'importance w_j^T du groupe g_j étant donné le texte T, qui exprime dans quelle mesure l'ensemble des objets $O(T)$ appartenant au texte T sont similaires thématiquement aux objets appartenant au groupe thématique g_j.
— La proximité relationnelle $S_{relat}(g_j, O(T))$ des objets $O(T)$ du texte T par rapport au groupe g_j, qui permet de savoir dans quelle mesure les objets $o_i \in O(T)$ du texte T sont liés à ceux du groupe thématique g_j, en terme de relations définies dans la ressource.

Ainsi, chaque élément x_j^{KR} du vecteur symbolique est calculé par la combinaison comme suit.

$$x_j^{KR} = w_j^T \times S_{relat}(g_j, O(T)) \tag{5.1}$$

Nous détaillons ensuite notre approche pour calculer, pour un texte T donné et un groupe d'objet g_j du référentiel, l'importance thématique w_j^T et le score de relation $S_{relat}(g_j, O(T))$.

Score d'importance du groupe thématique

Le score d'importance w_j^T du groupe thématique g_j exprime dans quelle mesure l'ensemble des objets $O(T)$ appartenant au texte T sont similaires thématiquement aux objets appartenant au groupe thématique g_j. Intuitivement, plus les objets des textes T et T' sont thématiquement similaires par rapport aux groupes thématiques, plus les textes T et T' sont similaires. En supposant que les objets appartenant à un texte représentent un groupe thématique, nous nous appuyons sur des travaux antérieurs traitant de la similarité de groupe (King, 1967) suggérant d'estimer la similarité entre deux ensembles d'objets en agrégeant les similarités entre les objets de ces deux ensembles. Le score d'importance du groupe thématique g_j par rapport à l'ensemble d'objets $O(T)$ est estimé par une agrégation $Agg_{(o_m,o_n)\in O(T)\times g_j}$ des scores sim_t entre chaque objet $o_m \in O(T)$ dans le texte T et chaque objet $o_n \in g_j$ dans le groupe thématique g_j. Plus formellement, ce score w_j^T est estimé par la fonction suivante :

$$w_j^T = Agg_{(o_m,o_n)\in O(T)\times g_j} sim_t(o_m, o_n) \tag{5.2}$$

où $Agg\_$ exprime une fonction d'agrégation (nous considérons ici le maximum pour capturer la meilleure similarité topique entre les objets) ; sim_t estime la similarité thématique entre les représentations vectorielles des objets (ici, nous utilisons la similarité cosinus entre les représentations *ParagraphVector* x_i^{desc} des descriptions des objets).

Proximité relationnelle entre le texte et le groupe thématique

La proximité relationnelle $S_{relat}(g_j, O(T))$ permet de savoir dans quelle mesure les objets $o_i \in O(T)$ appartenant au texte T sont liés à ceux du groupe thématique g_j. Autrement dit, si les objets $O(T)$ du texte T sont liés à un groupe thématique g_j, la valeur de la j^e dimension dans le vecteur symbolique x^{KR} devrait être élevée. Pour ce faire, ce score de relation $S_{relat}(g_j, O(T))$ est estimé en considérant la relation des objets $O(T)$ par rapport à un objet représentatif $\chi(g_j)$ du groupe thématique g_j (e. g., l'objet le plus fréquent dans la collection parmi les objets appartenant au groupe thématique g_j). L'impact de la méthode utilisée pour identifier le représentant $\chi(g_j)$ est étudié expérimentalement (cf. Section 3.2). Pour mesurer la relation entre les objets, nous nous basons sur les approches classiques de l'état-de-l'art qui reposent sur le calcul des chemins entre les objets (Pedersen et al., 2007). Plus formellement, étant donné un objet représentatif $\chi(g_j)$ du groupe g_j, la relation $S_{relat}(g_j, O(T))$ estime la longueur du chemin entre l'objet $\chi(g_j)$ et l'ensemble $O(T)$ par la fonction suivante :

$$S_{relat}(g_j, O(T)) = \sum_{o_m \in O(T)} \log\left(1 + sim_r(\chi(g_j), o_m)\right) \cdot \frac{avg\_no}{|O(T)|} \tag{5.3}$$

$$\text{avec } sim_r(o_n, o_m) = -\log \frac{dist(o_n, o_m)}{2D}$$

où o_m est un objet dans l'ensemble d'objets $O(T)$; $sim_r(o_n, o_m)$ est une mesure de relation entre les objets o_n et o_m, ici nous utilisons la mesure de Leacock (Leacock and Chodorow, 1998)) qui se calcule sur la distance plus courte *dist* entre deux objets (nombre de liens minimal entre deux nœuds), normalisée par deux fois la profondeur maximale D de la hiérarchie ; *avg_no* est le nombre moyen d'objets par document dans la collection. Etant donné que le facteur de normalisation $\frac{avg\_no}{|O(T)|}$ évite les biais dus aux différences de longueur du texte en termes de nombre d'objets.

2.2 *Architecture du réseau de neurones*

Cette section s'intéresse à la question de recherche **RQ2** : comment apprendre la fonction de pertinence d'une paire de document-requête en combinant la sémantique relationnelle et distributionnelle du texte. Pour cela, l'idée est d'injecter la représentation symbolique comme vecteur d'entrée d'un réseau de neurones. Inspiré par les approches neuronales basées sur la représentation (Section 3.1 du Chapitre 3), nous appliquons un réseau de neurones siamois pour exploiter les vecteurs de représentations fournis en entrée. Ce réseau, dit "de bout en bout", apprend une fonction d'ordonnancement des documents en utilisant en entrée les

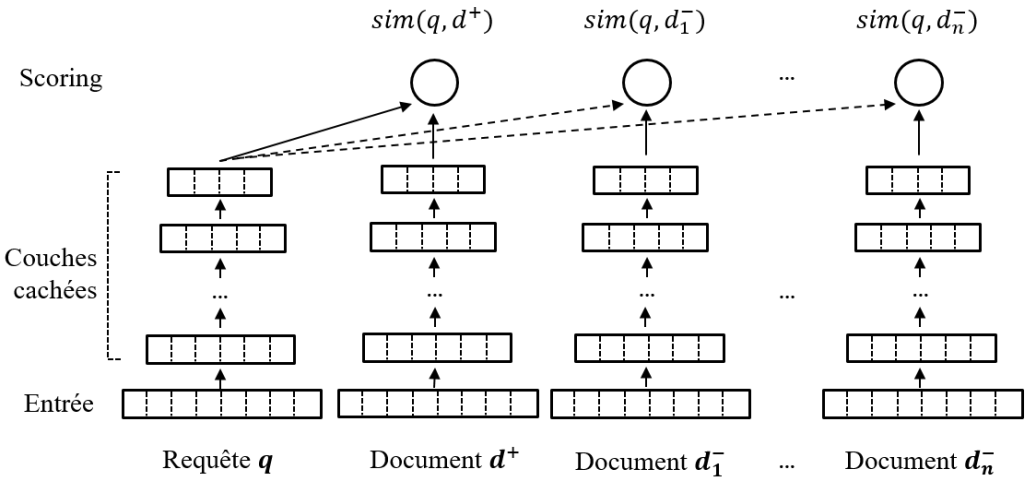


Figure 5.8 – Architecture du réseau DSRIM.

représentations distributionnelles et/ou les représentations symboliques de document/requête basées sur les ressources de la connaissance.

La Figure 5.8 illustre l'architecture de notre réseau de neurones. Ce dernier consiste en un réseau de plusieurs branches, chaque branche correspond à une représentation de texte (requête/document) en entrée. Pour une requête q et un document pertinent d^+, le réseau apprend à discriminer la similarité entre cette paire pertinente (q, d^+) avec les autres paires non-pertinentes (q, d^-). Ainsi, étant donné une représentation d'un texte (requête/document) en entrée, le réseau applique une série de transformations non-linéaires par des paramètres dans une branche, pour apprendre une représentation latente (à la dernière couche avant le scoring). Cette dernière sert à mesurer le score de similarité entre les textes pour apprendre une fonction de classification des documents.

2.2.1 *Vecteur d'entrée*

Afin de répondre la question de recherche **RQ2**, nous proposons de caractériser chaque texte T (un document ou une requête) par un vecteur d'entrée $x_{input} = (x^t, x^{KR})$ modélisé comme vecteur composé de deux parties :

• *Représentation du texte brut* x^t. Cette partie représente les mots du texte intégral T. En nous basant sur les résultats précédents soulignant l'efficacité des représentations sémantiques distribuées pour aborder la question du large vocabulaire dans les RI, nous utilisons le modèle *ParagraphVector* (Le and Mikolov, 2014) entraîné sur les documents de la collection.

• *Représentation symbolique augmentée par la ressource* x^{KR}. Cette partie exprime les objets appartenant au texte T et leurs relations sémantiques exprimées dans la ressource sémantique. Cette représentation est construite par la méthode présentée dans la Section 2.1.

2.2.2 *Apprentissage de la représentation latente*

Pour chaque branche du réseau, le vecteur d'entrée x_{input} du texte T est projeté dans un espace latent à l'aide de L couches cachées l^i ($i = 1, \cdots, L$) afin d'obtenir un vecteur sémantique latent y. Chaque couche cachée l^i et le vecteur sémantique latent y sont respectivement obtenus par les transformations non linéaires suivantes :

$$\begin{aligned} l^0 &= x_{input} \\ l^i &= f(W^{i-1} \cdot l^{i-1} + b^{i-1}) \qquad i = 1, ..., L \\ y &= f(W^L \cdot l^L + b^L) \end{aligned} \tag{5.4}$$

où W^i et b^i sont respectivement la matrice de poids et le biais de la $i^{ème}$ couche. La fonction d'activation $f(x)$ effectue une transformation non linéaire, à savoir ReLU (Unité de Rectification Linéaire) : $f(x) = max(0, x)$ qui est souvent utilisée dans les travaux de l'état-de-l'art (Dehghani et al., 2017).

Après avoir obtenu les vecteurs sémantiques latents y_d et y_q du document d et de la requête q par la transformation non linéaire des couches cachées, le score de similarité cosinus entre les vecteurs document et requête $sim(d|q)$ est calculé.

2.2.3 *Fonction de coût*

Comme la tâche de RI *ad-hoc* concerne un problème d'ordonnancement, nous optimisons les paramètres du réseau de neurones en utilisant une fonction de coût d'ordonnancement relatif, basée sur la distance de similarité Δ entre une paire de document-requête pertinente, notée (q, d^+), et des paires de document-requête non pertinentes, notées (q, d_p^-). Pour ce faire, nous construisons un échantillon de paires de document-requête dans lequel nous opposons, pour la même requête q, un document pertinent d^+ avec n documents non pertinents $d_p^-, p \in [1..n]$, comme suggéré dans (Huang et al., 2013). La différence Δ entre la similarité de la paire pertinente (q, d^+) et des paires non pertinentes (q, d_p^-) est définie comme suit :

$$\Delta = \sum_{p=1}^{n} \left[sim(q, d^+) - sim(q, d_p^-) \right] \tag{5.5}$$

où $sim(\bullet, \bullet)$ est la sortie du réseau de neurones. Pour chaque requête, le réseau est entraîné pour maximiser la distance de similarité Δ utilisant la fonction de coût de

type *hinge-loss L*, bien adapté pour les tâches d'apprentissage d'ordonnancement (Chen et al., 2009) :

$$L = \max(0, \alpha - \Delta) = \max(0, \alpha - \sum_{p=1}^{n} (sim(q, d^+) + sim(q, d_p^-)) \tag{5.6}$$

où α est la marge de L, selon l'amplitude de Δ.

3 Expérimentation et résultats

Dans cette section, nous présentons la méthodologie d'évaluation de notre contribution ainsi que des résultats obtenus. Notre objectif d'évaluation est double : analyse de la représentation symbolique de documents et mesure de l'efficacité de notre modèle en RI.

3.1 *Jeux de données*

Les jeux de données utilisés dans notre protocole consistent en une collection de documents et un ensemble de requêtes avec l'information des documents pertinents de chaque requête. Afin de tester la robustesse de notre modèle aux différentes applications sémantiques, nous évaluons notre approche sur une collection du domaine générique (les documents web) et une collection du domaine spécifique (articles médicaux). Ainsi, notre modèle est évalué sur deux jeux de données GOV2 et PMC dont les statistiques sont présentées dans le Tableau 5.1.

	GOV2	PMC
# Documents	25 000 000	733 138
Longueur moyenne des documents (#mots)	1 132,8	477,1
# Requêtes	150	60
# Paires pertinentes	25 100	8 346

Tableau 5.1 – Statistiques des collections GOV2 et PMC

• Le jeu de données GOV2 [3] est une collection des sites .gov utilisée dans la campagne TREC Terabyte. Nous utilisons les requêtes (*topics*) des campagnes 2004, 2005 et 2006 dont la partie narrative est utilisée comme requête pour entraîner le modèle. La Figure 5.9 illustre une requête dans cette collection.

3. http ://ir.dcs.gla.ac.uk/test_collections/gov2-summary.htm

```
<top>
    <num>711</num>
    <title>Train station security measures
</title>
    <desc>What security measures have been employed at
            train stations due to
            heightened security concerns?</desc>
    <narr>Use of national guard forces is considered relevant.
            Surveillance cameras, more police officers, K-9 units,
            and better ID checks are considered relevant.</narr>
</top>
```

Figure 5.9 – Exemple d'une requête de la collection GOV2.

• Le jeu de données PMC OpenAccess [4] qui regroupe le texte intégral biomédical de PubMed utilisé dans la campagne TREC-CDS. La partie "résumé" (*summary*) des sujets des campagnes d'évaluation 2014 et 2015 est utilisée comme des requêtes pour l'entraînement du modèle. Un exemple d'une requête dans cette collection est illustré dans la Figure 5.10.

```
<topic number="1" type="diagnosis">

    <description>A 58-year-old African-American woman presents to the ER
        with episodic pressing/burning anterior chest pain that began two
        days earlier for the first time in her life. The pain started while
        she was walking, radiates to the back, and is accompanied by nausea,
        diaphoresis and mild dyspnea, but is not increased on inspiration.
        The latest episode of pain ended half an hour prior to her arrival.
        She is known to have hypertension and obesity. She denies smoking,
        diabetes, hypercholesterolemia, or a family history of heart
        disease. She currently takes no medications. Physical examination
        is normal. The EKG shows nonspecific changes.</description>

    <summary>58-year-old woman with hypertension and obesity presents with
    exercise-related episodic chest pain radiating to the back.</summary>

</topic>
```

Figure 5.10 – Exemple d'une requête de la collection PMC.

Afin d'apprendre la sémantique du texte, nous utilisons des ressources sémantiques externes adaptées au domaine d'application de chaque jeu de données. Pour le corpus GOV2, nous considérons WordNet qui est une base de données lexicale anglaise incluant environ 117 000 *synsets* (groupes de mots associés au même concept). Ces *synsets* sont connectés par 6 relations sémantiques, par exemple, la plus commune est *"IS-A"* (hyponymie ou hyperonymie) que nous exploitons dans nos expérimentations (cf. Chapitre 2 - Section 1). Pour le corpus PMC, nous utilisons le thésaurus MeSH, version de 2015, construit par la Bibliothèque américaine de médecine (NLM). Cette ressource comprend 27 000 concepts, organisés en 16 catégories et structurés hiérarchiquement du plus général au plus spécifique (cf. Chapitre 2 - Section 1). Nous soulignons que dans ce travail, nous considérons

4. https ://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

seulement les associations mot-concept pour des mots simples (uni-grammes) et laissons les associations entre concepts et mots composés pour de futurs travaux.

3.2 *Détails d'implémentation et protocole d'évaluation*

Pour construire le vecteur d'entrée sémantique relationnel x^{KR}, les concepts sont extraits à l'aide d'outils d'extraction, à savoir *SenseRelate* (Pedersen and Kolhatkar, 2009) pour le jeu de données GOV2 et Cxtractor [5] basé sur *MaxMatcher* (Zhou et al., 2006) pour le jeu de données PMC.

Dans la modélisation des groupes thématiques pour la représentation x^{KR} (cf. Section 2.1.3), nous avons deux paramètres à configurer : 1) le nombre de groupes thématiques : nous fixons le nombre k de groupes thématique à $k \in \{100, 200\}$; 2) le choix de l'objet représentatif $\chi(g_j)$ dans chaque groupe thématique : nous utilisons trois stratégies : idf_{min}, soit l'objet le plus fréquent ; idf_{max}, l'objet le moins fréquent et *centroid*, objet le plus près du barycentre dans l'espace latent.

Concernant l'architecture de notre modèle, nous considérons les paramètres utilisés dans (Huang et al., 2013). Plus précisément, le nombre de couches cachées est fixé à 2 avec une taille de vecteur caché égale à 64 conduisant à une couche de sortie de 32 nœuds. Le nombre n de paires de document-requête non pertinentes opposé à un pertinent est 4 (Equation 5.5). Des paires de documents-requêtes pertinentes/non pertinentes sont construites sur la base de la vérité terrain de chaque jeu de données, fournissant des jugements de pertinence graduelle de 0 à 2 (critères de pertinence : 1 et 2). Ensuite, les 4 paires non pertinentes opposées à une paire pertinente sont extraites aléatoirement de l'ensemble de paires non pertinentes.

Pour apprendre les paramètres du modèle, nous appliquons la méthode de validation croisée sur 5 sous-échantillons. Les requêtes de chaque jeu de données sont divisées en 5 échantillons dont 4 pour l'apprentissage et la validation du modèle et 1 pour le test. La performance de l'ordonnancement du modèle est moyennée sur 5 échantillons de test. Le modèle est optimisé à l'aide d'une descente de gradient stochastique par mini-lots (SGD) de 5 échantillons. Nous utilisons une régularisation par la norme l2 et une *drop out* à $0,3$. Notre modèle converge généralement après 20 passages sur l'ensemble de données d'apprentissage.

Nous présentons ensuite les résultats obtenus sur différentes évaluations : l'analyse de la représentation sémantique de documents et les mesures de l'efficacité de notre modèle en RI.

5. https ://sourceforge.net/projects/cxtractor/

3.3 *Analyse de la représentation sémantique*

Dans cette section, nous proposons d'analyser notre représentation basée sur les ressources de la connaissance à travers deux objectifs : 1) identifier le paramétrage optimal de la représentation vectorielle et 2) évaluer la validité des vecteurs x^{KR} du document construit.

3.3.1 *Modèles de référence*

Pour évaluer la qualité de notre représentation basée sur les ressources de la connaissance, nous utilisons deux modèles pour construire des représentations de documents :

• **Top_concepts** : Une version naïve de notre représentation basée sur les connaissances et les ressources qui sélectionne les objets les plus fréquents de la collection de documents comme objets représentatifs ($k \in \{100, 200\}$). Ce modèle permet de mesurer l'apport de notre représentation thématique par rapport à cette approche basée sur l'occurence de concepts.

• **LDA** : Le modèle thématique (*topic model*) LDA représentant des groupes de sujets en texte brut (Blei et al., 2003). Cet modèle permet de comparer à notre approche de partitionnement thématique qui s'appuie sur les concepts et les relations dans DSRIM.

Pour notre modèle, nous avons différentes configurations pour choisir les meilleurs paramètres entre le nombre de groupes thématiques $k \in \{100, 200\}$ et les choix de l'objet représentatif $\chi(g_j)$ de chaque groupe : idf_{min}, idf_{max}, $centroid$.

3.3.2 *Résultats et Discussion*

L'évaluation de la qualité de la représentation vectorielle est basée sur l'intuition que des textes sémantiquement similaires, modélisés comme des sacs d'objets, devraient avoir des représentations vectorielles similaires construites selon notre approche et telles représentations devraient aussi écarter des documents non similaires (Mikolov et al., 2013a; Le and Mikolov, 2014). En effet, étant donné un document choisi au hasard (appelé "document pivot"), une bonne représentation vectorielle devrait 1) assurer que la distance entre le document pivot et chaque autre document de la collection n'est pas uniforme, et 2) maximiser la distance entre ses documents les plus similaires et les moins similaires. Cette intuition est illustrée dans la Figure 5.11.

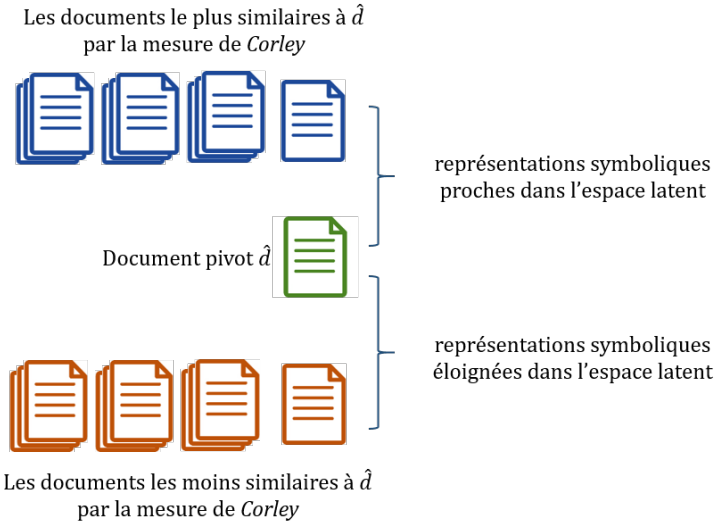


Figure 5.11 – Intuition de l'analyse des représentations symboliques.

Dans ce but, nous identifions d'abord pour chaque document pivot donné, l'ensemble $\mathcal{D}^p+$ de ses 10 documents les plus similaires sémantiquement et l'ensemble $\mathcal{D}^p_-$ de ses 10 documents les moins similaires. Les documents candidats sont sélectionnés sur l'ensemble des données en utilisant une mesure orientée concept proposée dans Corley and Mihalcea (2005), appelée la mesure de similarité agrégée croisée. Cette mesure appliquée aux deux textes T_i et T_j est calculée par la fonction suivante :

$$sim_{corley}(T_i, T_j) = \frac{sim_{corley}(T_i, T_j)_{T_i} + sim_{corley}(T_i, T_j)_{T_j}}{2} \tag{5.7}$$

$$\text{avec } sim_{corley}(T_i, T_j)_{T_i} = \frac{\sum_{o_k \in O(T_i)} (maxSim_{T_j}(o_k) * idf_{o_k})}{\sum_{o_k \in T_i} idf_{o_k}}$$

où $O(T_i)$ est l'ensemble des objets (concepts) appartenant au texte T_i ; $maxSim_{T_j}(o_k)$ est la valeur maximale de la mesure de Leacock (Leacock and Chodorow, 1998) entre l'objet o_k et les objets o_l dans T_j.
Puis nous calculons la similarité cosinus moyenne des représentations des documents pivots avec les ensembles $\mathcal{D}^p_+$ et $\mathcal{D}^p_-$.

Le Tableau 5.2 présente les résultats comparatifs de 100 documents pivots choisis au hasard. La colonne "Top_10" montre la valeur cosinus moyenne entre le document pivot et ses 10 documents les plus similaires $\mathcal{D}^p+$, la colonne "Flop_10" présente la valeur cosinus moyenne entre le document pivot et ses 10 documents les moins similaires $\mathcal{D}^p-$. La colonne "*diff*" calcule la différence entre "Top_10" et "Flop_10".

			GOV2			PMC		
	#groupes k	Rep. $\chi(g_i)$	Top_10	Flop_10	*diff*	Top_10	Flop_10	*diff*
Partitionnement	100 groupes	idf_max	0,7490	0,5776	0,1714	0,5455	0,3035	0,2420
		centroid	0,7411	0,5693	0,1719	0,4807	0,2862	0,1945
		idf_min	0,7018	0,5501	0,1518	0,4975	0,2717	0,2259
	200 groupes	idf_max	0,7595	0,5814	0,1781	0,6359	0,3885	0,2475
		centroid	0,7344	0,5536	0,1808	0,6464	0,3842	0,2621
		idf_min	0,7645	0,5660	0,1985	0,6485	0,4234	0,2251
Référence	Top_concept, $k = 200$		0,9034	0,9013	0,0021	0,9861	0,9616	0,0245
	Top_concept, $k = 100$		0,9123	0,9049	0,0074	0,9817	0,9572	0,0245
	LDA		0,4377	0,3189	0,1188	0,2884	0,0518	0,2639

Tableau 5.2 – Similarité cosinus de la représentation basée sur les ressources sémantiques pour les documents les plus similaires (Top_10) et les moins similaires (Flop_10), moyenne calculée sur 100 documents pivots aléatoires. *diff* : différence entre Top_10 et Flop_10.

Si on compare les résultats obtenus sur les différents jeux de données, on observe que la plage de valeurs cosinus est plus élevée pour GOV2 par rapport à PMC. Cela peut être expliqué par le fait que la représentation de textes utilisant des objets et des relations exprimées dans une ressource sémantique semble plus difficile pour le jeu de données PMC. En effet, ce jeu de données se concentre sur un domaine d'application particulier (à savoir, le domaine médical par opposition au domaine générique pour GOV2) qui pourrait impliquer un vocabulaire plus technique, les accronymes, les abréviations, etc.

En ce qui concerne la méthode utilisée pour définir l'espace de représentation vectorielle (sous-objectif 1 ; Section 2.1), nous constatons que notre approche pour identifier le référentiel en fonction de la classification d'objets est plus efficace que les deux modèles de référence, respectivement Top_concept et LDA. En effet, les différences de similarité des deux ensembles de documents $\mathcal{D}^p_+$ et $\mathcal{D}^p_-$ obtenues par les modèles de référence sont très faibles (la valeur de *diff* est inférieure à 0.11 pour les deux jeux de données, sauf pour LDA sur PMC, contre la différence élevée jusqu'à 0.26 pour notre approche de partitionnement thématique). Il convient de souligner que le choix d'objet représentatif Top_concept n'établit pas de distinction entre les documents les plus ou les moins similaires pour les deux jeux de données. Bien que cette approche donne une similarité élevée entre le document pivot et les documents les plus similaires, elle n'est pas capable de distinguer les documents différents (les valeurs de cosinus des Top_10 et Flop_10 sont toutes

élevées à 0, 90). De plus, les faibles valeurs du cosinus obtenues par le modèle LDA pour les documents les plus similaires (Top_10 inférieur à 0.5) montrent que la représentation de LDA n'est pas capable de capturer la similarité conceptuelle des documents. En revanche, nous soulignons que les valeurs du cosinus pour notre approche de partitionnement semblent être plus intuitives, avec un cosinus moyen pour le jeu de données GOV2 supérieur à 0, 6 pour les documents les plus similaires et inférieur à 0, 6 pour les moins similaires (respectivement 0, 5 pour le jeu de données PMC). Pour rappel, notre approche modélise un espace de représentation des documents basé sur les concepts et relations issues d'une ressource sémantique, tandis que le modèle LDA obtient un vecteur thématique de documents basé sur les statistiques d'occurrence des mots dans le corpus. Ces affirmations suggèrent que notre approche de construction référentielle basée sur le partitionnement thématique semble raisonnable.

En mettant l'accent sur les méthodes utilisées pour choisir le représentant du groupe thématique (*idf_max,idf_min, centroid*), nous pouvons remarquer que les similarités moyennes entre les documents pivot et l'ensemble des documents les plus similaires sont plus importantes pour un nombre plus élevé de groupes (e. g., valeur Top_10 élevée jusqu'à 0, 6485 pour $k = 200$ contre 0, 5455 pour $k = 100$ sur PMC). De plus, ces paramètres permettent d'obtenir des différences plus élevées entre les documents les plus similaires et les moins similaires (avec valeur de *diff* au moins 0, 2251 vs 0, 1945 pour respectivement $k = 200$ et $k = 100$ sur PMC, et 0, 1781 vs 0, 1518 sur GOV2). Nous pouvons ainsi déduire que le meilleur scénario pour $k = 200$ permet de distinguer les documents les plus similaires des documents les moins similaires. Sur PMC, ce scénario consiste à sélectionner l'objet le plus proche du barycentre (*centroid*) du groupe comme objet représentatif alors qu'il n'y a pas des différences significatives entre les trois configurations (*idf_max, idf_min, centroid*) pour GOV2. Étant donné que la méthode *centroid* est plus intuitive avec les hypothèses utilisées pour construire le référentiel, nous conservons le paramétrage avec 200 groupes thématiques et la méthode *centroid* pour encoder l'objet représentatif.

3.4 *Evaluation de l'efficacité du modèle*

Pour évaluer la performance de notre modèle et des différents modèles de référence, nous nous basons sur le protocole de réordonnancement présenté dans Ai et al. (2016a). Ce dernier propose d'utiliser les 2 000 premiers documents sélectionnés par le modèle BM25 pour recalculer le score final en combinant avec le modèle neuronal. Les résultats finaux sont estimés à l'aide des 1 000 premiers documents de chaque modèle de réordonnancement en fonction de la mesure MAP.

3.4.1 *Modèles de référence*

Pour évaluer l'efficacité du modèle, nous utilisons trois types de modèles de référence :

a) Modèles d'appariement exact pour mettre en évidence l'impact de l'utilisation de la sémantique relationnelle et des approches d'apprentissage profonds :
• **BM25** : Le modèle probabiliste classique (*BM25*).
• **LM-DI** : Le modèle de langue basé sur le lissage de Dirichlet, qui est un autre modèle d'appariement exact (Zhai and Lafferty, 2001).

b) Modèles d'appariement sémantique pour décrire l'impact d'un modèle neuronal profond guidé par des ressources sémantiques :
• **LM-QE** : Un modèle de langue appliquant une technique d'expansion de requête basée sur des concepts (Pal et al., 2014) dans lequel les termes candidats sont ordonnés en fonction de leur similarité avec les descriptions des objets dans la ressource sémantique. Les paramètres par défaut mentionnés dans l'article de référence (Pal et al., 2014) sont utilisés.
• **LM-LDA** : Le LM-LDA est un modèle topique latent utilisant le cadre de modélisation du langage (Wei and Croft, 2006).

c) Modèles d'appariement par réseau de neurones profond, également basé sur l'architecture siamoise, pour mettre en évidence l'impact de la combinaison de sémantique relationnelle et distributive dans les approches neuronales :
• **DSSM** : Le modèle d'appariement de l'état de l'art, basé sur un réseau de neurones (Huang et al., 2013). Nous utilisons le code public [6] avec leurs paramètres par défauts. Nous évaluons le modèle DSSM sur les documents en texte intégral.
• **CLSM** : L'extension du modèle DSSM dont l'entrée est remplacé par un réseau de convolution pour mieux capturer des structures contextuelles détaillées (Shen et al., 2014a). Nous utilisons également le code CLSM public [6] sur les documents en texte intégral avec les valeurs des paramètres par défaut.

Pour comparer la performance de notre modèle *DSRIM* sur différents types de vecteur d'entrée, nous utilisons les trois configurations suivantes :
• $DSRIM^{p2v}$: Notre modèle neuronal basé sur une représentation d'entrée des textes limités au texte brut, à savoir x^t (vecteur obtenu par ParagraphVector).
• $DSRIM^{kr}$: Notre modèle neuronal basé sur notre représentation symbolique du texte, à savoir x^{KR}.
• $DSRIM^{kr+p2v}$: Notre modèle neuronal basé sur une représentation améliorée des textes combinant la représentation de texte brut x_t et notre représentation symbolique x^{KR}.

6. https ://www.microsoft.com/en-us/research/project/dssm/

		GOV2			PMC		
Type	Modèle	MAP	%change	p-valeur	MAP	%change	p-valeur
Appariement exact	BM25	0,1777	+4,84	0,6691	0,0348	-1,15	0,9628
	LM-DI	0,1584	+17,61	0,1644	0,0379	-9,23	0,7109
Appariement sémantique	LM-QE	0,0738	+152,44	0,0001 ***	0,0106	+224,53	0,0008 ***
	LM-LDA	0,0966	+92,86	0,0001 ***	0,0185	+85,95	0,0323 *
Modèles neuronaux	DSSM	0,0418	+345,69	0,0001 ***	0,0095	+262,11	0,0008 ***
	CLSM	0,0365	+410,41	0,0001 ***	0,0069	+398,55	0,0001 ***
Notre approche	$\text{DSRIM}^{\text{p2v}}$	0,1115	+67,09	0,0001 ***	0,0183	+87,98	0,0460 *
	DSRIM^{kr}	0,1801	+3,44	0,7461	0,.0307	+12,05	0,6829
	$DSRIM^{kr+p2v}$	*0,1863*			*0,0344*		

Tableau 5.3 – Comparaison de l'efficacité du DSRIM et les modèles de référence sur les collections GOV2 et PMC. %change : amélioration/dégradation significative de $DSRIM^{kr+p2v}$ (+/-). *p-valeur* : Signification de T-test : * : $0.01 < \alpha \leq 0.05$, ** : $0.001 < \alpha \leq 0.01$, *** : $\alpha \leq 0.001$

3.4.2 *Résultats et Discussion*

Nous présentons ici les performances de notre modèle sur les jeux de données GOV2 et PMC. Le Tableau 5.3 présente un résumé des valeurs d'efficacité en termes de MAP pour notre modèle et les différents modèles de référence. La colonne *%change* indique le pourcentage d'amélioration ou de dégradation de notre configuration $DSRIM^{kr+p2v}$ par rapport à chaque modèle de référence. La colonne *p-valeur* montre la valeur-p du test de Student entre notre configuration $DSRIM^{kr+p2v}$ et chaque modèle de référence.

En comparant différentes configurations de notre approche, à savoir $DSRIM^{p2v}$, $DSRIM^{kr}$, et $DSRIM^{kr+p2v}$, nous pouvons voir que le modèle DSRIM appliqué sur notre représentation symbolique x^{KR} fournit une meilleure performance ($p < 0.001$) que celui avec la représentation basée sur le texte simple x^t (e. g., respectivement $0,0307$ et $0,0183$ pour le jeu de données PMC). Ce résultat renforce l'intuition invoquée dans des travaux récents (Guo et al., 2016) sur l'utilisation des interactions locales de termes et/ou de caractéristiques dans le texte. En effet, Guo et al. (2016) ont suggéré que l'extraction des interactions entre le texte de la requête et les documents améliore la performance des réseaux de neurones d'appariement pour la RI. Dans notre modélisation de la représentation symbolique x^{KR}, nous avons pris en compte des interactions entre différents objets entre les textes, ces interactions sont modélisées à travers les relations entre les objets définies dans la ressource. Par ailleurs, en combinant la sémantique distributionnelle et relationnelle à travers le modèle $DSRIM^{kr+p2v}$, on constate que la valeur MAP augmente

légèrement, avec une amélioration significative (e. g., +67.09% et +87.98% pour les données GOV2 et PMC respectivement) par rapport à $DSRIM^{p2v}$. Cela ouvre des perspectives intéressantes dans la combinaison de ces deux types de sémantique.

Dans cette optique, nous commentons les résultats des modèles référence par rapport au modèle $DSRIM^{kr+p2v}$. D'un point de vue général, nous pouvons voir, d'une part, que les modèles d'appariement exact ne sont pas significativement différents de notre modèle proposé, avec une attention particulière pour le jeu de données GOV2 avec de petites améliorations par rapport à BM25 (+4, 84%) et LM-DI (+17, 61%). D'autre part, notre approche surpasse les modèles d'appariement sémantiques et profonds avec des améliorations significatives. Par exemple, notre modèle obtient de meilleurs résultats significatifs pour le jeu de données GOV2 par rapport aux modèles LM-QE, LM-LDA, DSSM et CLSM, pour lesquels notre modèle obtient une valeur de MAP jusqu'à +410, 41% du taux d'amélioration. Ces observations sont similaires pour les deux jeux de données, ce qui souligne le fait que notre modèle est efficace pour tirer parti des ressources sémantiques générales (WordNet) et spécialisées (MeSH). Plus particulièrement, nous pouvons déduire les conclusions suivantes :

• Le modèle BM25 et les modèles de langage sont bien connus en RI comme des modèles de référence solides qui sont difficiles à surpasser avec des modèles d'appariement profonds entraînés sur de petits jeux de données qui ne permettent pas de généraliser la tâche. Les résultats présentés au Tableau 5.3 nous amènent à confirmer cet énoncé. Cependant, il est intéressant de noter que, contrairement à la plupart des approches neuronales antérieures basées sur l'architecture siamoise (Huang et al., 2013; Severyn and Moschitti, 2015; Shen et al., 2014a) qui ordonnent les documents courts (le titre) et utilisent une collection réelle à grande échelle pour former leur modèle, nous expérimentons plutôt notre modèle sur des collections de documents longs en texte intégral (la longueur moyenne des mots vaut 1132, 8 pour GOV2 et 477, 1 pour PMC).

• Le modèle LM-QE réalise une expansion de requête basée sur les ressources sémantiques. Puisque le DSRIM surpasse le modèle LM-QE, nous pouvons suggérer que les représentations sémantiques des documents et des requêtes qui sont apprises à partir de l'entrée construite sur notre représentation symbolique sont plus efficaces que les requêtes étendues avec des descripteurs d'objet pertinents.

• Le modèle LM-LDA est basé sur un modèle probabiliste qui est capable d'identifier les sujets thématiques entre les documents. Notre modèle surpasse généralement ce modèle de référence avec une amélioration significative de 89.95% pour la métrique MAP sur le jeu de données PMC. Ceci est cohérent avec les travaux précédents (Huang et al., 2013) qui soulignent l'efficacité des représentations latentes profondes des textes par rapport à celles obtenues par le modèle LDA.

• Dans la catégorie des modèles de RI neuronaux, notre modèle surpasse les modèles DSSM et CLSM (avec une valeur MAP atteignant 0, 0418 et 0, 0095 pour les deux ensembles de données respectivement). Ces résultats suggèrent que l'intégration de la sémantique relationnelle et distributionnelle au niveau du document (plutôt qu'au niveau du mot) dans la représentation d'entrée permet d'améliorer l'apprentissage du modèle neuronal profond tout en considérant les petites collections (plutôt que les vrais journaux de moteur de recherche) et les textes intégraux (plutôt que les titres).
Il est intéressant de noter que le modèle convolutif CLSM, qui dépasse initialement la DSSM dans Shen et al. (2014a) par des expériences menées sur des données réelles à grande échelle, est moins efficace que la DSSM. Cela pourrait s'expliquer par le fait qu'il est formé en utilisant des collections TREC caractérisées par un nombre limité de requêtes (comme montré également dans Guo et al. (2016)).

Pour mieux comprendre les résultats de notre modèle par rapport au modèle BM25, nous étudions dans quelle mesure l'efficacité de notre modèle dépend du niveau de difficulté des requêtes. Plus particulièrement, nous classons les requêtes selon trois niveaux de difficulté ("facile", "moyen", "difficile") en utilisant l'algorithme *k-means* appliqué aux valeurs MAP de la BM25. Pour cela, nous trions les requêtes en ordre croissant de la valeur MAP obtenue par BM25, puis l'algorithme *k-means* est appliqué sur ces valeurs pour obtenir trois classes de requêtes, selon la valeur MAP croissante, "facile", "moyen", "difficile". Puis on calcule les valeurs moyennes des caractéristiques des requêtes dans les classes. Les statistiques de chaque classe sont présentées dans le Tableau 5.4. Les colonnes MAP, #mots, #objets présentent respectivement les valeurs moyennes de la métrique MAP, nombre de mots et nombre d'objets de chaque classe. La colonne %*change* présente le taux d'amélioration par rapport au BM25 sur la même classe en termes de valeur de MAP obtenu par notre modèle $DSRIM^{kr+p2v}$.

	Difficulté	MAP	#mots	#objets	%change
GOV2	Facile	0,52	22,95	12,11	-16,60%
	Moyen	0,25	20,79	11,79	-5,15%*
	Difficile	0,05	22,15	12,14	+87,15%***
PMC	Facile	0,16	13	5,4	-0,22%
	Moyen	0,04	16,68	5,36	-25,78%
	Difficile	0,004	18,5	6,3	+63,60%*

Tableau 5.4 – Statistiques des requêtes selon leur niveau de difficulté.

Nous pouvons souligner que, pour le jeu de données PMC, les requêtes difficiles incluent significativement plus de termes et plus d'objets que les requêtes faciles et moyennes. Toutefois, il n'y a pas de différences significatives entre les

différentes classes de requêtes en termes de nombre de mots et d'objets sur le jeu de données GOV2. En se focalisant sur l'efficacité en RI, on peut voir que les améliorations de $DSRIM^{kr+p2v}$ par rapport au modèle BM25 sont à la fois positives et significatives pour les requêtes difficiles sur GOV2 et PMC. De plus, il convient de mentionner que les taux d'amélioration pour les requêtes difficiles ($+63.60\%$ pour le jeu de données PMC) sont significativement différents de ceux des requêtes moyennes et faciles (respectivement -25.78% et -0.22% pour le jeu de données PMC, sans différence significative entre requêtes faciles et moyennes, $p > 0.5$). Il est intéressant de noter que, en combinant les taux d'amélioration et le nombre d'objets pour les requêtes moyennes de le jeu de données GOV2, nous pouvons voir que la baisse significative de l'efficacité de notre modèle (-5.15%) pourrait s'expliquer par le plus petit nombre d'objets associés à cet ensemble de requêtes. Ces résultats montrent qu'il est plus efficace d'exploiter la sémantique relationnelle grâce à notre représentation symbolique pour résoudre les requêtes difficiles. Ceci est cohérent puisque ces requêtes sont généralement caractérisées par un grand nombre de mots et d'objets identifiés dans le texte. Par conséquent, nous pouvons affirmer que notre modèle est particulièrement util pour réduire le fossé sémantique entre les représentations basées sur les mots et celles basées sur les concepts, ce qui favorise probablement la discrimination entre les documents pertinents et non pertinents.

Nous avons remarqué dans le Tableau 5.4 que, pour le jeu de données PMC, notre modèle $DSRIM^{kr+p2v}$ permet d'améliorer l'efficacité de RI de par rapport au BM25 sur les requêtes difficiles, alors que ces dernières contiennent plus de termes et plus d'objets que les requêtes faciles et moyennes. Pour avoir une vision plus détaillée, nous illustrons quelques exemples des requêtes de chaque classe de difficulté dans le Tableau 5.5. Nous avons choisi, pour chaque niveau de difficulté, deux requêtes ayant le meilleur taux d'amélioration (colonne %change) obtenus avec notre modèle $DSRIM^{kr+p2v}$ par rapport au BM25 (colonne MAP indique la valeur obtenue par BM25). Les objets sont soulignés dans le texte de la requête.

Dans cet exemple, on peut voir que les requêtes difficiles sont plus longues et ainsi impliquent un plus grand nombre d'objets (concepts) dans le texte. Cela permet notre modèle de mieux capturer la sémantique de la requête via les concepts identifiés. Pourtant, comme nous considérons seulement des concepts mono-terme, notre modèle d'appariement conceptuel peut être sous-optimal. En effet, les concepts médicaux sont souvent composés de plusieurs termes, la prise en compte des concepts mono-terme entraîne une faible précision. Par exemple, dans la première requête de la classe Moyen, le concept "*atrophy*" (CUI C0333641) est annoté dans le texte, mais il existe le concept "*cortical atrophy*" (CUI C0235946) qui est plus précis dans ce cas.

Difficulté	MAP	%change	#objets	Requête
Facile	0,1033	2,3%	3	43-year-old woman with soft, flesh-colored, pedunculated lesions on her neck.
	0,2192	1,1%	5	An obese 28 yo female with non-ruptured ectopic pregnancy and history of adhesions.
Moyen	0,0221	2,4%	8	62-year-old man with progressive memory loss and involuntary leg movements. Brain MRI reveals cortical atrophy, and cortical biopsy shows vacuolar gray matter changes with reactive astrocytosis.
	0,0641	2%	8	A 65-year-old male complains of productive cough with tinges of blood. Chest X-ray reveals a round opaque mass within a cavity in his lung. Culture of the sputum revealed fungal elements.
Difficile	0,001	125%	10	10-year-old boy with progressive right knee and left leg pain and edema, lethargy and an osteolytic lesion. No history of trauma, fever, tachycardia, or urinary incontinence.
	0,002	202%	11	28-year-old female with neck pain and left arm numbness 3 weeks after working with stray animals. Physical exam initially remarkable for slight left arm tremor and spasticity. Three days later she presented with significant arm spasticity, diaphoresis, agitation, difficulty swallowing, and hydrophobia.

Tableau 5.5 – Exemple des requêtes du jeu de données PMC

Concernant le modèle BM25, nous constatons dans les exemples, une des raisons de la dégradation de la performance. Le premier exemple dans la classe Difficile contient la négation ("*No history of ...*") de quelques symptômes. Cet aspect est connu comme une des causes importantes du fossé sémantique que les systèmes de RI doivent résoudre (Koopman et al., 2016). En effet, les modèles RI classiques comme BM25 utilisent l'appariement exact de terms, ils peuvent pas reconnaître la négation des termes. Un document qui contient plusieurs mots "*trauma*", "*fever*", "*tachycardia*" aura un score élevé pour la requête "*No history of trauma, fever, tachycardia*. Pour ce faire, il est nécessaire prendre en compte la négation dans les requêtes, soit par une étape de prétraitement, soit par une modélisation spéciale dans la recherche. Cela suggère une amélioration pour notre modèle dans les futurs travaux.

4 Bilan

Nous avons proposé le modèle DSRIM, un réseau de neurones profond pour la RI. Notre modèle exploite à la fois la sémantique distributionnelle à travers l'algorithme *ParagraphVector*, et la sémantique relationnelle, à travers une représentation de textes basée sur les connaissances dans des ressources externes. Ces dernières visent à modéliser conjointement des objets intégrés dans le texte et des relations structurées entre objets.

L'évaluation expérimentale sur deux jeux de données TREC est effectuée pour évaluer la qualité des représentations en entrée du réseau ainsi que leur impact sur l'efficacité de l'ordonnancement des documents. Les résultats montrent que 1) notre représentation basée sur les connaissances dans les ressources externes permet de distinguer les textes sémantiquement similaires, et que 2) notre modèle surpasse les approches sémantiques ainsi que les modèles de RI neuronaux. Une extension possible de ce modèle est la prise en compte des relations hétérogènes des objets. En outre, il serait intéressant d'explorer la faisabilité d'un modèle de transition (*translation model*) qui exploiterait la ressource sémantique externe comme une troisième branche du réseau afin de traduire la relation sémantique entre la requête et le document.

CONCLUSION

CONCLUSION GÉNÉRALE

Synthèses des contributions

Les travaux présentés dans cette thèse s'inscrivent dans le contexte général de la recherche d'information et plus spécifiquement dans le cadre de la RI sémantique. Nous nous sommes particulièrement intéressés aux approches qui exploitent les ressources sémantiques ainsi que les modèles neuronaux pour améliorer l'efficacité de la RI. Nous avons axé notre état-de-l'art selon ces deux lignées de travaux, la première exploite la connaissances relationnelle dans les ressources sémantiques pour améliorer les modèles de RI, la deuxième utilise les modèles neuronaux pour capturer la sémantique distributionnelle dans le corpus de documents.

Dans ce contexte, nous avons posé cette principale question de recherche principale : Comment combiner la sémantique relationnelle exprimée dans les ressources externes avec la sémantique distributionnelle issue de corpus de textes pour améliorer les performances de la RI ? Plus spécifiquement, nous nous concentrons sur ces deux sous-questions de recherche :

1. Comment améliorer les représentations de textes à partir de ces deux sources d'informations, le corpus de documents et la ressource externe ?
2. Comment intégrer ces deux types de sémantique dans un modèle d'apprentissage d'ordonnancement pour mieux apparier les documents et les requêtes ?

Pour répondre à ces questions, nous avons proposé deux principales constributions. La première contribution consiste en des méthodes d'apprentissage de représentations de textes à plusieurs niveaux, en intégrant les contraintes relationnelles issues des ressources externes. Notre deuxième contribution porte sur un modèle d'appariement par un réseau de neurones avec une représentation symbolique des documents en entrée de l'apprentissage. Nous avons aussi mené plusieurs expérimentations pour mesurer et analyser l'efficacité de nos modèles. Nous décrivons brièvement ces contributions dans ce qui suit.

1. Concernant les modèles d'apprentissage de représentations distribuées de textes, nous avons proposé deux approches qui intègrent la sémantique relationnelle dans l'apprentissage de représentations. L'objectif de la première approche vise à améliorer la représentation de document pré-entraînée par

un modèle neuronal en combinant cette représentation originale basée sur le texte brut avec une nouvelle représentation conceptuelle apprise sur les concepts identifiés dans le même document. Ces deux types de représentations sont entraînés séparément sur deux types de collections, une basée sur les mots, une basée sur les concepts. Puis, pour chaque document, nous combinons ces deux représentations, à savoir la représentation sur le texte brut et la représentation conceptuelle, en utilisant une méthode d'optimisation. Ainsi, nous obtenons pour chaque document, une représentation dite optimale qui prend en compte à la fois la sémantique distributionnelle issue du corpus de documents et la sémantique relationnelle issue de la ressource externe. Nous proposons ensuite une deuxième approche qui apprend simultanément les représentations de documents, de mots et de concepts. Les représentations des documents sont apprises en maximisant la prédiction des vecteurs de mots et de concepts en fonction de leur contexte voisinage. Pour mieux capturer la sémantique relationnelle dans les représentations de texte, nous proposons d'intégrer les contraintes de relations entre les mots/-concepts dans ces deux méthodes d'apprentissage. Deux approches sont utilisées pour injecter ces contraintes relationnelles, une basée sur la régularisation de la fonction objectif, une autre basée sur les instances dans le texte d'entraînement. Ces propositions sont expérimentalement comparées en utilisant des jeux de données génériques ainsi que spécifiques au domaine (médecine). Nous avons analysé la qualité de nos représentations de documents sur les tâches TALN (i.e., tâches de similarité et de classification de phrases) et de RI (i.e., ré-ordonnancement et expansion de requêtes).

Nous avons analysé d'abord nos propositions de base qui intègrent les concepts dans l'apprentissage sans la contrainte des relations. Les résultats montrent en général des améliorations de performances de nos propositions par rapport aux modèles de référence. Plus spécifiquement, sur les tâches de RI, nos modèles obtiennent des améliorations plus importantes par rapport aux modèles de référence sur le domaine médical comparativement au domaine générique. Cela suggère que les domaines spécifiques nécessitent un modèle sémantique permettant de capturer davantage d'inférences qu'un modèle lexical (e.g., BM25). Sur les tâches de TALN, nos modèles obtiennent généralement de meilleurs résultats pour le jeu de données génériques que sur les jeux de données médicales. En effet, les tâches de TALN requièrent un raisonnement à un niveau global, niveau d'inférence plus facile à effectuer sur des jeux de données génériques.

En ajoutant la contrainte de relation dans nos modèles, les résultats obtenus sont généralement améliorés par rapport aux modèles de base en intégrant seulement les concepts. Cette amélioration est plus importante sur les tâches de TALN que sur la RI. Entre les deux méthodes d'intégration de contrainte relationnelle, la méthode basée sur la régularisation donne généralement de

meilleurs résultats, par rapport à la méthode basée sur l'instance, pour nos deux modèles d'apprentissage hors ligne et en ligne.

2. Nous modélisons dans la deuxième contribution un modèle neuronal pour l'appariement des paires de document-requête. À notre connaissance, il s'agit d'une des premières approches combinant la sémantique distributionnelle et relationnelle dans une architecture neuronale pour améliorer l'appariement de requête-document. Dans cette contribution, nous avons proposé une méthode pour construire une représentation symbolique de texte, qui s'appuie sur les concepts et leurs relations dans une ressource externe. Vu le grand nombre de relations objet-objet dans le texte, nous avons proposé une méthode *relation mapping* qui vise à projeter des paires dans un espace de groupes d'objets à faible dimension. Nous avons proposé ensuite un réseau de neurones pour apprendre la fonction d'appariement des documents en utilisant des vecteurs d'entrée combinant à la fois les représentations distributionnelles et les représentations de document/requête basées sur des ressources sémantiques. Une évaluation expérimentale est menée afin de valider notre approche. Pour cela, nous utilisons deux jeux de données TREC, à savoir TREC PubMed CDS et TREC GOV2 Terabyte et deux ressources sémantiques, respectivement MeSH et WordNet. Nous avons évalué la modélisation de la représentation symbolique à l'aide d'une mesure de similarité conceptuelle des documents. Les analyses nous permettent de valider la sémantique capturée par notre représentation symbolique, qui est plus performante en la comparant avec le modèle LDA. Concernant l'efficacité de RI, les résultats montrent que notre représentation symbolique basée sur les connaissances dans ressources externes permet d'améliorer la performance du modèle d'apprentissage, et que notre modèle surpasse les approches sémantiques (e. g., modèle basé sur LDA et modèle d'expansion de la requête), ainsi que les modèles de RI neuronaux.

Perspectives

Nos contributions et expérimentations peuvent bénéficier de plusieurs perspectives pour nos futurs travaux sur le court terme ainsi que sur le moyen terme.

A court terme, nos perspectives portent essentiellement sur les aspects suivants :

— Comme nos modèles exploitent les concepts identifiés dans les documents, il est crucial d'améliorer la qualité d'annotation des documents. Tandis que nous avons utilisé des méthodes d'extraction des concepts bien robustes dans l'état-de-l'art (Aronson, 2001), il reste toujours du bruit dans l'annotation.

Nous avons appliqué une solution permettant de réduire le bruit d'annotation, qui consiste à entraîner les modèles sur les corpus annotés manuellement (e. g., Wikipédia), puis continuer à entraîner sur le corpus principal de la tâche. Cette solution nécessite pourtant une grande collection annotée pour chaque domaine thématique. Par exemple, dans le domaine médical, ce type de collection annotée manuellement reste insuffisant par rapport à Wikipédia. Nous pensons à recueillir plus de collections annotées manuellement pour avoir une base plus robuste pour l'apprentissage de représentations sur les collections spécifique au domaine. Une possibilité est de prendre les pages Wikipédia liées à un domaine spécifique pour pré-entraîner les représentations. Il existe des projets sur Wikipédia (WikiProject) qui permettent d'identifier les pages dont le contenu est spécifique à un domaine (e. g., WikiProject Medicine).

— Concernant l'apprentissage du réseau de neurones DSRIM pour l'appariement des documents, il est nécessaire d'augmenter la qualité de l'apprentissage du modèle. Une possibilité est d'augmenter le nombre d'exemples pour l'apprentissage en utilisant la méthode d'*apprentissage faible supervisée* (Dehghani et al., 2017). Cette dernière consiste à utiliser les top documents renvoyés par un modèle de RI classique comme exemples positifs pour apprendre le score de pertinence étant donnée une requête.
— Il est nécessaire de tester la représentation distribuée apprise par nos modèles (Chapitre 4) dans le modèle de RI neuronal DSRIM pour évaluer la qualité de cette représentation sur l'efficacité de l'appariement neuronal. D'ailleurs, cela pourait raffiner la représentation distribuée en fonction des objectifs de la tâche d'apprentissage d'ordonnancement.

A moyen terme, nous envisageons d'étendre nos contributions selon les volets suivants :

— Un réseau de neurones qui prend en compte les signaux d'interactions entre la requête et le document. Plusieurs travaux récents de l'état-de-l'art ont montré que les signaux d'interaction sont importants pour une tâche d'appariement document-requête (Guo et al., 2016; Mitra and Craswell, 2018). Une possibilité à explorer est d'exploiter les interactions à plusieurs niveaux, à savoir les interactions entre le texte brut (mot vs mot) et les interactions entre les représentations latentes (*embeddings*). En plus, on peut aussi intégrer la sémantique relationnelle dans le processus de capture des interactions, par exemple, les relations peuvent jouer le rôle de transition entre deux représentations de mots ou concepts (Bordes et al., 2013).
— Nous nous intéressons aussi à une approche combinée entre deux propositions des chapitres 4 et 5 où un réseau de neurones *"end-to-end"* apprend à la fois une fonction d'appariement et les représentations des textes. Nous

pouvons nous inspirer du travail de Severyn and Moschitti (2015) dans lequel un réseau d'appariement avec une couche convolutive reçevant en entrée des matrices de représentations distribuées de textes. Dans ce modèle, on peut modifier la matrice de représentations en même temps avec l'apprentissage d'ordonnancement de deux textes.

A long terme, nous pensons à généraliser nos contributions selon les points suivants :

— Dans le contexte des travaux actuels, nous avons considéré toutes les relations de même type, sans distinction ni dans leur nature, ni de leur degré d'importance pour évaluer la similarité entre textes. Or les ressources traduisent des relations hiérarchiques, des relations de référence, etc. qui traduisent différents granules de connaissances avec différents niveaux de sémantique. De plus, nous avons considéré l'utilisation d'une ressource unique alors que les travaux de l'état-de-l'art en RI sémantique ont montré que l'utilisation de plusieurs ressources pour capturer la sémantique des textes (Dinh et al., 2013; Darmoni et al., 2009) permet d'améliorer l'efficacité de la RI. A long terme nous envisageons d'exploiter des relations hétérogènes définies dans des ressources différentes. Une solution possible est de privilégier des modèles d'apprentissage neuronaux équipés d'une fonction objectif régularisée avec les proximités/distances non symétriques.

— Nous pensons également à l'utilisation des autres sources d'information qui sont aussi porteuses de sémantique. Par exemple, avec le progrès important dans le domaine vision par ordinateur, les machines sont désormais capables de détecter les objets dans une image et de les annoter. Les images fournissent ainsi une compréhension globale du monde (par exemple via les co-ocurrences entre objets) qui permettraient d'augmenter la sémantique latente capturée par les méthodes distributionnelles textuelles. Cette perspective fait référence au "language grounding". Dans cet esprit, certains travaux ont démontré la complémentarité du texte et de l'image en mettant en avant le biais qu'il existe entre ces deux modalités (Glenberg and Kaschak, 2002; Gordon and Van Durme, 2013). En effet, les textes sont généralement porteurs de sémantique globale ou racontent généralement des évènements particuliers alors que les images portent des informations intrinsèques sur les objets et également les relations contextuelles qu'ils entretiennent avec les autres objets. Une perspective intéressante à cette thèse serait d'envisager des approches sémantiques multi-sources capturant la sémantique exprimée à la fois dans le texte, les bases de connaissances et également d'autres modalités porteuses de sens commun, tels que les images et vidéos. Une façon d'envisager cette perspective est d'étudier l'alignement entre ces différentes modalités afin de proposer des modèles de langue multi-modaux à partir de ces différentes sources.

BIBLIOGRAPHIE

Eneko Agirre, Xabier Arregi, and Arantxa Otegi. Document expansion based on wordnet for robust ir. In *ICCL*, pages 9–17, 2010.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. Semeval-2015 task 2 : Semantic textual similarity, english, spanish and pilot on interpretability. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 252–263, 2015.

Qingyao Ai, Liu Yang, Jiafeng Guo, and W. Bruce Croft. Improving language estimation with the paragraph vector model for ad-hoc retrieval. In *SIGIR*, pages 869–872. ACM, 2016a.

Qingyao Ai, Liu Yang, Jiafeng Guo, and W Bruce Croft. Analysis of the paragraph vector model for information retrieval. In *ICTIR*, pages 133–142. ACM, 2016b.

Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4) :357–389, 2002.

Massih-Reza Amini and Nicolas Usunier. A contextual query expansion approach by term clustering for robust text summarization. In *Document Understanding Conference (DUC)*, pages 48–55, 2007.

Alan R Aronson. Effective mapping of biomedical text to the umls metathesaurus : the metamap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association, 2001.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. 2016.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia : A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.

Collin F Baker, Charles J Fillmore, and John B Lowe. The berkeley framenet project. In *Proceedings of the 17th international conference on Computational linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics, 1998.

Saeid Balaneshin-kordan and Alexander Kotov. Optimization method for weighting explicit and latent concepts in clinical decision support queries. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, pages 241–250. ACM, 2016.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL*, pages 238–247, 2014.

Hannah Bast, Björn Buchhold, Elmar Haussmann, et al. Semantic search on text and knowledge bases. *Foundations and Trends® in Information Retrieval*, 10(2-3) : 119–271, 2016.

Mustapha Baziz, Nathalie Aussenac-Gilles, and Mohand Boughanem. Désambiguisation et expansion de requêtes dans un sri. *Revue des Sciences et Technologies de l'Information (RSTI), Hermes*, 8(4) :113–136, 2003.

Mustapha Baziz, Mohand Boughanem, and Nathalie Aussenac-Gilles. Conceptual indexing based on document content representation. In *Context : nature, impact, and role*, pages 171–186. Springer, 2005.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb) :1137–1155, 2003.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan) :993–1022, 2003.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5 :135–146, 2017. ISSN 2307-387X.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795, 2013.

Mohand Boughanem. *Les systèmes de recherche d'informations d'un modèle classique à un modèle connexioniste*. PhD thesis, Toulouse 3, 1992.

Rebecca Bruce and Janyce Wiebe. Word-sense disambiguation using decomposable models. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 139–146. Association for Computational Linguistics, 1994.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association*

for Computational Linguistics : Long Papers-Volume 1, pages 136–145. Association for Computational Linguistics, 2012.

Lou Burnard. Users reference guide british national corpus version 1.0. 1995.

Yonggang Cao, Feifan Liu, Pippa Simpson, Lamont Antieau, Andrew Bennett, James J Cimino, John Ely, and Hong Yu. Askhermes : An online question answering system for complex clinical questions. *Journal of biomedical informatics*, 44 (2) :277–288, 2011.

Claudio Carpineto, Renato De Mori, Giovanni Romano, and Brigitte Bigi. An information-theoretic approach to automatic query expansion. *ACM Transactions on Information Systems (TOIS)*, 19(1) :1–27, 2001.

Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1 : Semantic textual similarity - multilingual and cross-lingual focused evaluation. *CoRR*, abs/1708.00055, 2017.

Carlo Abi Chahine, Nathalie Chaignaud, Jean-Philippe Kotowicz, and Jean-Pierre Pécuchet. Conceptual indexing of documents using wikipedia. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 195–202. IEEE Computer Society, 2011.

Gael de Chalendar, Tiphaine Dalmas, F Elkateb-Gara, Olivier Ferret, Brigitte Grau, Martine Hurault-Plantet, Gabriel Illouz, Laura Monceaux, Isabelle Robba, and Anne Vilnat. The question answering system qalc at limsi, experiments in using web and wordnet. In *Proceedings of TREC*, volume 11, 2002.

Jean Charlet, Bruno Bachimont, and Raphaël Troncy. Ontologies pour le web sémantique. *Revue I3, numéro Hors Série "Web sémantique"*, 2004.

Wei Chen, Tie yan Liu, Yanyan Lan, Zhi ming Ma, and Hang Li. Ranking measures and loss functions in learning to rank. In *NIPS*, pages 315–323. 2009.

Jianpeng Cheng, Zhongyuan Wang, Ji-Rong Wen, Jun Yan, and Zheng Chen. Contextual text understanding in distributional semantic space. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 133–142. ACM, 2015.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*, 2014.

Edward Choi, Mohammad Taha Bahadori, Elizabeth Searles, Catherine Coffey, Michael Thompson, James Bost, Javier Tejedor-Sojo, and Jimeng Sun. Multi-layer

representation learning for medical concepts. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1495–1504. ACM, 2016.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing : Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

Alexis Conneau and Douwe Kiela. Senteval : An evaluation toolkit for universal sentence representations. *arXiv preprint arXiv :1803.05449*, 2018.

Francesco Corcoglioniti, Mauro Dragoni, Marco Rospocher, and Alessio Palmero Aprosio. Knowledge extraction for information retrieval. In *International Semantic Web Conference*, pages 317–333. Springer, 2016a.

Francesco Corcoglioniti, Marco Rospocher, and Alessio Palmero Aprosio. A 2-phase frame-based knowledge extraction framework. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, pages 354–361. ACM, 2016b.

Courtney Corley and Rada Mihalcea. Measuring the semantic similarity of texts. In *Workshop on empirical modeling of semantic equivalence and entailment*, pages 13–18. ACL, 2005.

Fabio Crestani. Exploiting the similarity of non-matching terms at retrieval time. *Information Retrieval*, 2(1) :27–47, 2000.

W Bruce Croft and John Lafferty. *Language modeling for information retrieval*, volume 13. Springer Science & Business Media, 2013.

Andrew M Dai, Christopher Olah, and Quoc V Le. Document embedding with paragraph vectors. *arXiv preprint arXiv :1507.07998*, 2015.

Stéfan J Darmoni, Suzanne Pereira, Saoussen Sakji, Tayeb Merabti, Élise Prieur, Michel Joubert, and Benoit Thirion. Multiple terminologies in a health portal : automatic indexing and information retrieval. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 255–259. Springer, 2009.

Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in neural information processing systems*, pages 2933–2941, 2014.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6) :391–407, 1990.

Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. Neural ranking models with weak supervision. *arXiv :1704.08803*, 2017.

Fernando Diaz, Bhaskar Mitra, and Nick Craswell. Query expansion with locally-trained word embeddings. *arXiv preprint arXiv :1605.07891*, 2016.

Duy Dinh and Lynda Tamine. Towards a context sensitive approach to searching information based on domain specific knowledge sources. *Web Semantics : Science, Services and Agents on the World Wide Web*, 12 :41–52, 2012.

Duy Dinh, Lynda Tamine, and Fatiha Boubekeur. Factors affecting the effectiveness of biomedical document indexing and retrieval based on terminologies. *Artificial intelligence in medicine*, 57(2) :155–167, 2013.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora : Exploiting massively parallel news sources. In *COLING 2004, 20th International Conference on Computational Linguistics, Proceedings of the Conference, 23-27 August 2004, Geneva, Switzerland*, 2004.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul) :2121–2159, 2011.

T. Edinger, AM. Cohen, S.Bedrick, K. Ambert K, and W. Hersh W. Barriers to retrieving patient information from electronic health record data : Failure analysis from the trec medical records track. In *AMIA Annual Symposium*, pages 180–188, 2012.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. Retrofitting word vectors to semantic lexicons. pages 1606–1615, 2015.

Christiane Fellbaum. Wordnet : An electronic lexical database. *Cambridge, MA : MIT Press*, 1998.

Paolo Ferragina and Ugo Scaiella. Tagme : on-the-fly annotation of short text fragments (by wikipedia entities). In *CIKM*, pages 1625–1628. ACM, 2010.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context : The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM, 2001.

John R Firth. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 1957.

Winthrop Nelson Francis. *A manual of information to accompany A standard sample of present-day edited American English, for use with digital computers*. Department of Linguistics, Brown University, 1971.

Gaihua Fu, Christopher B Jones, and Alia I Abdelmoty. Ontology-based spatial query expansion in information retrieval. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 1466–1482. Springer, 2005.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. Ppdb : The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 758–764, 2013.

Goran Glavaš and Ivan Vulić. Explicit retrofitting of distributional word vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, volume 1, pages 34–45, 2018.

Arthur M Glenberg and Michael P Kaschak. Grounding language in action. *Psychonomic bulletin & review*, 9(3) :558–565, 2002.

Julien Gobeill, Patrick Ruch, and Xin Zhou. Query and document expansion with medical subject headings terms at medical imageclef 2008. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 736–743. Springer, 2008.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Travis R Goodwin and Sanda M Harabagiu. Medical question answering for clinical decision support. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 297–306. ACM, 2016.

Jonathan Gordon and Benjamin Van Durme. Reporting bias and knowledge acquisition. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 25–30. ACM, 2013.

Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2) :199–220, 1993.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 55–64. ACM, 2016.

Shashank Gupta, Priya Radhakrishnan, Manish Gupta, Vasudeva Varma, and Umang Gupta. Enhancing categorization of computer science research papers

using knowledge bases. In *Proceedings of the First Workshop on Knowledge Graphs and Semantics for Text Retrieval and Analysis (KG4IR)*, 2017.

Lushan Han, Abhay L Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. Umbc_ebiquity-core : semantic textual similarity systems. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1 : Proceedings of the Main Conference and the Shared Task : Semantic Textual Similarity*, volume 1, pages 44–52, 2013.

Zellig S Harris. Distributional structure. *Word*, 10(2-3) :146–162, 1954.

Taher H Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526. ACM, 2002.

William Hersh, Chris Buckley, T. J. Leone, and David Hickam. Ohsumed : An interactive retrieval evaluation and new large test collection for research. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, pages 192–201. Springer-Verlag New York, Inc., 1994.

William Hersh, Susan Price, and Larry Donohoe. Assessing thesaurus-based query expansion using the umls metathesaurus. In *Proceedings of the AMIA Symposium*, page 344. American Medical Informatics Association, 2000.

Felix Hill, Roi Reichart, and Anna Korhonen. Simlex-999 : Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4) : 665–695, 2015.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. Learning distributed representations of sentences from unlabelled data. In *Proceedings of NAACL-HLT*, pages 1367–1377, 2016.

Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc., 1999.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, pages 2042–2050, 2014.

Gao Huang, Chuan Guo, Matt J Kusner, Yu Sun, Fei Sha, and Kilian Q Weinberger. Supervised word mover's distance. In *Advances in Neural Information Processing Systems*, pages 4862–4870, 2016.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. Learning deep structured semantic models for web search using clickthrough

data. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 2333–2338. ACM, 2013.

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. Sensembed : Learning sense embeddings for word and relational similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, volume 1, pages 95–105, 2015.

Nal Kalchbrenner and Phil Blunsom. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, 2013.

Tom Kenter, Alexey Borisov, and Maarten de Rijke. Siamese cbow : Optimizing word embeddings for sentence representations. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 941–951, 2016.

Douwe Kiela, Felix Hill, and Stephen Clark. Specializing word embeddings for similarity or relatedness. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2044–2048, 2015.

Benjamin King. Step-wise clustering procedures. *Journal of the American Statistical Association*, 62(317) :86–101, 1967.

Diederik P Kingma and Jimmy Ba. Adam : A method for stochastic optimization. *arXiv preprint arXiv :1412.6980*, 2014.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302, 2015.

Bevan Koopman, Guido Zuccon, Peter Bruza, Laurianne Sitbon, and Michael Lawley. Information retrieval as semantic inference : A graph inference model applied to medical search. *Information Retrieval*, 19(1-2) :6–37, 2016.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *International Conference on Machine Learning*, pages 957–966, 2015.

Victor Lavrenko and W Bruce Croft. Relevance-based language models. In *ACM SIGIR Forum*, volume 51, pages 260–267. ACM, 2017.

Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196, 2014.

Thi Hoang Diem Le, Jean-Pierre Chevallet, et al. Thesaurus-based query and document expansion in conceptual indexing with umls. In *RIVF'07*, 2007.

Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet : An electronic lexical database*, 49 :265–283, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553) :436, 2015.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2) :167–195, 2015.

Michael Lesk. Automatic sense disambiguation using machine readable dictionaries : how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM, 1986.

Jerome Y Lettvin, Humberto R Maturana, Warren S McCulloch, and Walter H Pitts. What the frog's eye tells the frog's brain. *Proceedings of the IRE*, 47(11) :1940–1951, 1959.

Lynda Said L'Hadj, Mohand Boughanem, and Karima Amrouche. Enhancing information retrieval through concept-based language modeling and semantic smoothing. *JASIST*, 67(12) :2909–2927, 2016.

Dekang Lin et al. An information-theoretic definition of similarity. In *Icml*, volume 98, pages 296–304. Citeseer, 1998.

Xiaojie Liu, Jian-Yun Nie, and Alessandro Sordoni. Constraining word embeddings by prior knowledge–application to medical information retrieval. In *Asia information retrieval symposium*, pages 155–167. Springer, 2016.

Lajanugen Logeswaran and Honglak Lee. An efficient framework for learning sentence representations. In *International Conference on Learning Representations*, 2018.

Zhengdong Lu and Hang Li. A deep architecture for matching short texts. In *Advances in Neural Information Processing Systems*, pages 1367–1375, 2013.

H. P. Luhn. A Statistical Approach to Mechanized Encoding and Searching of Literary Information. *IBM Journal of Research and Development*, 1957.

Kevin Lund and Curt Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, & computers*, 28(2) : 203–208, 1996.

Thang Luong, Richard Socher, and Christopher Manning. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113, 2013.

Massimiliano Mancini, Jose Camacho-Collados, Ignacio Iacobacci, and Roberto Navigli. Embedding words and senses together via joint knowledge-enhanced training. pages 100–111, 2017.

Donald Metzler and W Bruce Croft. A markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479. ACM, 2005.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013b.

George A. Miller. Wordnet : A lexical database for english. *Commun. ACM*, 38(11) : 39–41, 1995.

George A Miller, Claudia Leacock, Randee Tengi, and Ross T Bunker. A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, pages 303–308. Association for Computational Linguistics, 1993.

David Milne and Ian H. Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *In Proceedings of AAAI 2008*, 2008.

Bhaskar Mitra and Nick Craswell. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval (to appear)*, 2018.

Bhaskar Mitra, Eric Nalisnick, Nick Craswell, and Rich Caruana. A dual embedding space model for document ranking. *arXiv preprint arXiv :1602.01137*, 2016.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1291–1299. International World Wide Web Conferences Steering Committee, 2017.

Andriy Mnih and Geoffrey E Hinton. A scalable hierarchical distributed language model. In *Advances in neural information processing systems*, pages 1081–1088, 2009.

Andriy Mnih and Yee Whye Teh. A fast and simple algorithm for training neural probabilistic language models. *arXiv preprint arXiv :1206.6426*, 2012.

Frederic Morin and Yoshua Bengio. Hierarchical probabilistic neural network language model. In *Aistats*, volume 5, pages 246–252. Citeseer, 2005.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation : a unified approach. *Transactions of the Association for Computational Linguistics*, 2 :231–244, 2014.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. Counter-fitting word vectors to linguistic constraints. *arXiv preprint arXiv :1603.00892*, 2016.

Eric Nalisnick, Bhaskar Mitra, Nick Craswell, and Rich Caruana. Improving document ranking with dual word embeddings. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 83–84. International World Wide Web Conferences Steering Committee, 2016.

Roberto Navigli and Paola Velardi. An analysis of ontology-based query expansion strategies. In *Proceedings of the 14th European Conference on Machine Learning, Workshop on Adaptive Text Extraction and Mining, Cavtat-Dubrovnik, Croatia*, pages 42–49. Citeseer, 2003.

Gia-Hung Nguyen, Laure Soulier, Lynda Tamine, and Nathalie Bricon-Souf. Modèle neuronal de recherche d'information augmenté par une ressource sémantique. In *Conférence francophone en Recherche d'Information et Applications (CORIA 2017)*, 2017a.

Gia-Hung Nguyen, Laure Soulier, Lynda Tamine, and Nathalie Bricon-Souf. Dsrim : A deep neural information retrieval model enhanced by a knowledge resource driven representation of documents. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 19–26. ACM, 2017b.

Gia-Hung Nguyen, Lynda Tamine, Laure Soulier, and Nathalie Souf. Learning concept-driven document embeddings for medical information search. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 160–170. Springer, Cham, 2017c.

Gia-Hung Nguyen, Lynda Tamine, Laure Soulier, and Nathalie Bricon-Souf. Modèle neuronal tripartite pour la représentation de documents. In *Conférence francophone en Recherche d'Information et Applications (CORIA 2018)*, 2018a.

Gia-Hung Nguyen, Lynda Tamine, Laure Soulier, and Nathalie Souf. A tri-partite neural document language model for semantic information retrieval. In *European Semantic Web Conference*, pages 445–461. Springer, Cham, 2018b.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. Integrating distributional lexical contrast into word embeddings for antonym–synonym distinction. In *The 54th Annual Meeting of the Association for Computational Linguistics*, page 454, 2016.

Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. Hierarchical embeddings for hypernymy detection and directionality. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 233–243, 2017d.

Dipasree Pal, Mandar Mitra, and Kalyankumar Datta. Query expansion using term distribution and term association. *arXiv preprint arXiv :1303.0667*, 2013.

Dipasree Pal, Mandar Mitra, and Kalyankumar Datta. Improving query expansion using wordnet. *Journal of the Association for Information Science and Technology*, 65 (12) :2469–2478, 2014.

Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab Ward. Deep sentence embedding using long short-term memory networks : Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(4) : 694–707, 2016.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. Text matching as image recognition. In *AAAI*, pages 2793–2799, 2016.

Ted Pedersen and Varada Kolhatkar. Wordnet : :senserelate : :allwords : A broad coverage word sense tagger that maximizes semantic relatedness. In *NAACL-Demonstrations*, pages 17–20, 2009.

Ted Pedersen, Serguei V. S. Pakhomov, Siddharth Patwardhan, and Christopher G. Chute. Measures of semantic similarity and relatedness in the biomedical domain. *J. of Biomedical Informatics*, 40(3) :288–299, June 2007. ISSN 1532-0464.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove : Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 275–281, New York, NY, USA, 1998. ACM.

Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th international joint conference on Artificial intelligence-Volume 1*, pages 448–453. Morgan Kaufmann Publishers Inc., 1995.

Philip Resnik. Semantic similarity in a taxonomy : An information-based measure and its application to problems of ambiguity in natural language. *Journal of artificial intelligence research*, 11 :95–130, 1999.

Joseph John Rocchio. Relevance feedback in information retrieval. *The SMART retrieval system : experiments in automatic document processing*, pages 313–323, 1971.

Dwaipayan Roy, Debjyoti Paul, Mandar Mitra, and Utpal Garain. Using word embeddings for automatic query expansion. *arXiv preprint arXiv :1606.07608*, 2016.

Herbert Rubenstein and John B Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10) :627–633, 1965.

Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2) :99–121, 2000.

Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv :1609.04747*, 2016.

Gerard Salton. Information Storage and Retrieval. Reports on Analysis, Search, and Iterative Retrieval., 1968.

Aliaksei Severyn and Alessandro Moschitti. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 373–382. ACM, 2015.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 101–110. ACM, 2014a.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 373–374. ACM, 2014b.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

Luca Soldaini and Nazli Goharian. Learning to rank for consumer health search : a semantic approach. In *European Conference on Information Retrieval*, pages 640–646. Springer, 2017.

Nicola Stokes, Yi Li, Lawrence Cavedon, and Justin Zobel. Exploring criteria for successful query expansion in the genomic domain. *Information retrieval*, 12(1) : 17–50, 2009.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations : a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.

Ellen M Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69. Springer-Verlag New York, Inc., 1994.

Ellen M. Voorhees. Overview of the trec 2001 question answering track. In *In Proceedings of the Tenth Text REtrieval Conference (TREC*, pages 42–51, 2001.

Ivan Vulić and Marie-Francine Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 363–372. ACM, 2015.

Ivan Vulić and Nikola Mrkšić. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1134–1145, 2018.

Ivan Vulić, Nikola Mrkšić, Roi Reichart, Diarmuid Ó Séaghdha, Steve Young, and Anna Korhonen. Morph-fitting : Fine-tuning word vector spaces with simple language-specific rules. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, volume 1, pages 56–68, 2017.

Ivan Vulić, Goran Glavaš, Nikola Mrkšić, and Anna Korhonen. Post-specialisation : Retrofitting vectors of words unseen in lexical resources. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, pages 516–527. Association for Computational Linguistics, 2018.

Jörg Waitelonis, Claudia Exeler, and Harald Sack. Linked data enabled generalized vector space model to improve document retrieval. In *Proceedings of NLP & DBpedia 2015 workshop in conjunction with 14th International Semantic Web Conference (ISWC). CEUR-WS*, volume 1486, 2015.

Xiaojun Wan and Yuxin Peng. The earth mover's distance as a semantic measure for document similarity. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 301–302. ACM, 2005.

Chunye Wang and Ramakrishna Akella. Concept-based relevance models for medical and semantic information retrieval. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 173–182. ACM, 2015.

Sida Wang and Christopher D Manning. Baselines and bigrams : Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics : Short Papers-Volume 2*, pages 90–94. Association for Computational Linguistics, 2012.

Xing Wei and W Bruce Croft. Lda-based document models for ad-hoc retrieval. In *SIGIR*, pages 178–185. ACM, 2006.

Jason Weston, Sumit Chopra, and Keith Adams. # tagspace : Semantic embeddings from hashtags. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1822–1827, 2014.

Janyce Wiebe and Claire Cardie. Annotating expressions of opinions and emotions in language. language resources and evaluation. In *Language Resources and Evaluation (formerly Computers and the Humanities*, page 2005, 2005.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank : theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199. ACM, 2008.

Chenyan Xiong and Jamie Callan. Esdrank : Connecting query and documents through external semi-structured data. In *CIKM*, pages 951–960, 2015a.

Chenyan Xiong and Jamie Callan. Query expansion with freebase. In *ICTIR*. ACM, 2015b.

Chang Xu, Yalong Bai, Jiang Bian, Bin Gao, Gang Wang, Xiaoguang Liu, and Tie-Yan Liu. Rc-net : A general framework for incorporating knowledge into word representations. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*, pages 1219–1228. ACM, 2014.

Jinxi Xu and W Bruce Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems (TOIS)*, 18 (1) :79–112, 2000.

Yadollah Yaghoobzadeh and Hinrich Schütze. Intrinsic subspace evaluation of word embedding representations. *arXiv preprint arXiv :1606.07902*, 2016.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. Joint learning of the embedding of words and entities for named entity disambiguation. *CoNLL 2016*, 2016.

Wenpeng Yin and Hinrich Schütze. Convolutional neural network for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 901–911, 2015.

Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. Deep learning for answer sentence selection. *arXiv preprint arXiv :1412.1632*, 2014.

Mo Yu and Mark Dredze. Improving lexical embeddings with semantic knowledge. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, volume 2, pages 545–550, 2014.

Hamed Zamani and W Bruce Croft. Embedding-based query language models. In *Proceedings of the 2016 ACM international conference on the theory of information retrieval*, pages 147–156. ACM, 2016a.

Hamed Zamani and W Bruce Croft. Estimating embedding vectors for queries. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, pages 123–132. ACM, 2016b.

Haïfa Zargayouna, Catherine Roussey, and Jean-Pierre Chevallet. Recherche d'information sémantique : état des lieux. *Traitement Automatique des Langues*, 56(3), 2015.

Matthew D Zeiler. Adadelta : an adaptive learning rate method. *arXiv preprint arXiv :1212.5701*, 2012.

Chengxiang Zhai and John Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 403–410. ACM, 2001.

Ye Zhang, Md Mustafizur Rahman, Alex Braylan, Brandon Dang, Heng-Lu Chang, Henna Kim, Quinten McNamara, Aaron Angert, Edward Banner, Vivek Khetan, et al. Neural information retrieval : A literature review. *arXiv preprint arXiv :1611.06792*, 2016.

Tiancheng Zhao, Kyusong Lee, and Maxine Eskenazi. Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 1098–1107. Association for Computational Linguistics, 2018.

Xiaohua Zhou, Xiaodan Zhang, and Xiaohua Hu. Maxmatcher : Biological concept extraction using approximate dictionary lookup. In *PRICAI*. Springer-Verlag, 2006.

Guido Zuccon, Bevan Koopman, Peter Bruza, and Leif Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proceedings of the 20th Australasian document computing symposium*, page 12. ACM, 2015.