Modèle Transformer à base de Connaissances pour la Recherche d'Information dans des Domaines Spécialisés

Jibril Frej, Didier Schwab, Jean-Pierre Chevallet

Modèle Transformer à base de Connaissances pour la Recherche d'Information dans des Domaines Spécialisés

Jibril Frej, Didier Schwab, Jean-Pierre Chevallet

Univ. Grenoble Alpes, CNRS, Grenoble INP*, LIG, 38000 Grenoble, France
* Institute of Engineering Univ. Grenoble Alpes
{jibril.frej, didier.schwab, jean-pierre.chevallet}@univ-grenoble-alpes.fr,

Résumé. Le problème de la disparité des termes (*term mismatch*) occure fréquement en recherche d'information (RI). Il peut se produire lorsque la requête est courte et/ou ambiguë mais aussi dans des domaines spécialisés où les requêtes sont effectuées par des non-spécialistes et les documents sont rédigés par des experts. Récemment, le problème de disparité des termes a été abordé à l'aide de modèles neuronaux d'apprentissage de classement (*Neural Learning-To-Rank*) et de plongements de mots pour éviter d'utiliser uniquement la correspondance exacte des termes pour la recherche. Une autre approche au problème de la disparité des termes consiste à utiliser des bases de connaissances (*Knowledge Bases*) qui peuvent associer différents termes au même concept. Compte tenu du succès récent des encodeurs de type *transformers* en traitement automatique du language naturel (TALN), nous proposons KTRel : un modèle de type *Neural Learning-To-Rank* (NLTR) qui utilise des plongements de mots, des plongements de bases de connaissances et des encodeurs *transformers* pour la RI dans des domaines spécialisés. Dans cet article, nous évaluons KTRel sur une tâche de RI médicale.

1 Introduction

Dans des domaines spécialisés comme le domaine médical, les non-spécialistes expriment leurs requêtes en langage "courant", alors que les documents contiennent des termes spécifiques à un domaine. Par exemple, si un utilisateur demande *"Comment les régimes à base de plantes peuvent prolonger la vie ?"*, un système de recherche d'information (RI) basé sur un modèle *Bag-Of-Words* (BoW) ne pourra pas récupérer des documents pertinents tels que *"Un examen de la dépendance à la méthionine et du rôle de la restriction de méthionine dans la lutte contre le cancer et l'allongement de l'espérance de vie"*. Pour récupérer ce document, un système de RI doit pouvoir associer *"régimes à base de plantes"* avec *"restriction en méthionine"*.

D'une part, les modèles *Neural Learning-To-Rank* (NLTR) qui utilisent des plongements de mots appris sur de grandes quantités de texte brut forment une approche prometteuse à ce problème. Cependant, ces modèles n'ont toujours pas réalisé de percées significatives en RI (Guo et al., 2019b) et ont du mal à surpasser des méthodes de référence de type BoW sur des collections de RI standard telles que *Robust04* où la quantité de données annotées est limitée (Yang

et al., 2019).
D'autre part, il a souvent été proposé d'utiliser des bases de connaissances (KB) pour étendre les requêtes et/ou les documents avec des concepts afin de d'aborder la disparité des termes puisque le même concept est associé à des mots appartenant à des vocabulaires non-spécialistes et experts. Cependant, il s'agit d'une tâche difficile car la KB peut être incomplète, conduire à de l'ajout de bruit et nécessiter des adaptations au cas par cas (Zuccon et al., 2018). Dans ce travail, nous étudions la possibilité pour les modèles NLTR d'ignorer le bruit introduit par la KB et de se concentrer sur les connaissances pertinentes pour améliorer la RI dans des domaines spécialisés.
Compte tenu du succès récent des encodeurs de type *transformers* pour plusieurs tâches de TALN (Devlin et al., 2018; Vaswani et al., 2017; Radford et al., 2019; Lample et Conneau, 2019), nous proposons KTRel : un modèle NLTR qui utilise : **(1)** des plongements de mots pré-entrainés sur une grande quantité de texte spécialisé ; **(2)** des plongements de concepts pré-entrainés sur une KB spécialisé ; **(3)** des encodeurs *transformers* qui associent une séquence de plongements mots et de plongements de concepts à une représentation de taille fixe. KTRel est évalué dans le domaine médical en raison des KB et des collections de RI disponibles pour ce domaine (Goeuriot et al., 2016; Zuccon et al., 2018).

2 État de l'art

Plusieurs modèles ont déjà été proposées afin d'utiliser des KB pour la RI dans des domaines spécialisés. Ces modèles utilisent une (ou une combinaison) des trois stratégies suivantes : **(1)** des règles explicites ; **(2)** des méthodes d'apprentissage automatique basées sur des caractéristiques conçues manuellement ; **(3)** des méthodes d'apprentissage profond.
Règles explicites. Le modèle d'expansion de requêtes à l'aide d'entités (Dalton et al., 2014) qui utilise les connections entre éléments des KB pour étendre les requêtes avec des entités a été étudié en profondeur par Jimmy et al. (Zuccon et al., 2018) dans le domaine médical. Ils montrent que de telles méthodes nécessitent plusieurs choix clés et des décisions de conception pour être efficaces et sont donc difficiles à utiliser en pratique.
Apprentissage automatique. Soldaini et al. (Soldaini et Goharian, 2017) ont proposé d'utiliser des KB pour ajouter des caractéristiques médicales et de santé conçues manuellement afin d'améliorer la performance des méthodes d'apprentissage de classement pour la RI dans le domaine médical. Cependant, cette approche s'appuie sur des caractéristiques conçues manuellement qui nécessitent des connaissances spécifiques au domaine et à la base de connaissances lors de leur conception.
Apprentissage profond. Récemment, les KB ont été combinées avec succès avec les approches NLTR pour des systèmes de Question-Réponses (Shen et al., 2018) et pour la recherche sur le web dans le domaine général (Liu et al., 2018). Les modèles NLTR pour la RI fonctionnent de façon similaire aux modèles neuronaux pour le traitement automatique du langage naturel (TALN). La principale différence réside dans le fait que les fonctions objectifs utilisées par les modèles NLTR optimisent le classement d'une liste de documents par rapport à une requête. Des méthodes d'apprentissage non supervisé ont également été utilisées afin d'apprendre des réprésentations vectorielles de documents basées sur des concepts médicaux pour la recherche d'information dans le domaine médicale (Nguyen et al., 2017).
Dans ce travail, nous proposons un modèle NLTR supervisé de type *end-to-end* qui utilise les

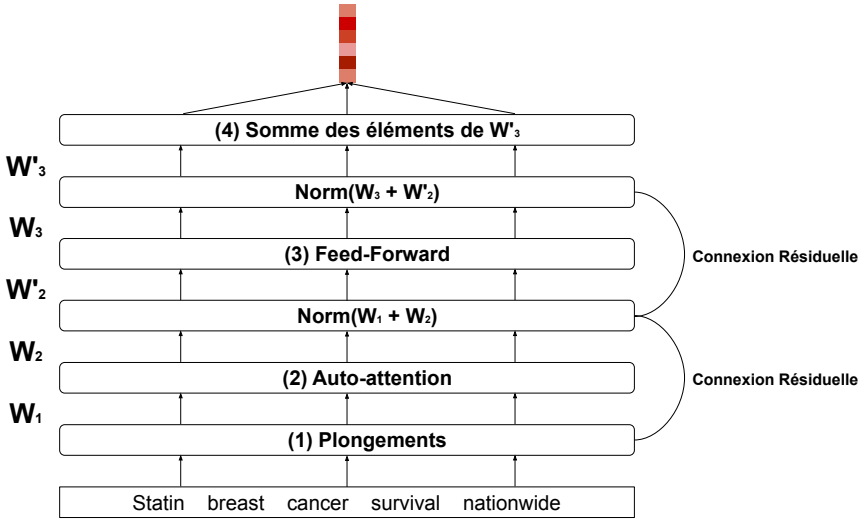

FIG. 1 – *Architecture d'un encodeur transformer*

connaissances d'une KB médical. Notre modèle n'est pas spécifique à un domaine dans sa conception et peut être adapté pour la RI dans d'autres domaines spécialisés.

3 KTRel

Dans cette section, nous décrivons les différentes étapes de notre méthode pour la RI dans un domaine spécialisé. L'architecture globale de KTRel est montrée sur la Figure 2.

Étape préliminaire. Afin d'inclure des connaissances *a priori* à notre modèle, nous pré-entraînons les plongements de mots sur du texte brut d'un domaine spécialisé et nous pré-entraînons les plongements de concepts sur une KB du même domaine spécialisé.

Étape conceptuelle. Les requêtes et les documents sont annotés avec un ensemble de concepts candidats originant de la KB du domaine spécialisé. Nous adoptons la même stratégie que Shen et al. (Shen et al., 2018) : les n-grammes sont annotés avec leurs K-meilleurs concepts candidats afin de traiter l'ambiguïté possible de certains n-grammes. Par exemple, la requête *"statin breast cancer survival nationwide"* est associée à la séquence de concepts suivante : [C0360714 , C0075191 , C0638232 , C0006142 , C0678222 , C0006142 , C1332438 , C0006142 , C0678222 , C0038952 , C0220921 , C0038952].

Étape Transformer. Considérant les gains de performance significatifs récemment obtenus par les *transformers* en TALN (Devlin et al., 2018; Vaswani et al., 2017; Radford et al., 2019; Lample et Conneau, 2019), nous proposons d'utiliser des encodeurs *transformers* pour associer aux séquences de mots et aux séquences de concepts une représentation de taille fixe. Plus précisement, l'encodeur *transformer* suit les étapes suivantes : **(1)** association des éléments de la séquence en entrée avec leurs plongements correspondantes ; **(2)** utilisation du mécanisme d'auto-attention (Vaswani et al., 2017) pour calculer une représentation des éléments de la séquence en fonction d'une combinaison linéaire des plongements de leur contexte ; **(3)** ap-

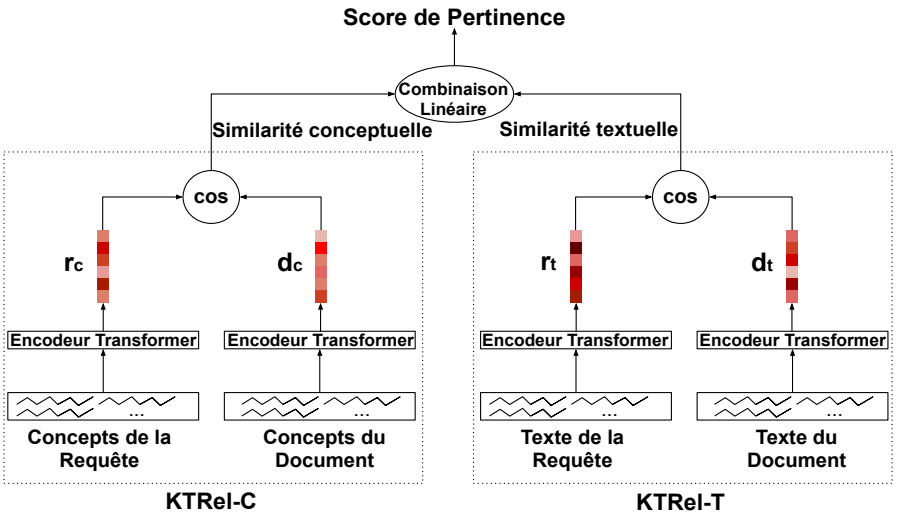

FIG. 2 – *Architecture de KTRel*

plication d'un réseau de neurones de type *Feed Forward*[1] sur chacun des éléments obtenus précédemment ; **(4)** somme des éléments de la séquence de représentations obtenues préalablement pour obtenir une représentation de taille fixe. Notons également qu'après les étapes (2) et (3) l'encodeur *transformer* effectue une somme avec les éléments de la couche précédente suivie d'une normalisation (voir Figure 1).

Étape de pertinence. Une similarité conceptuelle est calculée en utilisant le cosinus entre l'encodage *transformer* des concepts de la requête r_c et l'encodage *transformer* des concepts du document d_c. De façon similaire, nous calculons une similarité textuelle (voir Figure 2). Le score de pertinence entre la requête R et le document D est calculée en faisant une combinaison linéaire entre la similarité conceptuelle et la similarité textuelle :

$$\text{Rel}(R, D) = a \cos(r_t, d_t) + b \cos(r_c, d_c) \quad (1)$$

Avec $a \in \mathbb{R}$ et $b \in \mathbb{R}$ deux paramètres appris pendant l'entraînement. a et b sont bien des paramètres du modèle et non pas des hyperparamètres. Par conséquent ils sont appris de la même façon que les autres paramètres du modèle : en utilisant une méthode basée sur la descente de gradient (voir Section 4.2).

4 Expériences

4.1 Jeux de données

Collection. Nous évaluons KTRel sur la collection NFCorpus (Boteva et al., 2016) : une collection en langue anglaise, accessible au public. NFCorpus a été poposée pour développer

1. Ce réseau est composé d'une couche linéaire, d'un fonction d'activation et d'une autre couche linéaire

Type de requête	Exemple	Document pertinent associé
titles	ultra-processed foods	manufactured uncertainty ...
non-topic titles	coffee and artery function	caffeine enhances endothelial ...
video titles	the actual benefit of diet vs. drugs	...placebo study diet drug ...
video descriptions	a dairy-free diet may stop ...	significant dietary intake ...

TAB. 1 – *Exemples de requêtes et d'un document pertinent associé provenant du NFCorpus. Le corpus étant déjà pré-traité, les exemples ci-dessus ne contiennent ni majuscules, ni ponctuation.*

des méthodes de type *learning-to-rank* dans le domaine médical. Elle se compose de 5276 requêtes rédigées en langage courant et de 3633 documents composés de titres et de résumés de PubMed et de PMC dans un vocabulaire hautement technique. Les requêtes sont séparées en 4 catégories : *titles*, *non-topics titles*, *video titles* et *video descriptions*. Des examples de requêtes et de document pertinents associés sont illustrés sur la Table 1 Nous n'avons pas évalué notre modèle sur les collections de RI médicales standard telles que CLEF eHealth 2013 (Suominen et al., 2013) ou CLEF eHealth 2014 (Goeuriot et al., 2014) car elles ne contiennent que 55 requêtes annotées chacunes, ce qui n'est pas suffisant pour entraîner des modèles NLTR performants (Guo et al., 2019b; Yang et al., 2019).

Base de connaissances. Nous utilisons des concepts médicaux de la version *2018AA* du Metathesaurus UMLS (Bodenreider, 2004). Nous avons choisi UMLS principalement en raison de sa couverture : 3,67 millions de concepts tirés de 203 vocabulaires sources.

4.2 Configuration expérimentale

Concepts. Nous utilisons *MetamorphoSys* pour extraire le graphe relationnel des concepts médicaux d'UMLS. Nous éliminons les concepts qui n'appartiennent pas à un type sémantique médicale (par exemple, les concepts quantitatifs). Le texte est annoté avec des concepts médicaux en utilisant *QuickUMLS* (Soldaini et Goharian, 2016). Les paramètres de *QuickUMLS* sont réglés sur leur valeurs par défaut. Comme l'ont fait Shen et al. (Shen et al., 2018), nous fixons le nombre de concepts candidats K à 8.

Plongements de mots. Nous utilisons des plongements de mots entraînés avec *word2vec* (Mikolov et al., 2013) sur une combinaison de textes de PubMed et PMC disponibles sur : `http://bio.nlplab.org`. Les plongements de concepts sont entraînés sur le graphe relationnel UMLS avec TransE (Bordes et al., 2013). Toutes les plongements sont mises à jour pendant l'entraînement de nos modèles pour la RI et les dimensions des plongements de mots et de concepts sont fixés à 200.

Implémentation. KTRel est implémenté avec *pytorch* (`https://pytorch.org/`).

Fonction de perte. Les modèles sont entraînés en minimisant la fonction de perte de *Hinge* pour le classement :

$$L = \max(0, 1 - \text{Rel}(R, D^+) + \text{Rel}(R, D^-)) \tag{2}$$

Avec D^+ un document plus pertinent que D^- par rapport à la requête R .

Encodeur transformer. Le nombre de têtes d'attention et la dimension du réseau *feed forward* sont sélectionnés parmi $\{1, 2, 5, 10\}$ et $\{50, 100, 200, 500\}$ respectivement. Nous avons utilisé

la fonction d'activation ReLU. Des expériences préliminaires ont montré que l'utilisation d'une seule couche d'encodeur *transformer* donne les meilleurs résultats. Cela est probablement dû à la petite taille de notre collection.

Entraînement. L'optimisateur Adam (Kingma et Ba, 2015) est utilisé avec les valeurs par défaut des paramètres. La taille des *batchs* et le taux de *dropout* sont sélectionnés parmi $\{10, 20, 50\}$ et $\{0.1, 0.0.2, 0, 0.3, 0, 0.4, 0.5\}$ respectivement. Pour lutter contre le surapprentissage, en plus du *dropout*, nous utilisons la strategie *early stopping* sur la MAP de l'ensemble de validation.

Validation. Les hyper-paramètres listés ci-dessus sont réglés sur la MAP de l'ensemble de validation en utilisant une recherche par grille (*grid search*).

Évaluation. Nous utilisons 4 mesures standard d'évaluation en RI : la MAP, le Rappel, la Précision et la nDCG sur les 1000 meilleurs document. La nDCG@k (*normalized Discounted Cumulative Gain*) est une mesure similaire à la Précision@k qui : **(1)** prend en compte plus de niveaux de pertinence que simplement pertinent (1) et non-pertinent (0) ; **(2)** applique un coefficient de réduction pour donner plus d'importance aux documents dans le haut du classement ; **(3)** est normalisé par le score du classement idéal afin d'avoir une mesure égale à 1 si le classement retourné est idéal (Järvelin et Kekäläinen, 2002). Ces mesures sont calculées avec *pytrec_eval* (Van Gysel et de Rijke, 2018). Nous utilisons un test t (*two-tailed paired t-test*) avec correction de Bonferroni pour mesurer les différences statistiquement significatives entre les mesures d'évaluation. Comme le NFCorpus ne possède que 3633 documents, nous pouvons évaluer chaque paire (requête, document) dans un délai raisonnable afin d'éviter de dépendre d'une stratégie de reclassement (Guo et al., 2019b; Zamani et al., 2018). Par conséquent, le rappel de KTRel et des autres modèles NLTR de référence n'est pas restreint par une étape de classement antérieure.

4.3 Modèles de référence

Nous comparons KTRel à trois types de méthodes de référence : BoW, NLTR pour la RI et encodeur BERT pré-entraîné.

BoW. Comme suggéré par Yang et al. (Yang et al., 2019), nous utilisons Okapi BM25 (Robertson et Walker, 1994) et Okapi BM25 avec la stratégie de *pseudo-relevance feedback* RM3 (Lv et Zhai, 2009) comme références de type BoWs. La racinisation, l'indexation et l'évaluation de BM25 et BM25-RM3 sont effectués par Terrier (Ounis et al., 2005). Les valeurs des hyper-paramètres sont réglées sur la MAP de validation avec une recherche par grille.

NLTR. DUET (Mitra et al., 2017), KNRM (Xiong et al., 2017), DRMM (Guo et al., 2016) et Conv-KNRM (Dai et al., 2018) sont utilisés comme modèles de référence NLTR pour IR. L'entraînement et l'évaluation de ces modèles sont effectués avec *MatchZoo* (Guo et al., 2019a). Les valeurs des hyper-paramètres sont réglées sur la MAP de l'ensemble validation avec une recherche aléatoire sur 10 exécutions. Nous utilisons le *tuner* fourni par *MatchZoo* pour échantillonner des valeurs de l'espace des hyper-paramètres associé à chaque modèle.

Encodeur BERT pré-entraîné. Nous comparons également KTRel à l'encodeur BERT (Devlin et al., 2018) : un modèle de représentation linguistique état de l'art. Nous utilisons le modèle *"bert-base-uncased"* fourni par *Hugging Face* (Wolf et al., 2019), pré-entraîné sur BooksCorpus (Zhu et al., 2015) et English Wikipedia. Lors de l'entraînement, nous ne mettons à jour que la dernière couche du modèle. Nous verrons dans les perspectives l'utilisation de BERT pré-entraîné sur des corpus médicaux (Lee et al., 2019) au lieu d'utiliser la version pré-entraînée

modèles	P@5	P@10	nDCG@5	nDCG@10	MAP	Rappel
BM25	0.2846–	0.2419–	0.3524	0.3267–	0.1548–	0.4740–
BM25-RM3	0.3056–	0.2603–	0.3664	0.3431	0.1801–	0.6249–
DUET	0.1967–	0.1840–	0.1857–	0.1883–	0.1264–	0.7673–
KNRM	0.2082–	0.1887–	0.1914–	0.1914–	0.1216–	0.7764–
DRMM	0.2940–	0.2489–	0.3540	0.3330–	0.1651–	0.7051–
Conv-KNRM	0.3146–	0.2865–	0.3010–	0.3090–	0.2110–	0.8143–
BERT	0.2084–	0.1998–	0.2090–	0.2196–	0.1567–	0.7847–
KTRel-C	0.3127–	0.2889–	0.3285–	0.3295–	0.2194–	0.8047–
KTRel-T	0.3204–	0.3008–	0.3465–	0.3369–	0.2228–	0.8187–
KTRel	**0.3554**	**0.3294**	**0.3708**	**0.3584**	**0.2411**	**0.8520**

TAB. 2 – *Comparaison des performances des différents modèles sur le NFCorpus. (-) indique une dégradation statistiquement significative de la performance par rapport a KTRel (p-value < 0.05)*

sur des données textuelles du domaine général.

Architecture de KTRel. Pour étudier l'utilité de la combination de concepts et de mots, nous entraînons et évaluons séparément la partie de l'architecture KTRel qui utilise uniquement des concepts (dénotée KTRel-C) et la partie qui utilise uniquement le texte (dénommée KTRel-T) comme illustré dans Figure 2. KTRel-C et KTRel-T sont entraînés indépendamment en utilisant la même configuration que celle décrite dans la sous-section 4.2.

5 Résultats

Les performances de KTRel par rapport aux modèles de référence sont indiquées dans le tableau 2. Dans ce qui suit, nous proposons des réponses empiriques à plusieurs questions de recherche.

Est-il utile pour la RI d'utiliser à la fois des mots et des concepts médicaux ? KTRel surpasse tous les modèles de référence NLTR sur toutes les mesures de façon statistiquement significative. Le fait que KTRel surpasse également KTRel-W et KTRel-T prouve empiriquement que la RI dans un domaine spécialisé peut bénéficier de la combinaison de représentations conceptuelles pré-entraînées avec des représentations de mots pré-entraînées.

KTRel peut-il surpasser une base de référence solide de type BoW ? KTRel surpasse BM25 avec extension de requête RM3 sur toutes les mesures et le fait de façon statistiquement significative sur la majorité d'entre elles. Le classement général est largement amélioré par KTRel : +33.9% de MAP. Les exceptions notables sont la nDCG@5 et la nDCG@10 : même si KTRel surpasse BM25-RM3 en termes de nDCG@5 (+1.2%) et de nDCG@10 (+4.4%), les gains ne sont pas statistiquement significatifs. Il est intéressant de noter que le KTRel est statistiquement significatif par rapport à la P@5 (+16.3%) et la P@10 (+26.5%). La différence entre la P@k et la nDCG@k est que la précision ne prend en compte que la proportion de documents pertinents alors que la nDCG@k met davantage l'accent sur le classement lui-même et prend en compte les niveaux de pertinence des documents. Par conséquent, nous pouvons conclure que même si KTRel est capable de récupérer plus de documents pertinents dans les résultats

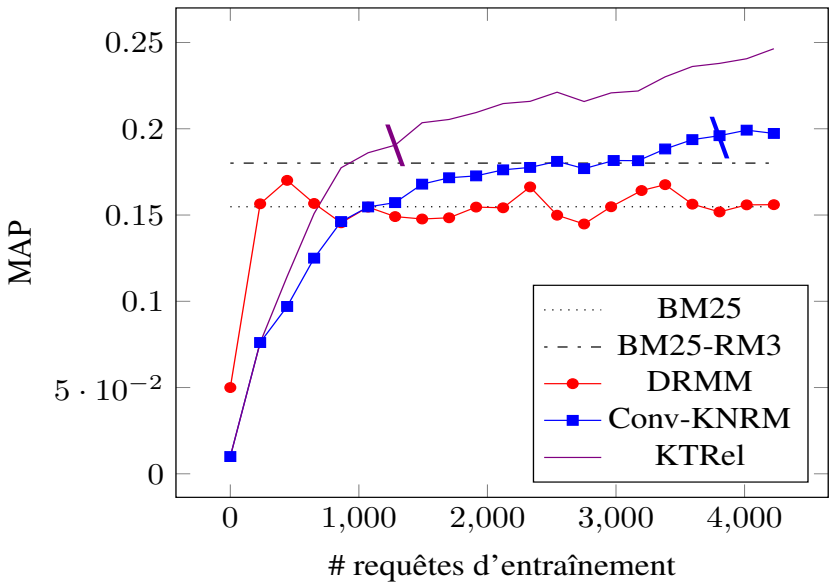

FIG. 3 – *MAP en fonction du nombre de requêtes utilisées pendant l'entraînement.* \ *indique quand un modèle NLTR a suffisamment de requêtes d'entraînement pour obtenir une amélioration statistiquement significative par rapport à BM25-RM3 (p-value < 0.05)*

du top-k, BM25-RM3 est toujours une référence solide quand il s'agit du classement des documents du top-k.

L'encodeur BERT fournit-il des représentations utiles pour la RI dans le domaine médical ? BERT obtient de moins bons résultats que BM25 malgré son succès dans plusieurs tâches de TALN (Devlin et al., 2018). Cela est probablement dû au fait qu'il est pré-entraîné sur des ensembles de données non médicales et que le NFCorpus est trop petit pour affiner correctement l'encodeur BERT pour la RI dans le domaine médical.

Les modèles de référence NLTR peuvent-ils surpasser une base de référence solide de type BoW ? Tout d'abord, nous remarquons que les modèles DUET et KNRM sont moins performants que BM25. Cela est probablement dû au fait que ces modèles ont été développés sur des ensembles de données beaucoup plus importants (Mitra et al., 2017; Xiong et al., 2017) que le NFCorpus. Deuxièmement, DRMM obtient de meilleurs résultats que BM25 (+6,7% de MAP, +3,3% de P@5 et +0,5% de nDCG@5), mais il ne parvient pas à surpasser BM25-RM3 ; enfin, Conv-KNRM est le seul modèle de référence NLTR qui parvient à surpasser BM25 et BM25-RM3 en terme de MAP, Précision et rappel (mais pas en terme de nDCG). Ces résultats confirment empiriquement que les modèles NLTR n'ont pas réalisé de percées significatives dans la RI (Guo et al., 2019b).

Les encodeurs *transformers* fournissent des représentations utiles pour la RI dans des domaines spécialisés ? KTRel-W et KTRel-C ont des résultats similaires à la meilleure méthode de référence NLTR. De plus, le fait que ces modèles reposent sur une simple similarité cosinus entre la la représentation de la requête et la représentation du document démontre empirique-

ment que les encodeurs *transformers* produisent des représentations utiles pour la RI.
Comment les modèles NLTR affectent le rappel ? Comme le nombre de documents est limité dans le NFCorpus, nous ne nous appuyons pas sur une stratégie de reclassement basée sur BM25 (Zamani et al., 2018). Par conséquent, le rappel des modèles NLTR n'est pas limité par le rappel de BM25. Les résultats indiquent que le gain de KTRel en termes de rappel est significatif par rapport à BM25 (+78.6%) et BM25-RM3 (+36.3%). Cela se produit parce que les modèles BoW ne peuvent récupérer que les documents qui contiennent des termes de la requête alors que les modèles NLTR n'ont pas cette restriction.
Combien de données sont nécessaires pour surpasser BM25-RM3 avec une méthode de type NLTR ? Comme nous pouvons le voir sur la Figure 3, le nombre de requêtes nécessaires pour qu'un modèle NLTR surpasse un modèles de référence tel que BM25 ou BM25-RM3 varie en fonction du modèle que l'on considère. Sur NFCorpus, envions 200 requêtes sont requises par DRMM pour obtenir des résultats comparables à ceux de BM25. Cela est dû au fait que DRMM est un modèle ayant très peu de paramètres ($\approx$ 450) et qu'il n'a donc pas besoin de beaucoup de données pour converger. Cependant, et pour les même raisons, DRMM ne bénéficie pas de l'apport de beaucoup de données d'entraînement et ne parvient même pas à surpasser le modèle de référence BM25-RM3. Le modèle Conv-KNRM quant à lui parvient à suprpasser BM25 et BM25-RM3, mais environ 3 000 requêtes sont nécessaires dans l'ensemble d'entraînement pour que Conv-KNRM surpasse BM25-RM3 sur NFCorpus. Il semble que moins de données ($\approx$ 1 500 requêtes) sont nécessaires pour le modèle KTRel. Cela suggère que l'utilisation de concepts peut être utile dans des scénarios de ressources limitées. Ces résultats confirment également que l'entraînement d'un modèle NLTR sur une collection qui ne contient que quelques centaines de requêtes est une tâche très difficile. Cela pourrait expliquer pourquoi des percées significatives n'ont toujours pas été réalisées par NLTR sur des collections de RI standard qui ne contennent que quelques centaines de requêtes au mieux et quelques dizaines au pire.

6 Conclusions et travaux futurs

Dans cet article, nous proposons KTRel : un modèle NLTR basé sur l'encodeur *transformers* qui utilise à la fois des mots et des concepts pour la RI dans des domaines spécialisés. Nous démontrons empiriquement que l'ajout de concepts à un modèle neuronal d'apprentissage de classement est utile pour la RI dans le domaine médical. Nous montrons que les encodeurs de *transformers* fournissent des représentations de séquences utiles pour la RI. Nous confirmons aussi empiriquement que BM25 avec l'expansion des requêtes RM3 est toujours une base de référence solide, en particulier en ce qui concerne les métriques de haute précision. Comme travail futur, nous prévoyons d'évaluer KTRel sur d'autres collections et d'autres domaines spécialisés. Afin de rendre notre modèle extensible à de plus grandes collections, nous l'adapterons pour apprendre des représentations creuses (*sparse*) basées sur des mots et des concepts compatibles avec un index inversé comme suggéré par Zamani et al. (Zamani et al., 2018).

Références

Bodenreider, O. (2004). The unified medical language system (umls) : integrating biomedical terminology. Volume 32, pp. D267–D270. Oxford University Press.

Bordes, A., N. Usunier, A. Garcia-Duran, J. Weston, et O. Yakhnenko (2013). Translating embeddings for modeling multi-relational data. In *NIPS*, pp. 2787–2795.

Boteva, V., D. Gholipour, A. Sokolov, et S. Riezler (2016). A full-text learning to rank dataset for medical information retrieval. In *ECIR*, pp. 716–722.

Dai, Z., C. Xiong, J. Callan, et Z. Liu (2018). Convolutional neural networks for soft-matching n-grams in ad-hoc search. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, New York, NY, USA, pp. 126–134. ACM.

Dalton, J., L. Dietz, et J. Allan (2014). Entity query feature expansion using knowledge base links. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, New York, NY, USA, pp. 365–374. ACM.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2018). Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*.

Goeuriot, L., G. J. F. Jones, L. Kelly, H. Müller, et J. Zobel (2016). Medical information retrieval : introduction to the special issue. *Information Retrieval Journal 19*(1), 1–5.

Goeuriot, L., L. Kelly, W. Li, J. Palotti, P. Pecina, G. Zuccon, A. Hanbury, G. J. F. Jones, et H. Müller (2014). ShARe/CLEF eHealth Evaluation Lab 2014, Task 3 : User-centred health information retrieval. In *Proceedings of CLEF 2014*, Sheffield, United Kingdom.

Guo, J., Y. Fan, Q. Ai, et W. B. Croft (2016). A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 55–64. ACM.

Guo, J., Y. Fan, X. Ji, et X. Cheng (2019a). Matchzoo : A learning, practicing, and developing system for neural text matching. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, New York, NY, USA, pp. 1297–1300. ACM.

Guo, J., Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, et X. Cheng (2019b). A deep look into neural ranking models for information retrieval. *arXiv preprint arXiv :1903.06902*.

Järvelin, K. et J. Kekäläinen (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS) 20*(4), 422–446.

Kingma, D. P. et J. Ba (2015). Adam : A method for stochastic optimization. In *ICLR*.

Lample, G. et A. Conneau (2019). Cross-lingual Language Model Pretraining. *arXiv e-prints*, arXiv :1901.07291.

Lee, J., W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, et J. Kang (2019). BioBERT : a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Liu, Z., C. Xiong, M. Sun, et Z. Liu (2018). Entity-duet neural ranking : Understanding the role of knowledge graph semantics in neural information retrieval. *arXiv preprint arXiv :1805.07591*.

Lv, Y. et C. Zhai (2009). A comparative study of methods for estimating query language models with pseudo feedback. In *CIKM*, pp. 1895–1898.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, et J. Dean (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119.

Mitra, B., F. Diaz, et N. Craswell (2017). Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 1291–1299. International World Wide Web Conferences Steering Committee.

Nguyen, G., L. Tamine, L. Soulier, et N. Souf (2017). Learning concept-driven document embeddings for medical information search. In *Artificial Intelligence in Medicine - 16th Conference on Artificial Intelligence in Medicine, AIME 2017, Vienna, Austria, June 21-24, 2017, Proceedings*, pp. 160–170.

Ounis, I., G. Amati, V. Plachouras, B. He, C. Macdonald, et D. Johnson (2005). Terrier information retrieval platform. In *European Conference on Information Retrieval*, pp. 517–519. Springer.

Radford, A., J. Wu, R. Child, D. Luan, D. Amodei, et I. Sutskever (2019). Language models are unsupervised multitask learners. *OpenAI Blog 1*(8).

Robertson, S. et S. Walker (1994). Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR*, pp. 232–241.

Shen, Y., Y. Deng, M. Yang, Y. Li, N. Du, W. Fan, et K. Lei (2018). Knowledge-aware attentive neural network for ranking question answer pairs. In *SIGIR*, pp. 901–904. Springer.

Soldaini, L. et N. Goharian (2016). Quickumls : a fast, unsupervised approach for medical concept extraction. In *MedIR workshop, SIGIR*.

Soldaini, L. et N. Goharian (2017). Learning to rank for consumer health search : a semantic approach. In *ECIR*, pp. 640–646. Springer.

Suominen, H., S. Salanterä, S. Velupillai, W. W. Chapman, G. Savova, N. Elhadad, S. Pradhan, B. R. South, D. L. Mowery, G. J. Jones, et al. (2013). Overview of the share/clef ehealth evaluation lab 2013. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pp. 212–231. Springer.

Van Gysel, C. et M. de Rijke (2018). Pytrec_eval : An extremely fast python interface to trec_eval. In *SIGIR*. ACM.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al. (2017). Attention is all you need. In *NIPS*, pp. 5998–6008.

Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et J. Brew (2019). Transformers : State-of-the-art natural language processing.

Xiong, C., Z. Dai, J. Callan, Z. Liu, et R. Power (2017). End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*, pp. 55–64. ACM.

Yang, W., K. Lu, P. Yang, et J. Lin (2019). Critically examining the" neural hype" : Weak baselines and the additivity of effectiveness gains from neural ranking models. *arXiv preprint*

arXiv :1904.09171.

Zamani, H., M. Dehghani, W. B. Croft, E. Learned-Miller, et J. Kamps (2018). From neural re-ranking to neural ranking : Learning a sparse representation for inverted indexing. In *CIKM*, pp. 497–506. ACM.

Zhu, Y., R. Kiros, R. S. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, et S. Fidler (2015). Aligning books and movies : Towards story-like visual explanations by watching movies and reading books. *CoRR abs/1506.06724*.

Zuccon, G., B. Koopman, et al. (2018). Payoffs and pitfalls in using knowledge-bases for consumer health search. *Information Retrieval Journal*, 1–45.

Summary

Vocabulary mismatch is a frequent problem in information retrieval (IR). It can occur when the query is short and/or ambiguous but also in specialized domains where queries are made by non-specialists and documents are written by experts. Recently, vocabulary mismatch has been addressed with neural learning-to-rank (NLTR) models and word embeddings to avoid relying only on the exact matching of terms for retrieval. Another approach to vocabulary mismatch is to use knowledge bases (KB) that can associate different terms to the same concept. Given the recent success of transformer encoders for NLP, we propose KTRel: a NLTR model that uses word embeddings, Knowledge bases and Transformer encoders for IR in specialized domains. In this work, we evaluate KTRel for medical IR.