Recherche d'information entre des bases de connaissances

Jean Dupuy

Université de Lyon, Lyon 2 ERIC EA 3083, Meetsys
jean.dupuy@univ-lyon2.fr

RÉSUMÉ. *Dans cet article nous nous intéresserons à la recommandation de contenus, et plus particulièrement au sein de bases de connaissances. Le sujet de thèse présenté ici se focalise sur la représentation de documents textuels en prenant en compte plusieurs échelles (phrase, paragraphe et document entier) et l'exploitation de celle-ci par un système de recommandation, soit au sein de la même base de connaissances, soit entre des bases différentes. Après un détail du corpus considéré pour ces travaux et un état de l'art sur les méthodes d'embedding actuelles, nous discuterons de l'utilité de l'exploitation de la structure du graphe du corpus pour la représentation, puis des perspectives du sujet et de sa contribution à la recherche d'information.*

ABSTRACT. *In this article we will focus on content recommendation, especially in knowledge bases. The Ph.D subject introduced focuses on multiscale text embedding (sentences, paragraphs and whole documents), and recommendation build on top of those representations. Recommendations could be done on a knowledge base or between different ones. After a short presentation of our corpus and a state of the art about current embedding methods we will discuss about the relevance of embedding graph structure, as well as future outlook of our work and its contribution to IR.*

MOTS-CLÉS : *Plongement de mots, Plongement de documents, recherche d'information, base de connaissances, recommandation*

KEYWORDS: *Word Embedding, Document Embedding, Information retrieval, Knowledge base, Recommandation*

DOI:10.3166/..1-9

1. Introduction

Les systèmes de recommandation de documents se sont longtemps appuyés sur des représentations basées sur des *sacs de mots* (*Bag of words*), pondérés (Salton, Buckley, 1988) ou sur des méthodes d'extraction de concepts (Blei *et al*., 2003). Néanmoins ces méthodes peinent à retransmettre les relations entre les mots, et ne capturent que grossièrement les liens entre eux. Depuis quelques années de nouveaux modèles de plongement de mots (*Word embedding*) se sont développés, comme *Word2Vec* (Mikolov, Yih, Zweig, 2013). Ceux-ci ont la particularité de fournir une représentation plus fine des relations entre les mots se basant sur leurs contextes.

L'application de ces nouvelles méthodes de plongement de mots a rapidement été étendue à la représentation de documents textuels, et donc aux systèmes de recommandation de contenu. En effet le possibilité de trouver des relations entre documents non plus basées sur la co-occurrence des termes qui les composent mais sur une proximité sémantique est une amélioration significative.

Cependant les systèmes de recommandation ne recherchent pas uniquement des similarités entre deux documents entiers, et certaines applications requièrent un niveau de granularité plus fin, dans une optique de recherche d'information. En considérant un document comme un agrégat de documents plus courts il devient envisageable de recommander non plus des documents entiers mais seulement des parties de ceux-ci, portant plus d'informations. Ce type d'application trouve une place toute naturelle dans la gestion de bases de connaissances, celles-ci étant constituées de documents portant sur des sujets et des concepts précis, mais dont certains elements peuvent tout de même être liés entre eux.

Nous nous intéresserons lors de la présentation de ce sujet de thèse à l'étude des approches de word embedding et document embedding à différentes échelles, en commençant par exposer le contexte de cette thèse et ses enjeux. Il y sera également fait mention du type de corpus sur lequel se sont basés les premières expériences réalisées. S'en suivra un bref état de l'art sur les méthodes de plongements de mots et de documents non supervisées les plus courantes aujourd'hui, ainsi que quelques détails sur le plongement de graphes. Puis nous discuterons de l'utilité de l'exploitation du graphe de notre corpus, avant de proposer les pistes que nous souhaitons explorer plus avant lors de cette thèse.

2. Contexte et motivations

Le sujet de cette thèse est né d'un besoin industriel. L'entreprise gère des bases de connaissances industrielles, le plus souvent liées à la résolution de problèmes, contenant des fiches issues soit de la capitalisation d'experts de l'entreprise, soit créées et complétées par les utilisateurs eux même. Le besoin de pouvoir recommander des documents à un utilisateur et lui permettre de faire le lien entre différentes informations devient donc un enjeu important.

Chaque base de connaissances est indépendante, et porte sur un domaine précis. Les fiches, des documents textuels techniques ou scientifiques, peuvent être liées manuellement entre elles par les utilisateurs eux mêmes. Ces liens représentent donc une similarité forte entre les sujets, puisque représentant un lien pertinent dans un contexte métier. De plus l'existence de ces liens fait émerger un graphe. Néanmoins puisque ces liens sont produits par les utilisateurs leur nombre est très variable d'une base à une autre.

Nous souhaitons donc mettre en oeuvre une approche de représentation de ces documents en prenant en compte leur structure, c'est à dire représenter dans le même espace le document, les paragraphes qui le constituent, ainsi que les phrases et les mots qui les composent. L'idée générale étant de pouvoir fournir une recommandation la plus fine possible, en ne ciblant que les elements les plus pertinents.

Le second sujet qui motive ce projet de recherche porte sur la recommandation entre différentes bases de connaissances. Comme noté précédemment les bases sont indépendantes, et portent sur des sujets et des domaines qui peuvent être différents. Il peut donc devenir intéressant de pouvoir extraire des similarités entre des documents, quand bien même ceux-ci ne partagent pas le même vocabulaire technique. Certains termes peuvent en effet être conceptuellement proches tout en appartenant à des vocabulaires métiers éloignés.

3. État de l'art de la représentation de documents

Le plongement de mots a été largement exploité ces dernières années, et principalement le modèle Word2Vec (Mikolov, Chen *et al.*, 2013). Ce modèle a rapidement été adopté pour la représentation d'autres types de données, comme les documents ou les graphes, ou adapté dans un cadre probabiliste (Vilnis, McCallum, 2014).

3.1. Plongement de mots

La représentation de textes se base en partie sur l'exploitation du contexte des mots qui les composent. La base de l'hypothèse distributionnelle (Harris, 1954) repose sur l'idée que les mots qui partagent des contextes similaires tendent à être sémantiquement proches. Cette approche est à la base des méthodes que nous allons présenter.

3.1.1. Word2Vec

Word2Vec (Mikolov, Chen *et al.*, 2013 ; Mikolov, Le, Sutskever, 2013) prend la forme d'un réseau de neurones à une couche cachée visant à construire une représentation vectorielle des mots. Il exploite pour cela la notion de contexte d'un mot. Chaque mot est considéré avec les mots apparaissant à proximité de celui-ci. La représentation du mot est donc apprise en prenant en compte son contexte, permettant ainsi de capturer des liens sémantiques forts.

Deux architectures sont possibles: CBOW et Skip-Gram.

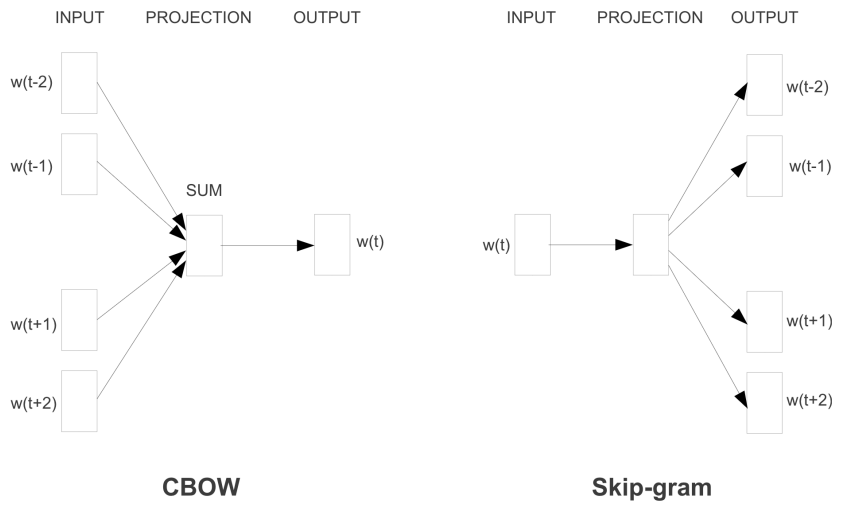

FIGURE *1. Les modèles CBOW et Skip-Gram décrit dans (Mikolov, Le, Sutskever, 2013)*

3.1.1.1. CBOW

CBOW, pour *Continuous Bag-of-words*, apprend une représentation pour prédire un mot à partir de son contexte, c'est à dire les mots présents dans la fenêtre.

3.1.1.2. Skip-Gram

Skip-Gram quant à lui fait l'opération inverse. Il apprend une représentation des mots en tentant de prédire son contexte. Plus formellement, étant donné une suite de mots $w_1, w_2, w_3, \ldots, w_T$ le modèle maximise la moyenne des probabilités des mots qui les entourent:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p\left(w_{t+j}|w_t\right)$$

où c représente la taille de la fenêtre de contexte. La valeur de c influe directement sur la précision des représentations apprises, mais également sur la complexité de l'entraînement.

$$P\left(w_O|w_I\right)=\frac{\exp\left(v'^{T}_{w_O}v_{w_I}\right)}{\sum_{w=1}^{W}\exp\left(v'^{T}_{w}v_{w_I}\right)}$$

où v_w et v'_w sont respectivement les vecteurs de représentation d'entrée et de sortie du mot w, et W la taille du vocabulaire.

3.1.2. FastText

(Bojanowski *et al.*, 2017) enrichie la proposition faite par Skip-Gram en représentant les mots comme la somme des représentations des *n*-grams qui les composent. Ainsi le mot *apple*, quand *n* = 3, sera représenté par <*ap, app, ppl, ple, le*> ainsi que par la sequence complète <*apple*> (les symboles < et > servant à marquer respectivement le début et la fin d'un mot pour pouvoir distinguer prefixes et suffixes). Cette méthode permet également d'obtenir de meilleures représentations des mots rares.

3.1.3. Gaussian Embedding

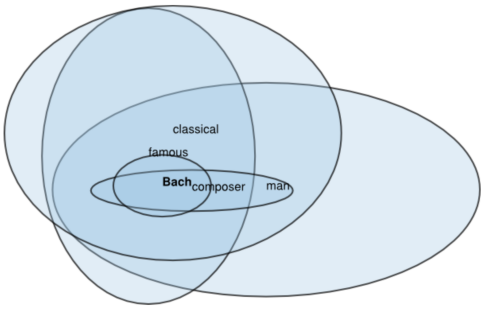

FIGURE 2. *Représentation du gaussian embedding tel que présenté dans (Vilnis, McCallum, 2014)*

(Vilnis, McCallum, 2014) propose une version probabiliste de *Word2Vec*. On estime le vecteur de représentation du mot mais également la loi de probabilité de celui-ci. Ce type de représentation donne la possibilité de capturer la "diffusion" du sens d'un mot. En effet un mot au sens très général tendra à avoir une variance élevée, comparée à un terme plus précis. Il est de plus possible de déterminer des similarités qui ne soient pas symétriques entre les mots, ce qui dans le cas de recommandations au sein de bases de connaissances a un intérêt tout particulier. Si un concept développé dans un document peut être fortement lié à un autre, la réciproque n'est en effet pas nécessairement vraie.

3.1.4. GloVe

GloVe (Pennington *et al.*, 2014) (pour GLObal VEctors) diffère de Word2vec puisque ce n'est pas une méthode prédictive. Elle se base sur la construction d'une matrice de co-occurrence comptant à quelle fréquence un mot apparaît prés d'un autre,

dans une fenêtre glissante. A la difference de Word2Vec, GloVe prend en compte l'information portée par l'entièreté du corpus, et non plus seulement sur une fenêtre de mots. Une fois la matrice de co-occurrence obtenue le modèle apprend la représentation des mots de telle sorte que leur produit scalaire soit égal au logarithme de leur probabilité de co-occurrence.

3.2. Plongement de documents

Les modèles de plongement de mots ont bien évidemment été adapté à la représentation de documents, mais peu de modèles ont finalement été développés. Si il a été montré que sommer ou moyenner les vecteurs de représentations de chaque mots constituant un document fonctionnait, la principale méthode utilisée est Paragraph2Vec (Le, Mikolov, 2014), prolongement directe de Word2Vec. Le modèle diffère effectivement très peu de son équivalent en word embedding. L'idée principale étant d'ajouter à chaque contexte un "faux token", un mot absent du document mais permettant de l'identifier, faisant office de mémoire. Là aussi deux variante existent:

– PV-DM (Distributed Memory Model of Paragraph Vectors): similaire à CBOW présenté plus tôt. On concatène le contexte et le token d'identification du document pour prédire un mot.

– PV-DBOW (Distributed Bag of Words version of Paragraph Vector): similaire lui à Skip-Gram.

S'il a été noté que PV-DM permet d'obtenir la plupart du temps de meilleurs résultats que PV-DBOW lors de tâches de classification, les auteurs constatent que la combinaison des deux représentations offre toujours des résultats au moins aussi bons que l'utilisation d'une seule.

3.3. Plongement de Graphes

Enfin les résultats de représentations continues de mots ont été exploité dans l'apprentissage de représentation de graphes. Des travaux tels que DeepWalk(Perozzi *et al.*, 2014) se basent sur Skip-Gram en utilisant en entrée le résultat de marches aléatoires. L'idée de l'hypothèse distributionnelle demeure, à savoir que les sommets qui partagent des voisins similaires tendent à être proches. En traitant les suites des sommets des marches aléatoires comme des phrases, on peut appliquer *Word2Vec* pour obtenir une représentation des sommets d'un graphe, qui préserve sa structure. DeepWalk a ensuite été étendu à l'étude de graphes plus riches, notamment aux graphes avec des contenus textuels comme sommets.

Le prolongement de graphe exploitant également les évolutions de *Word2Vec*, nous retrouvons des version probabilistes dans (Dos Santos *et al.*, 2016), (He *et al.*, 2015) ou (Bojchevski, Günnemann, 2017).

Il existe bien sûr d'autres méthodes ne se basant pas sur les marches aléatoires. On citera par exemple *Laplacian Eigenmaps* (Belkin, Niyogi, 2002) qui représente les sommets en factorisant la matrice d'adjacence du graphe. D'autres méthodes comme *GCN* (Kipf, Welling, 2017) (Graph Convolutional Networks) tirent profit de l'apprentissage profond et des réseau de neurones récurrents pour apprendre des représentations des sommets.

4. Premiers tests sur le corpus et discussions sur l'utilisation du graphe

Pour vérifier si les liens présents dans le corpus étudié, produit manuellement par les utilisateurs, sont liés à une similarité dans les textes, nous testons différentes méthodes de représentation et de comparer les similarités entre documents et les liens existants. Le corpus considéré est une base de connaissance de 1689 pages décrivant des phénomènes physico-chimiques et 196 liens entre pages.

Les trois méthodes utilisées sont les suivantes:

– Word2Vec avec *Skip-Gram* (Mikolov, Chen *et al*., 2013) entraîné sur le corpus. Les documents sont ensuite représentés en moyennant les représentations des mots qui les composent.

– Doc2Vec avec *PV-DBOW* (Le, Mikolov, 2014)

– TF-IDF

Nous calculons la matrice de similarité du corpus, au moyen de la similarité cosinus, puis pour chaque document nous regardons les documents les plus similaires. La figure 3 représente donc le nombre de liens correctement prédits pour tous les documents par rapport au nombre de plus proches voisins pris en compte.

On constate que la similarité entre les documents semble être corrélée à la présence de liens, les document ayant tendance à apparaître plus souvent dans la partie haute du classement des pages les plus similaires.

Nous devons également tester s'il est intéressant de s'appuyer sur le graphe des bases de connaissances pour aider à guider l'apprentissage des représentations. La méthodologie d'évaluation prendrait la forme d'une tâche de prédiction de liens à partir des représentations conjointes des données textuelles et du graphe. Néanmoins la nature des données considérées pour le moment ne nous donne pas la possibilité d'une telle évaluation. Les liens entre documents sont encore trop rares (de l'ordre d'une arête pour plus de 20 sommets), puisque directement liés à l'activité des utilisateurs.

C'est dans l'optique de palier au caractère très creux de ce corpus que nous travaillerons avec un corpus similaire issu de Wikipedia, où les liens entre les pages sont représentés par la section *Articles connexes*. Ces liens internes fonctionnent exactement comme les liens au sein des bases puisqu'ils sont eux aussi choisis manuellement par les contributeurs. L'augmentation du nombre de documents et de liens permettra ainsi de pouvoir tester la pertinence de la représentation du graphe, en attendant que les graphes de l'entreprise se densifient.

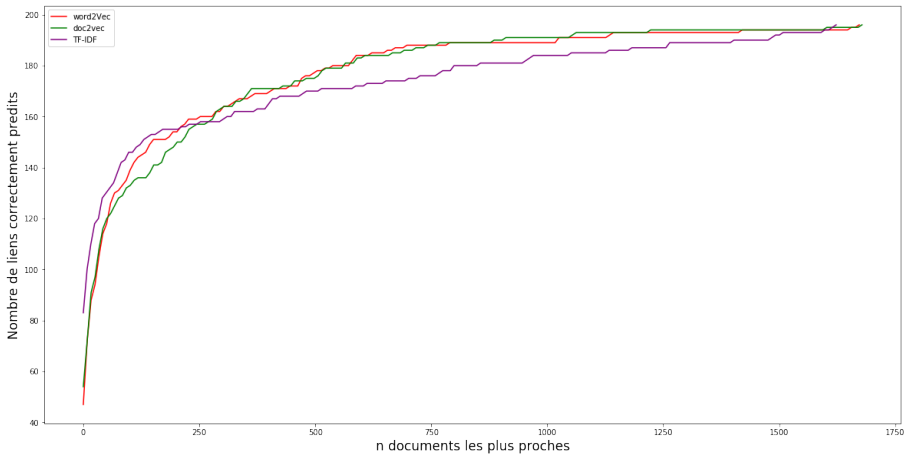

FIGURE *3. Représentation cumulative du nombre de liens correctement prédits en fonction des n documents les plus proches (word2vec en rouge, Paragraph2Vec en vert et TF-IDF en violet)*

5. Perspectives et conclusion

Nous avons présenté les idées de bases de ce travail de thèse, à savoir représenter des bases de connaissances en tenant compte de la granularité des documents afin d'affiner les recommendations, au sein d'une même base ou entre des bases différentes.

Les prochains objectifs seront dans un premier temps de tester et d'évaluer un modèle de plongement de document représentant dans le même espace les mots, les phrases, les paragraphes et les documents à la manière de ce qui a été proposé avec Paragraph2Vec. Il serait également intéressant d'adapter cette représentation dans une version plus proche du gaussian embedding mentionné plus tôt, afin de verifier si la forme des distributions capture également à l'échelle des document la "précision" du sujet abordé. A partir de ces représentations multi-échelle nous tenterons de mettre en place un système de recommandation faisant ressortir les similarités entre différentes portions de textes pour créer un réseau plus fin de liens entre les informations. Le second aspect à aborder sera la recommandation entre des bases de domaines et de vocabulaires différents, en cherchant à aligner les représentations de bases disjointes.

Bibliographie

Belkin M., Niyogi P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In T. G. Dietterich, S. Becker, Z. Ghahramani (Eds.), *Advances in neural information processing systems 14*, p. 585–591. MIT

Press. Consulté sur http://papers.nips.cc/paper/1961-laplacian-eigenmaps-and-spectral-techniques-for-embedding-and-clustering.pdf

Blei D. M., Ng A. Y., Jordan M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, vol. 3, n° Jan, p. 993–1022. (Citation Key: blei_latent_2003)

Bojanowski P., Grave E., Joulin A., Mikolov T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, vol. 5, p. 135–146.

Bojchevski A., Günnemann S. (2017). Deep gaussian embedding of attributed graphs: Unsupervised inductive learning via ranking. *CoRR*, vol. abs/1707.03815.

Dos Santos L., Piwowarski B., Gallinari P. (2016). Multilabel classification on heterogeneous graphs with gaussian embeddings. In P. Frasconi, N. Landwehr, G. Manco, J. Vreeken (Eds.), *Machine learning and knowledge discovery in databases*, p. 606–622. Springer International Publishing.

Harris Z. (1954). Distributional structure. *Word*, vol. 10, n° 23, p. 146–162.

He S., Liu K., Ji G., Zhao J. (2015). Learning to represent knowledge graphs with gaussian embedding. In *Proceedings of the 24th acm international on conference on information and knowledge management*, p. 623–632. ACM. Consulté sur http://doi.acm.org/10.1145/2806416.2806502

Kipf T. N., Welling M. (2017). Semi-supervised classification with graph convolutional networks. In *International conference on learning representations (iclr)*.

Le Q., Mikolov T. (2014, 22–24 Jun). Distributed representations of sentences and documents. In E. P. Xing, T. Jebara (Eds.), *Proceedings of the 31st international conference on machine learning*, vol. 32, p. 1188–1196. Bejing, China, PMLR.

Mikolov T., Chen K., Corrado G. S., Dean J. (2013). Efficient estimation of word representations in vector space. *CoRR*, vol. abs/1301.3781.

Mikolov T., Le Q. V., Sutskever I. (2013). Exploiting similarities among languages for machine translation. *CoRR*, vol. abs/1309.4168.

Mikolov T., Yih W.-t., Zweig G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, p. 746–751. Association for Computational Linguistics.

Pennington J., Socher R., Manning C. D. (2014). Glove: Global vectors for word representation. In *Empirical methods in natural language processing (emnlp)*, p. 1532–1543.

Perozzi B., Al-Rfou R., Skiena S. (2014). Deepwalk: Online learning of social representations. *arXiv:1403.6652 [cs]*, p. 701–710. (arXiv: 1403.6652)

Salton G., Buckley C. (1988, janvier). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, vol. 24, n° 5, p. 513–523.

Vilnis L., McCallum A. (2014, décembre). Word Representations via Gaussian Embedding. *arXiv:1412.6623 [cs]*. Consulté sur http://arxiv.org/abs/1412.6623 (arXiv: 1412.6623)