HAL
open science

Modèle Neuronal de Recherche d'Information Augmenté par une Ressource Sémantique

Gia-Hung Nguyen, Laure Soulier, Lynda Tamine, Nathalie Bricon-Souf

To cite this version:

Gia-Hung Nguyen, Laure Soulier, Lynda Tamine, Nathalie Bricon-Souf. Modèle Neuronal de Recherche d'Information Augmenté par une Ressource Sémantique. Conférence francophone en Recherche d'Information et Applications (CORIA 2017), LSIS, Université de Marseille; ARIA : Association Francophone de Recherche d'Information (RI) et Applications, Mar 2017, Marseille, France. pp.1-16. hal-01517032

HAL Id: hal-01517032

https://hal.sorbonne-universite.fr/hal-01517032

Submitted on 2 May 2017

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

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

Modèle Neuronal de Recherche d'Information Augmenté par une Ressource Sémantique

Gia-Hung Nguyen* — **Laure Soulier**** — **Lynda Tamine*** — **Nathalie Bricon-Souf***

** IRIT, Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France, 118 Route Narbonne, Toulouse, France*
*** Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France*

RÉSUMÉ. *De nombreux travaux en recherche d'information (RI) ont montré l'apport de la sémantique des mots pour améliorer l'appariement de document-requête. D'une part, la sémantique symbolique dérivée de ressources externes permet de représenter des entités et leurs relations explicites. D'autre part, la sémantique distributionnelle inférée des corpus permet de représenter les relations sémantiques implicites d'un corpus. Dans cet article, nous proposons de combiner ces deux types de représentations sémantiques. Ainsi, nous présentons un modèle neuronal pour la RI ad-hoc qui exploite les représentations sémantiques latentes des documents et des requêtes en bénéficiant des concepts et des relations exprimés au sein d'une ressource externe. Les évaluations sur deux jeux de données prouvent l'efficacité de notre modèle par rapport aux modèles neuronaux profonds d'appariement de l'état de l'art.*

ABSTRACT. *In information retrieval task, the words semantic has been recognized as significant mean to improve the document-query matching. First, the symbolic semantics extracted from external resources allows to represent entities and their explicit relations. Second, the distributed semantics inferred from the corpus allows to exploit the implicit relations hidden in a corpus. In this paper, we introduce a neural model that leverages the latent semantic representations of documents and queries by taking advantage of the concepts and relations expressed within an external resource. Experimental results obtained on two datasets indicate our model effectiveness in comparison with state-of-the-art deep neural retrieval models.*

MOTS-CLÉS : *Recherche d'information, ressource terminologique, réseau de neurones*

KEYWORDS: *Ad-hoc IR, knowledge resource, deep neural architecture*

1. Introduction

Les approches traditionnelles de la Recherche d'information (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. Cet enjeu a été abordé sous différents angles dont principalement la RI 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. Une lignée de travaux se base sur le principe de la *sémantique relationnelle* en exploitant des ressources sémantiques. Ces dernières peuvent être classées en deux grandes catégories : (1) celles qui représentent des connaissances linguistiques, soit générales (e.g. WordNet), soit axées sur un domaine particulier (e.g. UMLS) 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. 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 et Callan, 2015b), l'expansion de document (Agirre *et al.*, 2010), ou plus récemment, l'utilisation des facteurs enrichis décrivant la relation entre les documents et les requêtes (Xiong et Callan, 2015a).

Une autre lignée de travaux concerne la réduction de l'espace de représentation des documents et des requêtes pour en faire émerger les facteurs d'associations latentes. Ces modèles, comme le LSA[1] (Furnas *et al.*, 1988) ou PLSA[2] basés sur *la sémantique distribuée*, sont capables d'inférer les sens des mots par association à d'autres mots en analysant leurs co-occurrences dans le corpus de documents. Plus récemment, une direction de travaux émergeante permet d'inférer ces associations par apprentissage non supervisé en utilisant un réseau de neurones sur des corpus de données à large échelle. Dans ces méthodes, un mot est projeté sur une représentation latente par un vecteur appelé *word embeddings* qui est capable de capturer la sémantique contextuelle des mots. Plusieurs travaux s'intéressent à la définition de l'algorithme d'apprentissage pour mieux projeter les mots et leurs contextes dans un espace vectoriel réduit (Mikolov *et al.*, 2013 ; Pennington *et al.*, 2014). D'autres travaux s'intéressent à l'utilisation de telles représentations distribuées pour favoriser une tâche de traitement

1. Latent Semantic Analysis
2. Probabilistic Latent Semantic Analysis

automatique du langage naturel (Le et Mikolov, 2014 ; Bengio *et al.*, 2006) ou de RI (Huang *et al.*, 2013 ; Severyn et Moschitti, 2015).

Cependant des travaux antérieurs (Iacobacci *et al.*, 2015) ont montré que ces représentations ne permettent pas d'exprimer les différents sens d'un mot ni ceux qui sont exprimés dans une ressource sémantique. Ainsi, nous supposons que la sémantique distribuée et la sémantique relationnelle sont des approches complémentaires pour représenter les différents sens des mots. Nous examinons alors l'hypothèse selon laquelle la combinaison de ces deux types de sémantiques permettrait d'améliorer la représentation de texte obtenue dans l'espace latent et à terme, améliorer l'appariement requête-document. En adoptant une architecture neuronale, la représentation des textes selon l'approche sémantique relationnelle induit alors la difficulté de représenter un grand nombre de relations objet-objet exprimées dans une ressource sémantique ; pour y répondre, nous proposons une méthode de *hâchage de relations* qui vise à projeter ces relations dans un espace de plus petite dimension. Plus précisément, nous présentons dans cet article les contributions suivantes :

• Un modèle d'appariement requête-document basé sur un réseau de neurones augmenté par des sources d'évidence (entités et relations) issues d'une ressource sémantique. Ce modèle est capable d'exploiter à la fois la sémantique distribuée du texte dans la collection et les caractéristiques sémantiques relationnelles fournies par une ressource sémantique. Ces caractéristiques relationnelles sont représentées par une technique de *hâchage de relations*.

• Une évaluation expérimentale utilisant deux jeux de données TREC, à savoir TREC PubMed CDS et TREC GOV2 Terabyte et deux ressources sémantiques, respectivement MeSH[3] et WordNet[4]. A la différence des travaux précédents qui utilisent les textes courts (e.g. titre du document) (Huang *et al.*, 2013 ; Severyn et Moschitti, 2015), nos expérimentations utilisent les textes intégraux des documents.

2. Travaux connexes

2.1. *Utilisation de réseaux neuronaux profonds en RI*

Récemment, de nombreux travaux ont montré que les approches d'apprentissage par les réseaux de neurones profonds sont très efficaces dans plusieurs tâches de RI (e.g. appariement de textes (Bengio *et al.*, 2006 ; Huang *et al.*, 2013), et tâches de question-réponse (Bordes *et al.*, 2014)). Une première catégorie de travaux utilise des représentations distribuées de mots (*word embeddings*) obtenues par les modèles neuronaux pour les intégrer dans les fonctions d'appariement requête-document (Ai *et al.*, 2016a ; Mitra *et al.*, 2016).

3. https://www.nlm.nih.gov/mesh/
4. http://wordnet.princeton.edu

La deuxième catégorie de travaux consiste en des modèles d'appariement qui apprennent la pertinence des paires document-requête à partir des vecteurs sémantiques latents (Huang *et al.*, 2013 ; Hu *et al.*, 2014). Par exemple, le *Deep Semantic Structured Model* (DSSM) (Huang *et al.*, 2013) est connu pour être un des modèles les plus efficaces dans la tâche de recherche sur le Web. Ce modèle applique un réseau de neurones profond sur la représentation d'un document et d'une requête obtenues par une méthode de hâchage de mots qui permet d'apprendre leurs représentations latentes à partir de leur valeur de pertinence. Une extension du modèle DSSM, proposée dans (Shen *et al.*, 2014) s'appuie sur un réseau de convolution, appelée *Convolutional Latent Semantic Model* (CLSM). Dans le même contexte, Severyn et Moschitti (Severyn et Moschitti, 2015) utilisent une couche de convolution au niveau de la couche d'entrée pour apprendre la représentation optimale des paires de textes au travers d'une fonction de similarité. Au lieu de focaliser sur l'apprentissage d'une représentation complexe, une autre lignée de travaux vise plutôt à construire 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 les mots. Par exemple, le modèle DeepMatch (Lu et Li, 2013) intègre un modèle thématique probabiliste dans un réseau profond entièrement connecté appliqué sur la matrice d'appariements de mots.

2.2. *Utilisation de ressources de connaissance en RI*

Les ressources linguistiques générales/spécifiques (e.g. WordNet ou UMLS respectivement) et les graphes de connaissances (e.g. Freebase) représentent des ressources externes qui fournissent des informations pertinentes sur la sémantique des mots à travers des objets (e.g. des mots, des entités ou des concepts) et leurs relations associées (e.g. «est-un», «partie-de»). Basé sur l'utilisation de ces ressources, une première catégorie de travaux en RI proposent des méthodes d'expansion des requêtes (Xiong et Callan, 2015b ; Pal *et al.*, 2014 ; Bai *et al.*, 2005) ou des documents (Agirre *et al.*, 2010) pour estimer la probabilité de vraisemblance des mots entre les requêtes et les documents. Parmi les modèles d'expansion de requête, Xiong et Callan. (Xiong et Callan, 2015b) proposent deux algorithmes basés sur la catégorisation de mots dans FreeBase. Tandis que l'approche non supervisée estime la similarité entre la distribution des catégories de mots dans les documents et les requêtes, l'approche supervisée exploite une vérité terrain pour estimer l'influence des mots. Pal et al. (Pal *et al.*, 2014) proposent une technique d'expansion de requête utilisant des mots extraits de plusieurs sources d'information. Pour chaque mot de la requête, les mots candidats dans les premiers documents renvoyés pour une requête sont classés en fonction de leur importance dans les documents pseudo-pertinents et de leur similarité sémantique en fonction de leur définition dans WordNet. Par ailleurs, Agirre et al. (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 les plus connexes. La deuxième catégorie de travaux exploite des relations modélisées dans les ressources sémantiques pour l'ordonnancement des documents (Xiong et Callan, 2015a). 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'ordonnancements 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.

3. Le Modèle neuronal de RI

Nous décrivons dans cette partie notre modèle neuronal conçu pour une tâche de RI *ad-hoc* en 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. Plus précisément, à partir d'un document ou d'une requête, notre réseau de neurones vise à projeter la représentation combinée initiale du document ou de la requête sur un espace latent. Ces vecteurs sont obtenus en apprenant la pertinence entre une requête et un document. La Figure 1 illustre l'architecture de notre réseau de neurones qui apprend simultanément les représentations sémantiques latentes d'un document et d'une requête dans le but d'estimer leur pertinence. La représentation d'un document/requête inclut à la fois une représentation du texte brut et une représentation basée sur des ressources sémantiques. Cette architecture à deux branches est utilisée dans plusieurs contributions en RI (Huang *et al.*, 2013 ; Severyn et Moschitti, 2015 ; Hu *et al.*, 2014). A la différence de modèles s'appuyant sur des caractéristiques du texte brut, nous utilisons une couche d'entrée basée sur la composition de deux types d'entrée : les mots d'une part et les concepts et relations entre concepts d'autre part.

3.1. *Architecture du modèle*

3.1.1. *Vecteur d'entrée*

Pour chaque texte T (qu'il soit extrait d'un document ou d'une requête), un vecteur d'entrée $x_{input} = (x^t, x^o, x^{or})$ est représenté comme un vecteur à trois composants :

• *Représentation de texte brut* x^t. Ce composant représente les mots du texte intégral T. En se basant sur des travaux existants qui mettent en évidence l'efficacité des représentations sémantiques distribuées, nous proposons d'estimer un vecteur sémantique de dimension réduite en utilisant le modèle *ParagraphVector* (Le et Mikolov, 2014).

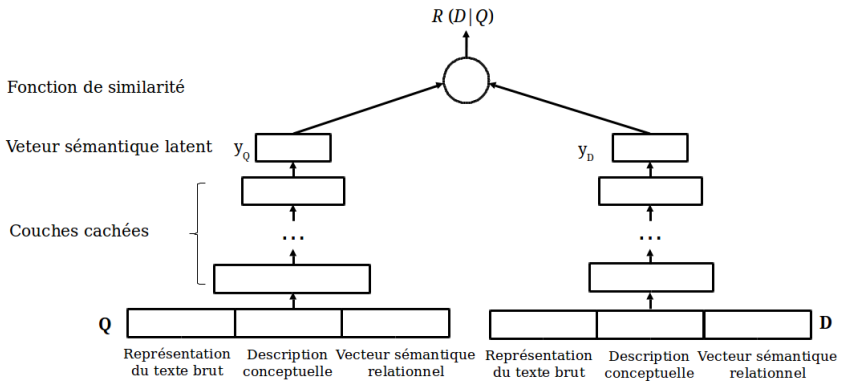

Figure 1. *Architecture du réseau.*

• *Représentation des descripteurs des objets sémantiques* x^o. Ce composant représente les objets de la ressource sémantique identifiés dans le texte T en utilisant les termes préférés, définis par la ressource. Les termes préférés d'un objet sont des ensembles de mots qui font référence au concept représenté par l'objet. De façon similaire à la représentation en texte brut x^t, nous construisons un vecteur sémantique à dimension réduite à l'aide du modèle *ParagraphVector*.

• *Représentation sémantique relationnelle* x^{or}. Ce composant représente les relations entre les objets de la ressource sémantique identifiés dans le texte T. Cette représentation est construite en utilisant une méthode de *hâchage de relations*.

Les deux dernières représentations concernant la sémantique du texte ainsi que la méthode de *hâchage de relations* sont détaillées dans la section 3.2.

3.1.2. *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 :

$$\begin{aligned} l_0 &= x_{input} \\ l_i &= f(W_{i-1} \cdot l_{i-1} + b_{i-1}) \qquad i = 1, ..., L \\ y &= f(W_L \cdot l_L + b_L) \end{aligned} \tag{1}$$

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) : $f(x) = max(0, x)$.

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 $R(D|Q)$ est calculé.

3.1.3. *Fonction de coût*

Comme la tâche de RI *adhoc* concerne un problème d'ordonnancement, nous optimisons les paramètres du réseau de neurones en utilisant un coût d'ordonnancement relatif, basée sur la distance de similarité Δ entre des paires de document-requête pertinentes, 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 :

$$\Delta = \sum_{p=1}^{n} \left[sim(Q, D^+) - sim(Q, D_p^-) \right] \qquad [2]$$

où $sim(\bullet, \bullet)$ et la sortie du réseau de neurones. Comme $sim(\bullet, \bullet) \in [-1, 1]$, l'amplitude de Δ est $[-2n, 2n]$.

Ensuite, le réseau est entraîné pour maximiser la distance de similarité Δ utilisant la fonction de coût "en coude" (*hingeloss*) L, bien adapté pour les tâches d'apprentissage d'ordonnancement (Chen *et al.*, 2009) :

$$L = max(0, \alpha - \Delta) \qquad [3]$$

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

3.2. *Représentation vectorielle de la sémantique relationnelle*

Notre objectif est de représenter la sémantique des documents/requêtes en s'appuyant sur une ressource sémantique externe. L'intuition liée à notre proposition de représentation repose sur les hypothèses suivantes : (H1) un texte est un sac d'objets identifiés à partir d'une ressource sémantique, et (H2) des textes sémantiquement similaires comportent des objets similaires/connexes.

Formellement, une ressource sémantique est un graphe relationnel $G = (V, E)$ où V est un ensemble de nœuds et E est un ensemble d'arêtes entre

ces nœuds. Chaque nœud $v_i =< o_i, desc_i, >$ est une paire d'objets o_i (e.g. mot, entité) et son étiquette textuelle $desc_i$ (e.g. termes préférés). Etant donné un ensemble O des objets dans la ressource sémantique G, nous pouvons identifier, pour chaque texte T, un ensemble d'objets $O(T) \subset O$. En accord avec nos hypothèses H1 et H2, nous proposons la représentation suivante :

1) Chaque vecteur d'entrée x^{input} inclut une représentation x^o des descripteurs textuels $desc_i$ des objets $o_i \in O(T)$ (hypothèse H1).

2) Pour répondre à la contrainte de similarité (hypothèse H2), nous représentons les similarités sémantiques entre les objets $o_i \in O(T)$ dans le texte T. Une approche naïve pour représenter de telles relations consiste à utiliser un vecteur binaire encodant la présence/absence de toutes les relations. Etant donné le grand nombre de relations objet-objet dans la ressource sémantique, nous proposons la méthode de *hâchage de relations*. De la même manière que la méthode de *hâchage de mots* basés sur les trigrammes (Huang *et al.*, 2013), notre méthode vise à réduire la dimension du vecteur sémantique relationnel en utilisant une dimension inférieure ou égale au nombre total d'objets dans la ressource sémantique (qui est beaucoup plus faible que le nombre de relations). Nous proposons de projeter les relations objet-objet sur un ensemble d'objets représentatifs $\mathcal{R}$ (tels que $|\mathcal{R}| < |O|$). En construisant le référentiel d'objets $\mathcal{R}$, nous estimons la similarité sémantique entre les objets identifiés dans le texte T avec chacun des objets dans le référentiel. Une façon de réduire la taille du référentiel de $|O|$ à $|\mathcal{R}|$ est de s'appuyer sur la structure de la ressource externe. Par exemple, une structure hiérarchique permet de faire une coupe de niveau K. Pour chaque objet représentatif o_i dans le référentiel $\mathcal{R}$, la composante associée $x^{or}(i)$ de la représentation sémantique relationnelle x^{or} est estimée comme la probabilité d'atteindre l' objet o_i de l'ensemble $O(T)$ des entités identifiées dans le texte T comme suit :

$$x^{or}(i) = \sum_{o_j \in O(T)} \log P(o_j \mid o_i) = \sum_{o_j \in O(T)} \log \frac{sim(o_i, o_j)}{\sum_{o_k \in O(T)} sim(o_k, o_i)} \qquad [4]$$

où $sim(o_i, o_j)$ est une mesure de similarité sémantique entre les objets o_i et o_j. Nous utilisons la similarité Leacock & Chodorow (Leacock et Chodorow, 1998) qui est basée sur le chemin le plus court entre les objets de la ressource sémantique. Par conséquent, plus les objets identifiés dans les documents sont similaires/connexes, plus les documents sont similaires.

4. Evaluation expérimentale

4.1. *Jeux de données*

Notre modèle est évalué sur deux jeux de données dont les statistiques sont présentées dans le Tableau 1 :

• Le jeu de données GOV2[5] est une collection (*crawl*) des sites .gov utilisée dans la campagne TREC Terabyte. Nous utilisons requêtes (*topics*) des campagnes 2004, 2005 et 2006 dont la partie narrative est utilisée comme requête pour entraîner le modèle.

• Le jeu de données PMC OpenAccess[6] qui regroupe le texte intégral biomédical de PubMed utilisé dans la campagne TREC-CDS. La partie "résumé" des sujets des campagnes d'évaluation 2014 et 2015 est utilisé comme des requêtes pour l'entraînement du modèle.

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 la terminologie 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 "EST-UN" (hyponymie ou hyperonymie) que nous exploitons dans nos expérimentations. Pour le corpus PMC, nous utilisons le thésaurus MeSH, version de 2015, construit par la Bibliothèque américaine de médecine (NLM). Cette ressource comprend 27000 concepts, organisés en 16 catégories et structurés hiérarchiquement du plus général au plus spécifique.

Tableau 1. *Statistiques de TREC Terabyte et de TREC-CDS*

	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

4.2. *Détails d'implémentation et protocole d'évaluation*

Pour former le vecteur d'entrée, nous appliquons deux modèles *ParagraphVector*, un sur le corpus de textes pour le vecteur x^t et un autre sur le descripteur conceptuel pour le vecteur x^o. Les deux vecteurs sont de dimension 100, ce qui est cohérent avec les résultats précédents indiquant que les modèles de vecteurs de paragraphes de dimensions réduites sont capables de capturer des structures complexes (Ai *et al.*, 2016a). Pour construire le vecteur d'entrée sémantique relationnel x^{or}, les concepts sont extraits à l'aide d'outils d'extraction, à savoir *SenseRelate* (Pedersen et Kolhatkar, 2009) pour le jeu de données

5. http://ir.dcs.gla.ac.uk/test_collections/gov2-summary.htm
6. https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

GOV2 et Cxtractor[7] basé sur *MaxMatcher* (Zhou *et al.*, 2006) pour le jeu de données PMC. Une fois que les concepts sont identifiés, le vecteur sémantique relationnel x^{or} est construit en utilisant la méthode de *hâchage de relations* avec le niveau du référentiel $K = 4$. Ce seuil représente un compromis entre la spécificité des concepts et la dimension du vecteur, à savoir 3746 et 1096 pour respectivement le corpus GOV2 et le corpus PMC.

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 300 conduisant à une couche de sortie de 128 noeuds. Le nombre n de paires de document-requête non pertinentes opposé à un pertinent est 4 (Equation 2). 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.

Pour évaluer la performance de notre modèle et des différentes modèles de référence, nous réalisons la technique de réordonnacement comme dans (Ai *et al.*, 2016b). Pour cela, les 2000 premiers documents sélectionnés par le modèle BM25 sont retenus et les résultats finaux sont calculés en utilisant les 1000 premiers documents de chaque modèle de réordonnancement selon les métriques suivantes : MAP, P@10 et nDCG@10.

4.3. *Modèles de référence*

Nous utilisons trois types de modèles de référence : les modèles d'appariement exact ($BM25, LM - DI$), les modèles d'appariement basé sur la sémantique latente ($QE, LM - LDA$) et les modèles d'appariement neuronaux profonds ($DSSM, CLSM$) :

- $BM25$: Le modèle probabiliste classique *BM25*.
- $LM - DI$: Le modèle de langue basé sur le lissage de Dirichlet, qui est un

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

autre modèle d'appariement exact.
• $LM - QE$: 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.
• $LM - LDA$: Un modèle latent utilisant le modèle de langue (Wei et Croft, 2006). Par souci de comparabilité, nous avons fixé le nombre de sujets (*topics*) égaux à la taille du vecteur de sortie y dans notre modèle, soit 128.
• $DSSM$: 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[8] avec les valeurs par défaut des paramètres. Nous évaluons le modèle DSSM sur les documents en texte intégral.
• $CLSM$: L'extension du modèle DSSM dans lequel le réseau neuronal est remplacé par un réseau de convolution pour mieux capturer des structures contextuelles détaillées (Shen *et al.*, 2014). Nous utilisons également le code CLSM public[8] sur les documents en texte intégral avec les valeurs des paramètres par défaut.

5. Résultats expérimentaux

5.1. *Analyse de l'efficacité*

Dans cette section, nous nous intéressons à l'évaluation des performances de notre modèle sur les deux jeux de données GOV2 et PMC. Le Tableau 2 présente l'efficacité en termes des mesures MAP, P@10 et NDCG@10 pour notre modèle et les différents modèles de référence.

D'un point de vue général, nous pouvons constater d'une part que les modèles classiques d'appariement (à savoir, BM25 et LM-DI) fournissent des résultats significativement meilleurs que notre modèle. En effet, ces modèles obtiennent respectivement une MAP de $0,1777$ et de $0,1584$ pour le jeux de données GOV2 tandis que notre modèle obtient $0,1197$. D'autre part, notre approche dépasse les modèles sémantiques ($LM - QE$ et $LM - LDA$) et les modèles neuronaux profonds ($DSSM$ et $CLSM$) avec des améliorations significatives. Par exemple, notre modèle présente des résultats significativement plus élevés pour le jeu de données GOV2 par rapport aux modèles LM-QE, DSSM et CLSM, caractérisés respectivement par une valeur de MAP égale à $0,0738$, $0,0418$ et $0,0365$. Ces observations sont similaires pour les deux jeux de données, soulignant le fait que notre modèle est efficace dans l'exploitation des ressources générales (WordNet) ainsi que des ressources spécifiques à

8. https://www.microsoft.com/en-us/research/project/dssm/

Tableau 2. *Comparaison de l'efficacité des modèles sur les jeux de données GOV2 et PMC. Amélioration/dégradation significative de notre modèle p.r.à chaque modèle est indiquée (+/−) (p-value≤ 0.05)*

	GOV2			PMC		
Model	MAP	P@10	nDCG@10	MAP	P@10	nDCG@10
BM25	$0,1777^{-}$	$0,4376^{-}$	$0,36^{-}$	0,037	$0,2033^{-}$	$0,1824^{-}$
LM-DI	$0,1584^{-}$	$0,406^{-}$	$0,3243^{-}$	0,0388	$0,1967^{-}$	$0,1827^{-}$
LM-QE	$0,0738^{+}$	$0,1477^{+}$	$0,1169^{+}$	$0,0106^{+}$	0,0767	0,0738
LM-LDA	0,0966	$0,1651^{+}$	0,1278	0,0185	0,1067	0,1025
DSSM	$0,0418^{+}$	0,2403	0,1838	$0,0125^{+}$	$0,0533^{+}$	0,0521
CLSM	$0,0365^{+}$	0,2007	0,1496	$0,0114^{+}$	$0,034^{+}$	$0,0201^{+}$
Notre modèle	0,1197	0,2329	0,1702	0,0233	0,0967	0,0737

un domaine (MeSH). Plus particulièrement, nous pouvons observer les points suivants :

• Le modèle BM25 et les modèles de langue sont connus comme des modèles de référence en RI. Il est difficile d'obtenir de meilleurs résultats dans le cas des modèles d'appariement neuronaux entraînés sur des petits jeux de données comparativement aux jeux de données à grande échelle, souvent propriétaires, comme ceux utilisés dans (Huang *et al.*, 2013). En effet, l'apprentissage des représentations latentes du texte au travers d'architectures neuronales profondes requiert l'optimisation d'un grand nombre de paramètres. Par conséquent, l'efficacité de notre modèle est sous-optimisée en raison du faible nombre de requêtes de nos jeux de données GOV2 et PMC, respectivement 150 et 60 requêtes. Ce constat nous suggère d'expérimenter à moyen terme notre modèle sur un jeu de données plus important. En outre, il est important de souligner qu'à la différence de la majorité des approches neuronales de l'état de l'art (Guo *et al.*, 2016 ; Huang *et al.*, 2013 ; Severyn et Moschitti, 2015 ; Shen *et al.*, 2014) qui effectuent l'appariement entre les titres des documents et les requêtes, nous avons expérimenté notre modèle sur le contenu intégral des documents (la longueur moyenne d'un document est de 1132.8 et 477.1 mots pour respectivement GOV2 et PMC). Ce choix d'expérimentation renforce notre intuition sur la nécessité d'apprendre les représentations latentes sur des jeux de données plus grands.

• Notre modèle présente des accroissements significatifs par rapport au modèle LM-QE. Ce résultat suggère que les représentations sémantiques latentes des documents et des requêtes basées sur la méthode de *hâchage de relations* sont plus efficaces pour l'appariement que les requêtes étendues textuellement avec des concepts pertinents.

• L'efficacité de notre modèle est généralement plus élevée que celle du modèle LDA-LM, avec par exemple une amélioration significative de 22% pour la

Tableau 3. *Similarité moyenne des paires de requêtes de documents pertinentes aux couches d'entrée et de sortie*

	GOV2		PMC	
	Entrée	Sortie	Entrée	Sortie
DSSM	0,1482	0,3955	0,1049	0,1111
Notre Modèle	0,1916	0,7369	0,1413	0,4436

métrique P@10 sur le jeu de données GOV2. Ceci est cohérent avec le travail précédent (Huang *et al.*, 2013) qui met en évidence l'efficacité des représentations latentes des textes obtenues par un modèle neuronal par rapport à celles obtenues par les modèles probabilistes de type LDA.

• L'efficacité de notre modèle dépasse les modèles de l'état de l'art basés sur des architectures neuronales, à savoir DSSM et CLSM. Ces résultats suggèrent que l'intégration de la sémantique relationnelle des mots dans la couche d'entrée permet d'améliorer l'apprentissage du modèle d'appariement neuronal. Curieusement, le modèle de convolution CLSM initialement expérimenté sur un jeu de données à grande échelle et plus efficace que le modèle DSSM, est moins efficace que le modèle DSSM lorsqu'il est utilisé sur des collections plus petites. Ce constat est également identifié dans (Guo *et al.*, 2016). Cette dernière observation, combinée aux observations issues de la comparaison de notre modèle aux modèles de référence BM25 et LM-DI, ouvre de nombreuses perspectives de recherche en RI neuronale en termes d'apprentissage de représentation sur de petits jeux de données en introduisant par exemple des approches de supervision distante.

5.2. *Analyse des représentations latentes des documents et requêtes*

Afin d'étudier l'impact de l'intégration de la ressource sémantique externe dans un modèle d'appariement neuronal, nous proposons de comparer les représentations des documents et requêtes pour notre modèle et le modèle DSSM. Le Tableau 3 présente les mesures de la similarité cosinus entre les paires de document-requête au niveau des vecteurs d'entrée et de sortie pour chacun des modèles. Nous pouvons observer que les similarités des vecteurs document-requête en entrée sont de la même plage de valeurs pour les deux jeux de données et les deux modèles. Pourtant, l'amélioration de la similarité entre les vecteurs d'entrée/sortie est plus importante pour notre modèle (+166.88% et 5.91% pour GOV2 et PMC respectivement) que pour le modèle DSSM (+284.63% et 213.94% pour GOV2 et PMC respectivement). Ces résultats suggèrent que l'utilisation des ressources sémantiques externes, et plus particulièrement la sémantique relationnelle induite par le *hâchage de relations*, permet une meilleure discrimination entre les documents pertinents et non pertinents.

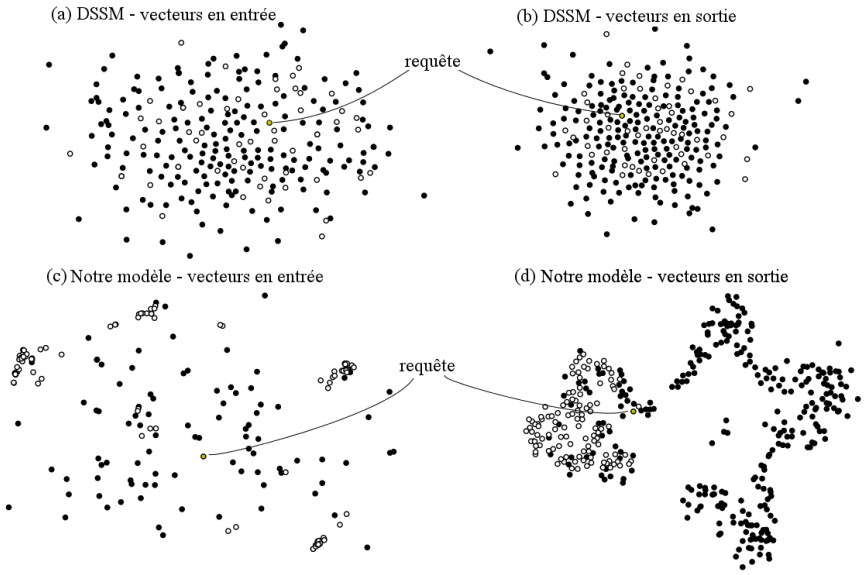

Figure 2. *Analyse comparative de la projection t-SNE entre les vecteurs d'entrée (a)(c) et de sortie (b)(d) pour la requête 707 du jeu de données GOV2.* ○ *: documents pertinents -* ● *: documents non pertinents.*

Ces résultats quantitatifs sont également appuyés par les visualisations des représentations de documents et de requêtes effectuée par une projection t-SNE. La Figure 2 illustre, pour un exemple de requête (707 du jeu de données GOV2), les projections des vecteurs d'entrée et de sortie obtenues par le modèle DSSM (Figure 2. (a), (b)) et par notre modèle (Figure 2. (c), (d)). Nous pouvons observer qu'à l'entrée du réseau les documents pertinents (cercles blancs) et les documents non pertinents (cercles noirs) sont répartis aléatoirement autour de la requête, à la fois pour le modèle DSSM et notre modèle (Figures 2(a) et 2(c) respectivement). Nous pouvons également remarquer que les projections d'entrée de DSSM et de notre modèle ne sont pas similaires, expliqué par le fait que le modèle DSSM considère une représentation de mots (ou trigramme) tandis que nous proposons une représentation augmentée du texte comprenant une composante sémantique. Un autre aspect visible de la Figure 2(c) est que quelques petits groupes de documents pertinents apparaissent déjà au niveau de l'entrée. Cela renforce notre intuition sur le fait que les ressources sémantiques sont utiles pour améliorer les représentations des documents en entrée d'une architecture neuronale profonde. Après les transformations non linéaires effectuées par le modèle DSSM et notre réseau de neurones, nous observons que les projections en sortie du modèle DSSM (Figure 2 (b)) sont centralisées autour de la requête, ce qui laisse supposer que la discrimination des documents pertinents/non pertinents est difficile. En revanche, la projection des documents

et requêtes en sortie de notre modèle (Figure 2(d)) montre que les documents sont clairement regroupés en deux classes, désignant généralement l'ensemble des documents pertinents (à gauche) et l'ensemble des documents non pertinents (à droite)). Cette classification des documents est d'autant plus nette que la requête est visuellement proche du groupe de documents pertinents. Ces résultats qualitatifs corroborent les résultats quantitatifs obtenus à partir de la comparaison des scores de similarité des représentations de documents et de requêtes (Tableau 3).

6. Conclusion

Nous avons présenté dans cet article un modèle neuronal pour la RI *ad-hoc*. La particularité de notre modèle est l'exploitation d'une ressource sémantique externe pour la modélisation des objets et des relations inclus dans les requêtes et documents. Pour résoudre le problème de la dimensionnalité sous-jacente à la représentation des relations entre objets, nous proposons la méthode de *hâchage de relations* basée sur l'hypothèse que des documents similaires comportent des concepts similaires et/ou reliés. Les représentations latentes des documents et requêtes, ainsi que leur appariement sont obtenus à l'aide d'un réseau de neurones. L'évaluation expérimentale sur deux jeux de données TREC, à savoir GOV2 et PMC met en évidence l'efficacité de notre modèle comparativement aux approches orientées sémantique ainsi que les modèles RI neuronaux de l'état de l'art. Dans un futur proche, nous envisageons de poursuivre nos expérimentations sur des jeux de données plus importants, ainsi que des investigations plus profondes pour mieux évaluer l'effet des différentes composantes de notre modèle. 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.

7. Bibliographie

Agirre E., Arregi X., Otegi A., « Document expansion based on WordNet for robust IR », *ICCL*, p. 9-17, 2010.

Ai Q., Yang L., Guo J., Croft W. B., « Analysis of the Paragraph Vector Model for Information Retrieval », *ICTIR*, ACM, p. 133-142, 2016a.

Ai Q., Yang L., Guo J., Croft W. B., « Improving Language Estimation with the Paragraph Vector Model for Ad-hoc Retrieval », *SIGIR*, ACM, p. 869-872, 2016b.

Bai J., Song D., Bruza P., Nie J.-Y., Cao G., « Query expansion using term relationships in language models for information retrieval », *CIKM*, 2005.

Bengio Y., Schwenk H., Senécal J.-S., Morin F., Gauvain J.-L., « Neural probabilistic language models », *Innovations in Machine Learning*, 2006.

Bordes A., Chopra S., Weston J., « Question Answering with Subgraph Embeddings », *EMNLP*, p. 615-620, 2014.

Chen W., yan Liu T., Lan Y., ming Ma Z., Li H., « Ranking Measures and Loss Functions in Learning to Rank », *NIPS*, p. 315-323, 2009.

Furnas G. W., Deerwester S., Dumais S. T., Landauer T. K., Harshman R. A., Streeter L. A., Lochbaum K. E., « Information retrieval using a singular value decomposition model of latent semantic structure », *Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, p. 465-480, 1988.

Guo J., Fan Y., Ai Q., Croft W. B., « A Deep Relevance Matching Model for Ad-hoc Retrieval », *CIKM*, 2016.

Hu B., Lu Z., Li H., Chen Q., « Convolutional neural network architectures for matching natural language sentences », *NIPS*, p. 2042-2050, 2014.

Huang P.-S., He X., Gao J., Deng L., Acero A., Heck L., « Learning deep structured semantic models for web search using clickthrough data », *CIKM*, 2013.

Iacobacci I., Pilehvar M. T., Navigli R., « SensEmbed: Learning Sense Embeddings for Word and Relational Similarity », *ACL*, p. 95-105, 2015.

Le Q. V., Mikolov T., « Distributed Representations of Sentences and Documents. », *ICML*, 2014.

Leacock C., Chodorow M., « Combining local context and WordNet similarity for word sense identification », *WordNet: An electronic lexical database*, vol. 49, n° 2, p. 265-283, 1998.

Lu Z., Li H., « A deep architecture for matching short texts », *NIPS*, 2013.

Mikolov T., Chen K., Corrado G., Dean J., « Efficient estimation of word representations in vector space », *arXiv preprint arXiv:1301.3781*, 2013.

Mitra B., Nalisnick E., Craswell N., Caruana R., « A dual embedding space model for document ranking », *arXiv preprint arXiv:1602.01137*, 2016.

Pal D., Mitra M., Datta K., « Improving query expansion using WordNet », *Journal of the Association for Information Science and Technology*, vol. 65, n° 12, p. 2469-2478, 2014.

Pedersen T., Kolhatkar V., « WordNet::SenseRelate::AllWords: A Broad Coverage Word Sense Tagger That Maximizes Semantic Relatedness », *NAACL-Demonstrations*, p. 17-20, 2009.

Pennington J., Socher R., Manning C., « Glove: Global Vectors for Word Representation », *EMNLP*, p. 1532-1543, 2014.

Severyn A., Moschitti A., « Learning to rank short text pairs with convolutional deep neural networks », *SIGIR*, p. 373-382, 2015.

Shen Y., He X., Gao J., Deng L., Mesnil G., « A latent semantic model with convolutional-pooling structure for information retrieval », *CIKM*, 2014.

Wei X., Croft W. B., « LDA-based document models for ad-hoc retrieval », *SIGIR*, ACM, p. 178-185, 2006.

Xiong C., Callan J., « EsdRank: Connecting Query and Documents Through External Semi-Structured Data », *CIKM*, p. 951-960, 2015a.

Xiong C., Callan J., « Query expansion with Freebase », *ICTIR*, ACM, 2015b.

Zhou X., Zhang X., Hu X., « MaxMatcher: Biological Concept Extraction Using Approximate Dictionary Lookup », *PRICAI*, Springer-Verlag, 2006.