Knowledge Based Transformer Model for Information Retrieval

Jibril Frej
jibril.frej@univ-grenoble-alpes.fr
Univ. Grenoble Alpes, CNRS, Grenoble INP*, LIG
* Institute of Engineering Univ. Grenoble Alpes

Didier Schwab
didier.schwab@univ-grenoble-alpes.fr
Univ. Grenoble Alpes, CNRS, Grenoble INP*, LIG
* Institute of Engineering Univ. Grenoble Alpes

Jean-Pierre Chevallet
jean-pierre.chevallet@univ-grenoble-alpes.fr
Univ. Grenoble Alpes, CNRS, Grenoble INP*, LIG
* Institute of Engineering Univ. Grenoble Alpes

ABSTRACT

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.

KEYWORDS

Information Retrieval, Neural Networks, Learning-to-Rank, Knowledge Base

1 INTRODUCTION

In specialized domains like the medical one, non-specialists express their queries using plain English, whereas documents contain domain-specific terms. For example, if a user asks *"How plant-based diets may extend our lives?"* a bag-of-words (BoW) based IR system will be unable to retrieve relevant documents such as *"A review of methionine dependency and the role of methionine restriction in cancer growth control and life-span extension"*. To retrieve this document, an IR system should associate *"plant-based diets"* with *"methionine restriction"*.

On the one hand, neural learning-to-rank (NLTR) models that use prior knowledge from word embeddings trained on large amounts of raw text are a promising approach to this problem. However, most NLTR models are not interpretable, with unknown or rare words and struggle to outperform a well-tuned BoW baseline on standard IR collections such as *Robust04* where the amount of annotated data is limited [30].

On the other hand, using knowledge bases (KB) to expand queries and/or documents with concepts has often been proposed to tackle the vocabulary mismatch since the same concept/entity can be related to words belonging to non-specialist and expert vocabularies. However, it is a challenging task since KB can be incomplete, lead to noise addition and require hand-crafted features [33]. In this work, we study the potential for NLTR models to ignore the noise introduced by KB and focus on the relevant knowledge to improve search.

Given the recent success of transformer encoders for several NLP tasks [6, 13, 21, 27], we propose KTRel: a NLTR model that uses: **(1)** word embeddings pre-trained on large amount of text; **(2)** concept embeddings pre-trained on a specialized KB; **(3)** transformer encoders that associate sequence of word embeddings and concept embeddings to a fixed-size representation.

2 RELATED WORK

Several methods to include KB for IR in specialized domains have already been proposed. These models use one (or a combination) of the following three strategies: **(1)** explicit rules; **(2)** machine learning methods based on hand-crafted features; **(3)** deep learning methods.

Explicit rules. The entity query feature expansion [5] that uses relations between KB elements to extend queries with entities has been studied in depth by Jimmy et. al. [33] in the medical field. They show that such methods require several key choices and design decisions to be effective and are therefore difficult to use in practice.

Machine learning. Soldaini et al. [25] proposed to use KBs to add medical and health hand-crafted features to improve the performance of learning-to-rank methods for IR in the medical field. However, this approach relies on hand-crafted features that require domain and KB specific knowledge when they are designed.

Deep learning. Recently, KBs have been successfully combined with NLTR approaches for Question-Answer systems [23] and for web search in the general domain [15]. NLTR models for IR work similarly to neural models for automatic natural language processing (TALN). The main difference is that the objective functions used by the NLTR models optimize the ranking of a list of documents with respect to a query. Unsupervised learning methods were also used to learn vector representations of documents based on medical concepts for information retrieval in the medical domain [19].

3 KTREL

In this section, we describe the different steps our model follows to perform IR. The overall architecture of KTRel is shown in Figure 1.

Prior step. In order to include prior knowledge to our model, we pre-train word embeddings on raw text and we pre-train concept embeddings on a specialized domain KB.

Knowledge step. Queries and documents are annotated with a set

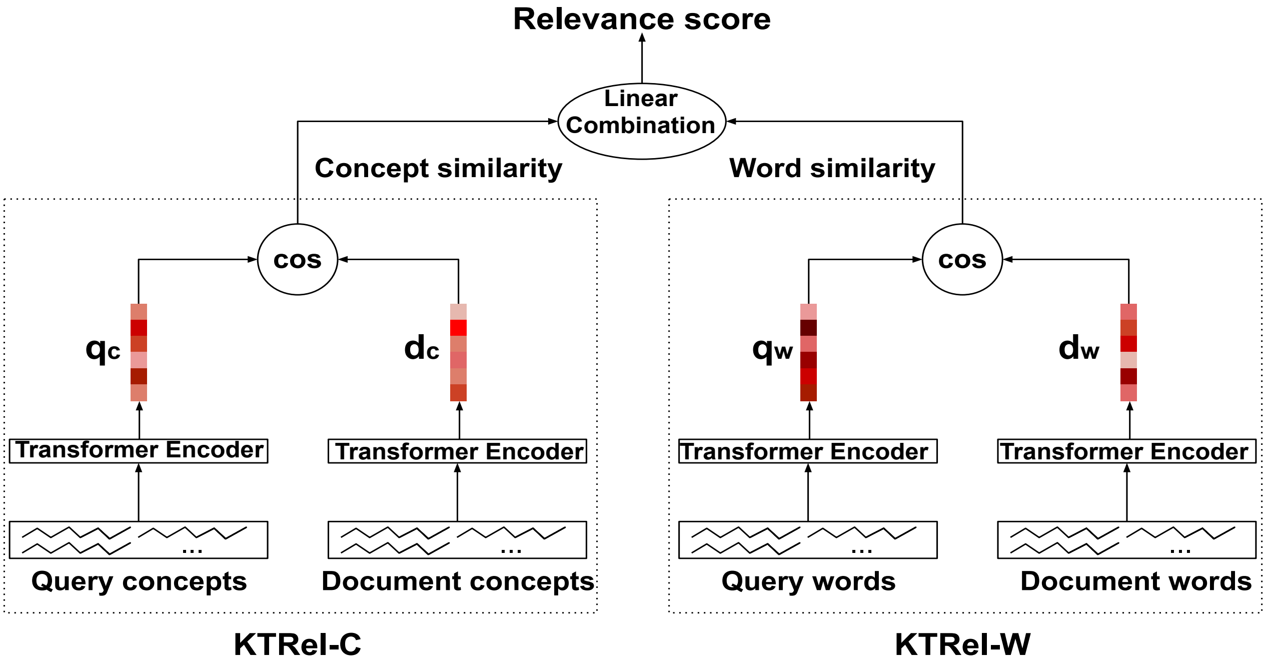

Figure 1: Architecture of KTRel

of candidate concepts from the specialized domain KB. We adopt the same strategy as Shen et al. [23]: n-grams are annotated with their top-K candidate concepts in order to deal with the possible ambiguity of some n-grams.

Transformer step. Considering the significant performance gains recently obtained by transformers in NLP [6, 13, 21, 27], we propose to use transformers encoders to associate both sequences of words and sequences of concepts with a fixed size representation using the following steps: **(1)** a mapping of the elements of the input sequence with their corresponding embeddings; **(2)** a self-attention mechanism [27] to compute a context aware representations of elements of the sequence; **(3)** a position-wise Feed Forward Network; **(4)** an element-wise sum of the representations obtained previously to get a fixed size sequence encoding.

Relevance step. A concept-based similarity is computed using the cosine between the transformer encoding of the query's concepts q_c and the transformer encoding of the document's concepts d_c. Analogously, we calculate a word-based similarity (see Figure 1). The final relevance score between query Q and document D consists in a linear combination between the concept-based similarity and the word-based similarity:

$$\mathrm{Rel}(Q, D) = a \cos\left(q_w, d_w\right) + b \cos\left(q_c, d_c\right) \qquad (1)$$

With $a \in \mathbb{R}$ and $b \in \mathbb{R}$ two parameters learned during training.

4 EXPERIMENTS

In this section, we describe the empirical evaluation of our NLTR models. We first present the data (Section 4.1), our baselines (Section 4.3) and the experimental setup (Section 4.2).

4.1 Datasets

Collection. We evaluate KTRel on the NFCorpus [3]: a publicly available collection for learning-to-rank in the medical domain. It consists of 5,276 different queries written in plain English and 3,633 documents composed of titles and abstracts from PubMed and PMC with a highly technical vocabulary. We did not evaluate our model on standard medical *ad hoc* IR collection such as CLEF eHealth 2013 [26] or CLEF eHealth 2014 [7] because they contain about 50 annotated queries each which is not enough to train NLTR models [9, 30].

Knowledge base. We use medical concepts from the version *2018AA* of the UMLS Metathesaurus [1]. We choose the UMLS Metathesaurus mainly because of its huge coverage: 3.67 million concepts from 203 source vocabularies.

4.2 Experimental setup

Concepts. We use *MetamorphoSys* to extract the relational graph of medical concepts from UMLS. We discard concepts that do not belong to a medical semantic type (e.g. Quantitative Concept). Text is annotated with medical concepts using *QuickUMLS* [24] with default parameter values. As done by Shen et al. [23], the number of candidate concepts K is set to 8.

Pre-trained Embeddings. We use word embeddings trained with word2vec [17] on a combination of PubMed and PMC texts available at: http://bio.nlplab.org. Concept embeddings are trained on the UMLS relational graph with TransE [2]. All embeddings are updated during training and both word and concept embedding dimensions are set to 200.

Implementation. KTRel is implemented in *pytorch* (https://pytorch.

org/).

Loss. Models are trained to minimize the Margin Ranking Loss:

$$L = \max(0, 1 - \mathrm{rel}(Q, D^{+}) + \mathrm{rel}(Q, D^{-})) \qquad (2)$$

Where D^{+} is a document more relevant to query Q than D^{-}.

Transformer encoder. The number of attention heads and the dimension of the feed forward network are selected from $\{1, 2, 5, 10\}$ and $\{50, 100, 200, 500\}$ respectively. We used ReLU activation function. Preliminary experiments showed that using a single transformer encoder layer yields the best results. This is probably due to the small size of our collection.

Training. Adam optimizer [12] is used with default parameter values. Batch size and dropout rate are selected from $\{10, 20, 50\}$ and $\{0.1, 0.2, 0.3, 0.4, 0.5\}$ respectively. We apply early stopping on the validation MAP.

Validation. Hyper-parameters listed above are tuned on the MAP on the validation set using grid search.

Evaluation. We use 4 standard evaluation metrics: MAP, Recall, Precision and nDCG on the top 1,000 documents. These metrics are implemented with *pytrec-eval* [11]. We use a two-tailed paired t-test with Bonferroni correction to measure statistically significant differences between the evaluation metrics. Because the NFCorpus has only 3,633 documents we can evaluate every (query, document) pair in a reasonable amount of time in order to avoid relying on a re-ranking strategy [9, 31]. Therefore the recall of KTRel and the NLTR baselines is not upper-bounded by a prior ranking stage.

4.3 Baselines

We compare KTRel with three types of baselines methods: BoW and NLTR for IR and Pre-trained BERT encoder.

BoW. As suggested by Yang et al. [30], we use Okapi BM25 [22] and Okapi BM25 with RM3 pseudo-relevance feedback [16] as our BoWs baselines. Stemming, indexing and evaluation of BM25 and BM25-RM3 are performed by Terrier [20]. Hyper-parameter values are tuned on the validation MAP with grid search.

NLTR. DUET [18], KNRM [29], DRMM [8] and Conv-KNRM [4] are used as NLTR baselines for IR. Training and evaluation of these models is performed with *MatchZoo* [10]. Hyper-parameter values are tuned on the validation MAP with random search over 10 runs. We use the tuner provided by *MatchZoo* to sample values from the hyper-parameter space associated with each model.

BERT. We also compare KTRel against BERT [6] encoder: a state of the art language representation model. We use the *"bert-base-uncased"* model provided by *Hugging Face* [28], pre-trained on the BooksCorpus [32] and English Wikipedia. When training, we fine tune the last layer of the model.

BioBERT. Finally, we compare KTRel against BioBERT [14]: a biomedical language representation that train BERT language model on large-scale biomedical corpora.

To study the usefulness of combining concepts and words, we also train and evaluate separately the part of the KTRel architecture that uses only concepts (denoted by KTRel-C) and the part that uses only words (denoted by KTRel-W) as pictured in Figure 1. KTRel-C and KTRel-W are trained independently using the same setup as described in subsection 4.2

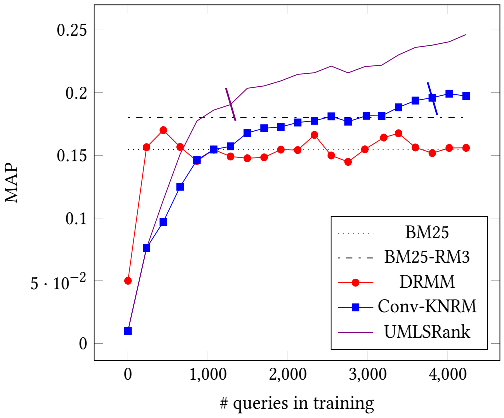

Figure 2: MAP on all query fields against the # of queries used in training. \ indicate when a NLTR model has enough queries in training to achieve statistically significant improvement compared to BM25-RM3 (p-value < 0.05)

5 RESULTS

The performance of KTRel against baselines are shown in Table 1. In the following, we propose empirical answers to several research questions.

Is it useful for ranking to use both words and medical concepts? KTRel outperforms all the NLTR baselines on all metrics with statistical significance. The fact that KTRel also outperforms both KTRel-W and KTRel-C provides empirical evidence that IR in specialized domain can benefit from combining pre-trained concept representations with pre-trained word representations.

Can KTRel outperform a strong BoW baseline? KTRel achieves statistical significance against BM25 with RM3 query expansion on most metrics. The overall ranking is largely improved by KTRel: +33.9% w.r.t MAP. The notable exceptions are nDCG@5 and nDCG@10: even if KTRel outperforms BM25-RM3 in terms of nDCG@5 (+1.2%) and nDCG@10 (+4.4%), it does not achieve statistical significance. Interestingly, KTRel do achieve statistical significance against P@5 (+16.3%) and P@10 (+26.5%). The difference between P@k and nDCG@k is that precision only looks at the proportion of relevant documents whereas nDCG@k emphasizes more on the ranking itself and takes into account the relevance levels of documents. Therefore, we can conclude that even if KTRel is able to retrieve more relevant documents in the top-k results, BM25-RM3 is still a strong baseline when it comes to the ranking of the top-k documents.

How is BERT performing in IR in specialised domains? BERT performs worst than BM25 despite it's success in several NLP tasks [6]. Because BioBERT outperforms BERT with a high margin, we can conclude that, when using a language model in a specialized domain, it is essential to pre-train the model on text of the same domain.

Can baseline NLTR models outperform a strong BoW baseline? First, we notice that the DUET and KNRM models perform worst than BM25. The reason is probably that these models were developed on much larger datasets [18, 29] than the NFCorpus. Second, DRMM performs slightly better than BM25 (+6.7% w.r.t MAP,

model	P@5	P@10	P@20	nDCG@5	nDCG@10	nDCG@20	MAP	Recall
BM25	0.2846^-	0.2419^-	0.1733^-	0.3524^-	0.3267^-	0.3038^-	0.1548^-	0.4740^-
BM25-RM3	0.3056	0.2603	0.1912	0.3664	0.3431	0.3249	0.1801	0.6249
DUET	0.1967^-	0.1840^-	0.1561^-	0.1857^-	0.1883^-	0.1892^-	0.1264^-	0.7673^+
KNRM	0.2082^-	0.1887^-	0.1617^-	0.1914^-	0.1914^-	0.1936^-	0.1216^-	0.7764^+
DRMM	0.2940	0.2489	0.1819	0.3540	0.3330	0.3116	0.1651	0.7051^+
Conv-KNRM	0.3146	0.2865	0.2378^+	0.3010^-	0.3090^-	0.3138	0.2110^+	0.8143^+
BERT	0.2084^-	0.1998^-	0.1536^-	0.2090^-	0.2196^-	0.2062^-	0.1567^-	0.7847^+
BioBERT	0.3148	0.2989^+	0.2373^+	0.3508	0.3377	0.3228	0.2358^+	0.8265^+
KTRel-C	0.3127	0.2889^+	0.2304^+	0.3285^-	0.3295^-	0.3094^-	0.2194^+	0.8047^+
KTRel-W	0.3204^+	0.3008^+	0.2377^+	0.3465	0.3369	0.3141	0.2228^+	0.8187^+
KTRel	**0.3554**$^+$	**0.3294**$^+$	**0.2498**$^+$	**0.3708**	**0.3584**	**0.3424**$^+$	**0.2411**$^+$	**0.8520**$^+$

Table 1: Performance comparison of different models on the NFCorpus. + (resp. -) denotes a significant performance gain (resp. degradation) against BM25-RM3 (p-value < 0.01). Best performances are highlighted in bold.

+3.3% w.r.t P@5 and +0.5% w.r.t nDCG@5) but it does not manages to outperform BM25-RM3. Finally, Conv-KNRM is the only NLTR baseline that manages to outperform BM25 and BM25-RM3 w.r.t MAP, Precision and Recall (but not w.r.t nDCG). These results empirically confirm that NLTR models have not achieved significant breakthroughs in IR [9].

Do transformer encoders provide useful representations for IR in specialised domains? KTRel-W and KTRel-C perform similarly to the best NLTR baseline. Moreover, the fact that these models rely on a simple cosine similarity between the query and the document representation empirically demonstrate that transformer encoders do produce useful representations for IR.

How do NLTR models affect the recall? Since the number of documents is limited in the NFCorpus, we do not rely on a reranking strategy based on BM25 [31]. Therefore the recall of the NLTR models is not upper bounded by the recall of BM25. The results indicate that the gain of KTRel in terms of recall is significant compared to BM25 (+78.6%) and BM25-RM3 (+36.3%). This happens because BoW models can only retrieve documents that contain terms of the query whereas NLTR models do not have this restriction.

How much data is needed to outperform BM25-RM3 with a neural network? As we can see on Figure 2, the number of queries required for an NLTR model to outperform BM25 or BM25-RM3 baselines varies depending on the model under consideration. On the NFCorpus, about 200 queries are required by DRMM to obtain results comparable to those of BM25. This is due to the fact that DRMM is a model with very few parameters (≈ 450) and therefore does not need a lot of data to converge. However, and for the same reasons, DRMM does not benefit from a lot of training data and does not even outperform the BM25-RM3 reference model when given more training queries. The Conv-KNRM model manages to outperform BM25 and BM25-RM3, but about 3,000 queries are required in the training set for Conv-KNRM to outperform BM25-RM3 over NFCorpus. It seems that less data (≈ 1,500 queries) are needed for the KTRel model. This suggests that the use of concepts can be useful in resource-constrained scenarios. These results also confirm that training an NLTR model on a collection containing only a few hundred queries is a very difficult task. This may explain why significant breakthroughs have yet to be achieved by NLTR on standard IR collections that contain only a few hundred queries at best and a few dozen at worst.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we propose KTRel: a transformer-based NLTR model that uses both words and concepts for IR in specialized domains. We empirically demonstrate that adding concepts to a neural learning-to-rank model is useful for IR in the medical domain. We show that transformer encoders provide effective sequence representations for IR. We also empirically confirm that BM25 with RM3 query expansion is still a strong baseline, especially with respect to high-precision metrics. As future work we plan to evaluate KTRel on more collections and other specialized domains. To make our model scalable to larger collections, we will adapt it to learn word-based and concept-based sparse representations compatible with an inverted index as suggested by Zamani et al. [31].

REFERENCES

[1] Olivier Bodenreider. 2004. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research* 32, suppl_1, D267–D270.

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *NIPS*. 2787–2795.

[3] Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A Full-Text Learning to Rank Dataset for Medical Information Retrieval. In *ECIR*. 716–722.

[4] Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan 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* (Marina Del Rey, CA, USA) *(WSDM '18)*. ACM, New York, NY, USA, 126–134. https://doi.org/10.1145/3159652.3159659

[5] Jeffrey Dalton, Laura Dietz, and James Allan. 2014. Entity Query Feature Expansion Using Knowledge Base Links. In *Proceedings of the 37th International ACM SIGIR Conference on Research &#38; Development in Information Retrieval* (Gold Coast, Queensland, Australia) *(SIGIR '14)*. ACM, New York, NY, USA, 365–374. https://doi.org/10.1145/2600428.2609628

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Lorraine Goeuriot, Liadh Kelly, Wei Li, Joao Palotti, Pavel Pecina, Guido Zuccon, Allan Hanbury, Gareth J. F. Jones, and Henning Müller. 2014. ShARe/CLEF eHealth Evaluation Lab 2014, Task 3: User-centred health information retrieval. In *Proceedings of CLEF 2014*. Sheffield, United Kingdom.

[8] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce 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*. ACM, 55–64.

[9] Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W Bruce Croft, and Xueqi Cheng. 2019. A Deep Look into Neural Ranking

Models for Information Retrieval. *arXiv preprint arXiv:1903.06902* (2019).

[10] Jiafeng Guo, Fan Yixing, Ji Xiang, and Cheng Xueqi. 2019. 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* (Paris, France) *(SIGIR'19)*. ACM, New York, NY, USA, 1297–1300. https://doi.org/10.1145/3331184.3331403

[11] Christophe Van Gysel and Maarten de Rijke. 2018. Pytrec_eval: An Extremely Fast Python Interface to trec_eval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*. 873–876. https://doi.org/10.1145/3209978.3210065

[12] D. P. Kingma and J. Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. http://arxiv.org/abs/1412.6980

[13] Guillaume Lample and Alexis Conneau. 2019. Cross-lingual Language Model Pretraining. *arXiv e-prints*, Article arXiv:1901.07291 (Jan 2019), arXiv:1901.07291 pages. arXiv:cs.CL/1901.07291

[14] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *CoRR* abs/1901.08746 (2019). arXiv:1901.08746 http://arxiv.org/abs/1901.08746

[15] Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2018. Entity-Duet Neural Ranking: Understanding the Role of Knowledge Graph Semantics in Neural Information Retrieval. *arXiv preprint arXiv:1805.07591* (2018).

[16] Yuanhua Lv and ChengXiang Zhai. 2009. A comparative study of methods for estimating query language models with pseudo feedback. In *CIKM*. 1895–1898.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[18] Bhaskar Mitra, Fernando Diaz, and Nick 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*. International World Wide Web Conferences Steering Committee, 1291–1299.

[19] Gia-Hung Nguyen, Lynda Tamine, Laure Soulier, and Nathalie 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*. 160–170. https://doi.org/10.1007/978-3-319-59758-4_17

[20] Iadh Ounis, Gianni Amati, Vassilis Plachouras, Ben He, Craig Macdonald, and Douglas Johnson. 2005. Terrier information retrieval platform. In *European Conference on Information Retrieval*. Springer, 517–519.

[21] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1, 8 (2019).

[22] Stephen Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR*. 232–241.

[23] Ying Shen, Yang Deng, Min Yang, Yaliang Li, Nan Du, Wei Fan, and Kai Lei. 2018. Knowledge-aware Attentive Neural Network for Ranking Question Answer Pairs. In *SIGIR*. Springer, 901–904.

[24] Luca Soldaini and Nazli Goharian. 2016. Quickumls: a fast, unsupervised approach for medical concept extraction. In *MedIR workshop, SIGIR*.

[25] Luca Soldaini and Nazli Goharian. 2017. Learning to rank for consumer health search: a semantic approach. In *ECIR*. Springer, 640–646.

[26] Hanna Suominen, Sanna Salanterä, Sumithra Velupillai, Wendy W Chapman, Guergana Savova, Noemie Elhadad, Sameer Pradhan, Brett R South, Danielle L Mowery, Gareth JF 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*. Springer, 212–231.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, et al. 2017. Attention is all you need. In *NIPS*. 5998–6008.

[28] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Transformers: State-of-the-art Natural Language Processing. arXiv:cs.CL/1910.03771

[29] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell 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*. ACM, 55–64.

[30] Wei Yang, Kuang Lu, Peilin Yang, and Jimmy Lin. 2019. Critically Examining the "Neural Hype": Weak Baselines and the Additivity of Effectiveness Gains from Neural Ranking Models. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*. 1129–1132. https://doi.org/10.1145/3331184.3331340

[31] Hamed Zamani, Mostafa Dehghani, W Bruce Croft, Erik Learned-Miller, and Jaap Kamps. 2018. From neural re-ranking to neural ranking: Learning a sparse representation for inverted indexing. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 497–506.

[32] Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning Books and Movies: Towards Story-like Visual Explanations by Watching Movies and Reading Books. *CoRR* abs/1506.06724 (2015). arXiv:1506.06724 http://arxiv.org/abs/1506.06724

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