Digital Personalized Health and Medicine
L.B. Pape-Haugaard et al. (Eds.)

doi:10.3233/SHTI200152

Evaluation of Document Retrieval Systems on a Medical Corpus in French: Indexation vs. Feature Learning

Arnaud ROBERT[a,1], Francis DAMACHI[a,1], Mina BJELOGRLIC[a,1], Jean-Philippe GOLDMAN[a,1], and Christian LOVIS[a,1]
[a]*Division of Medical Information Sciences, University Hospitals of Geneva and University of Geneva, Geneva, Switzerland*

Abstract. This paper presents five document retrieval systems for a small (few thousands) and domain specific corpora (weekly peer-reviewed medical journals published in French) as well as an evaluation methodology to quantify the models performance. The proposed methodology does not rely on external annotations and therefore can be used as an ad hoc evaluation procedure for most document retrieval tasks. Statistical models and vector space models are empirically compared on a synthetic document retrieval task. For our dataset size and specificities the statistical approaches consistently performed better than its vector space counterparts.

Keywords. Medical Information Retrieval, Document Retrieval, Natural Language Processing, Word Embedding.

1. Introduction

The field of Information Retrieval (IR) is a well-studied and yet still critical field of research. In the last few decades, additional effort has been made in the biomedical domain with the digitalization of medical data and the multiplication of biomedical scientific literature [1,2]. The goal of the present paper is to give a clear methodology in evaluating several IR systems on small (few thousands) and domain specific corpora (medical literature in French). For a specific query, document ranking is determined by a numerical score assigned by the IR system [3]. On the one side, early IR systems in the early 60's were based on Boolean and statistical approaches [4]. In term-based approaches documents are represented by a predefined arbitrary feature with no encoded semantic information (words are represented by an index in a vocabulary). On the other side, vector space (VS) models represent documents as a composition of vector representation of terms [5]. In VS each word is encode as a vector in a high dimensional space allowing to encode semantic information. Independently of the representation

[1] Authors contributed equally to this work, Division of Medical Information Sciences, University Hospitals of Geneva and University of Geneva, Geneva, Switzerland E-mail: Firstname.Lastname@hcuge.ch

used, most Document Retrieval systems compute a ranking between a query and all the documents in the corpus [3]. The documents with the highest score (rank) are considered most relevant and are returned [6,7]. Statistical models (*Boolean* and *Weighted*), vector space models (*Word2Vec-Mean* and *Word2Vec-Weighted, Word2Vec-Max*), are compared on a dataset consisting of 4,201 articles of weekly peer-reviewed medical journals published in French. All models are evaluated empirically on the same document retrieval task.

2. Data/corpus

The source of the dataset is the *Revue Medical Suisse* (RMS). The dataset consists of 4,201 articles of weekly peer-reviewed medical journals published in French in XML format spanning from years 2014 to 2019. Sections of the document such as the abstract, title, body, conclusion and references can be distinguished by specific XML tags. In order to evaluate the models performance we kept articles for which title, abstract and body were available and in French, which resulted into 1,401 documents. All models benefited from the same preprocessing steps. The first step consists into the extraction of the documents from the XML format to plain text. Then, each document is converted into lemmatized tokens (49,573 unique terms). Lemmatization is done by *Spacy*'s pre-trained statistical models for French [8]. Stopwords are removed (they do not add any semantic information), verbs are removed (add small semantic information), and word that occurs less than 2 were removed (to filter out mostly misspellings and extremely rare terms). Following this procedure, approximately 42% of words were filtered out, leaving a vocabulary size of 28,791 terms.

3. Methodology

We investigate different methodologies to rank documents according to a query. The proposed models fall in one of the two following categories: Term-based or VS models. The term-based models evaluated (*Boolean* and *Weighted* models) represents words as an index in a reference vocabulary while VS models represent words as vector in a high dimensional space [5], which encode semantic properties of the word. The evaluated VS models (*Word2Vec-Mean*, *Word2Vec-Weighted*, *Word2Vec-Max*) represent each term as a m-dimensional vector v using the well-known *Word2Vec* algorithm [5], where m is a user chosen parameter (called embedding size).

3.1. Term-based Models

In the *Boolean model*, for a vocabulary size of n words, the model encodes a text as an n-dimensional vector containing a '1' for each word present in the text and '0' otherwise. In order to evaluate the similarity between a query q and a document d the model first normalizes the two vectors: $\overline{q} = \frac{q}{\sum_i q_i}$, $\overline{d} = \frac{d}{\sum_i d_i}$, and then compute the cosine similarity between $\overline{q}$ and $\overline{d}$, where the cosine similarity between two vectors a and b is defined as follow:

$$sim(a,b) = \frac{\sum_i a_i\, b_i}{\sqrt{\sum_i a_i^2 \cdot \sum_i b_i^2}} \quad (1)$$

The main drawback of the *Boolean model* is that it gives the same importance to all the words in a document. To include the word importance, the *Weighted model* uses the TF-IDF (for term frequency - inverse document frequency) metric that weight the importance of a term in a document. More formally TF-IDF will compute the weight of a term as follow:

$$\widetilde{w}_d(t) = freq_d(t) * (log\, \frac{D+1}{d(t)+1} + 1),$$
$$w_d(t) = \frac{\widetilde{w}_d(t)}{\sum_{t'} \widetilde{w}_d(t')}, \text{ with } \widetilde{w}_d(t) = freq_d(t) * (log\, \frac{D+1}{d(t)+1} + 1) \quad (2)$$

where $freq_d(t)$ denotes the number of times the term t appears in the document d, i.e the term frequency and $d(t)$ denotes the number of documents that contain the term t, i.e. the document frequency. Query and documents are compared using cosine similarity (see eq. 1) [9].

3.2. Vector Space Models

For all VS models word representation was computed with *Word2Vec* algorithm, with an embedding size of 32, a context window size of 7, and 64 negative samples.
In *Word2Vec-Mean model*, documents are represented as the mean of their word vectors: $d_j = \frac{1}{|d_j|}\sum_{i \in d_j} v_i$, where v_i is the m-dimensional vector representing word *i*. Queries are represented with the same methodology. For a given query, the relevance of a document is computed with the cosine similarity (see eq. 1). The main drawback of the previous method is that it assigns the same weight to all words of the document. An alternative (*Word2Vec-Weighted model)* is to use the TF-IDF weights (see eq. 2), and represent a document as follow: $d_j = \frac{1}{\sum_i w_{d_j}(i)}\sum_{i \in d_j} w_{d_j}(i) v_i$

When representing a document with the above methods, the mean can involve a potentially large number of vector (several hundreds), hence drowning the information in the mass. The *Word2Vec-Max model* solves this problem by focusing on the few words that are the most relevant regarding the query. For each term of the query q the algorithm find the closest term (w.r.t. the cosine similarity, see eq. 1) in the document and average them to compute the final similarity score: $\sum_{q_i \in q} max_{d_j \in d}(sim(q_i, d_j))$.

4. Experiments

The dataset for testing consists of 1400 medical articles from RMS Journal. In order to evaluate the above information retrieval systems we used an article's title as a query. The precision of the model is then evaluated, by looking if the corresponding article appears in the top-N most relevant documents (N=1, 5, 10). Performance of some methods might fluctuate depending on the length of the documents composing the corpus, therefore, experiments were conducted on 3 different corpuses:

- Corpus A, contains only the abstracts of the articles (25 to 184 words);

- Corpus B, contains only the body of the articles (431 to 8,839 words);
- Corpus A+B, contains both the abstract and the body of the articles.

Titles, used as queries, have from 1 to 29 words (mean=9.7 words) and the total number of words is the whole corpus is 1,853,939 (in titles + abstracts + bodies).

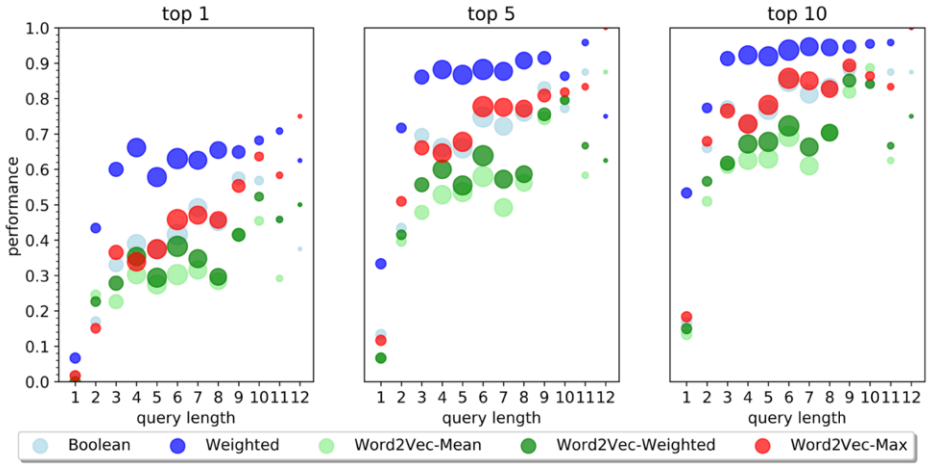

Figure 1. Performance according to query length. From left to right, it shows the top1, top5, and top10 performance. Models are color coded, and the size of the marker is proportional to the support size (i.e. number of queries).

For all the documents, all models, and the three different corpuses, the top1, top5, and top10 performances are shown in Table 1. The best performance for almost all sub-tasks is obtained by the *Weighted model*. In order to quantify the impact of the query length, Figure 1 shows the performance of each model according to the query length. The dominance of the *Weighted model* (in dark blue) is confirmed by this new perspective. The performance is the ratio between the number of queries where the correct document was in a specific ranking (i.e. top1, top5, top10) and the total number of queries. We achieve up to 94.5% of retrieval with top10 metric for 7 concept long queries.

Table 1. Shows the top1/top5/top10 performance of the five models on 1400 different queries

Method	Abstracts only	Bodies only	Abstracts + bodies
Boolean	801/1071/1138	564/963/1084	595/979/1092
Weighted	876/**1186**/**1240**	**838**/1195/**1276**	**861**/**1218**/**1295**
Word2Vec- Mean	422/785/920	405/749/900	421/766/915
Word2Vec- Weighted	486/804/949	458/812/941	474/831/955
Word2Vec-Max	**882**/1135/1192	572/982/1098	598/1004/1114

5. Discussion

First, it is important to note that the task is prone to advantage index-based methods since it is very likely that words that appear in the query (title) will occur in the target article. Hence, it might explain the very good results obtain by index based methods. Secondly

it seems that VS space models that compute document representation using all terms in the documents fail to provide a good representation, as suggested by the poor results obtained by *Word2Vec-Mean* and *Word2Vec-Weighted models*. This is also confirmed by the performance gain seen with Word2Vec-Max (which use less terms to represent documents).

6. Conclusion

This paper shows and evaluates how various document representations can impact the performance of a Document Retrieval system. The presented models are separated into two groups: term based and vector space models. In total we described and evaluated 5 models: *Boolean, Weighted, Word2Vec-Mean, Word2Vec-Weighted, Word2Vec-Max models*. The relevance of a document with respect to a query was computed using the cosine similarity between their appropriate representations. By varying the query length we see how the models perform in retrieving the top 1/5/10 documents.
Future work should focus on: (i) evaluating the presented methods against a much larger corpus (rather than only 1400 articles); (ii) investigating more in depth document representation (the *Word2Vec Max* use only a small subset of words, as many as the query length, to represent a document, this information loss impact the severely the document representation); (iii) efficient implementation of those systems to scale to larger database. The methodology proposed in this paper can be used as an ad hoc evaluation procedure for most document retrieval tasks in small and medium size text databases, in order to adopt the most effective IR strategy.

References

[1] Roberts K, Simpson M, Demner-Fushman D, Voorhees E, Hersh W. State-of-the-art in biomedical literature retrieval for clinical cases: a survey of the TREC 2014 CDS track. Inf Retr J 2016 Apr 1;19(1):113–148. [doi: 10/ggbx8s]
[2] Goeuriot L, Jones GJF, Kelly L, Müller H, Zobel J. Medical information retrieval: introduction to the special issue. Inf Retr J 2016 Apr;19(1–2):1–5. [doi: 10/ggbx9j]
[3] Singhal A. Modern Information Retrieval: A Brief Overview. Bull IEEE Comput Soc Tech Comm Data Eng 2001;24:35–43.
[4] Bordogna G, Pasi G. A fuzzy linguistic approach generalizing Boolean Information Retrieval: A model and its evaluation. J Am Soc Inf Sci 1993;44(2):70–82. [doi: 10/cvk4q4]
[5] Mikolov T, Chen K, Corrado G, Dean J. Efficient Estimation of Word Representations in Vector Space. ArXiv13013781 Cs [Internet] 2013 Jan 16 [cited 2019 May 17]; Available from: http://arxiv.org/abs/1301.3781
[6] Kusner MJ, Sun Y, Kolkin NI, Weinberger KQ. From Word Embeddings To Document Distances. Int Conf Mach Learn 2015;957–966.
[7] Ganguly D, Roy D, Mitra M, Jones GJF. Word Embedding based Generalized Language Model for Information Retrieval. Proc 38th Int ACM SIGIR Conf Res Dev Inf Retr - SIGIR 15 [Internet] Santiago, Chile: ACM Press; 2015 [cited 2019 Sep 23]. p. 795–798. [doi: 10/gf8tnc]
[8] French · spaCy Models Documentation [Internet]. French. [cited 2019 Aug 7]. Available from: https://spacy.io/models/fr
[9] Xu B, Lin H, Lin Y, Ma Y, Yang L, Wang J, Yang Z. Improve Biomedical Information Retrieval Using Modified Learning to Rank Methods. IEEE/ACM Trans Comput Biol Bioinform 2018 Nov 1;15(6):1797–1809. [doi: 10/ggbx8z]