Non-Parametric Subject Prediction

Shenghui Wang[1(✉)], Rob Koopman[1], and Gwenn Englebienne[2]

[1] OCLC Research, Schipholweg 99, 2316XA Leiden, The Netherlands
{shenghui.wang,rob.koopman}@oclc.org
[2] University of Twente, Hallenweg 19, 7522NH Enschede, The Netherlands
g.englebienne@utwente.nl

Abstract. Automatic subject prediction is a desirable feature for modern digital library systems, as manual indexing can no longer cope with the rapid growth of digital collections. This is an "extreme multi-label classification" problem, where the objective is to assign a small subset of the most relevant subjects from an extremely large label set. Data sparsity and model scalability are the major challenges we need to address to solve it automatically. In this paper, we describe an efficient and effective embedding method that embeds terms, subjects and documents into the same semantic space, where similarity can be computed easily. We then propose a novel Non-Parametric Subject Prediction (NPSP) method and show how effectively it predicts even very specialised subjects, which are associated with few documents in the training set and are not predicted by state-of-the-art classifiers.

Keywords: Random projection · Subject prediction · Non-parametric method · Semantic embedding

1 Introduction

Because of the ever-increasing number of documents that information systems deal with, automatic subject indexing, i.e., identifying and describing the subject(s) of documents to increase their findability, is one of the most desirable features for many such systems. Subject index terms are normally taken from knowledge organization systems (e.g., thesauri, subject headings systems) and classification systems (e.g., dewey decimal classification) which easily contain tens or hundreds of thousands terms or codes. Automatically assigning a small set of most relevant subjects from the huge label space – the Extreme Multi-label Text Classification (XMTC) problem – is therefore very difficult. Data sparsity and scalability are the major challenges.

In this paper, we solve this in two steps. First, we propose a novel embedding method which extends random projection by weighting and projecting raw term embeddings orthogonally to an average language vector, thus improving the discriminating power of resulting term embeddings, and build more meaningful document embeddings by assigning appropriate weights to individual terms.

A. Doucet et al. (Eds.): TPDL 2019, LNCS 11799, pp. 312–326, 2019.
https://doi.org/10.1007/978-3-030-30760-8_27

Subjects are treated as special terms which get embedded into the same semantic space where terms and documents live. Secondly, we propose a novel Non-Parametric Subject Prediction (NPSP) method to predict subjects for unseen documents. We compare this method with the state-of-the-art deep learning method and the direct subject-document-similarity based method.

2 Related Work

Automatic Subject Indexing. According to [9], there are three groups of approaches: text categorisation, where supervised machine learning is used to predict subjects from features extracted from documents [24]; document clustering, where unsupervised machine learning is used to group documents and the resulting groups are then associated with certain topics [6]; and document classification, where no training data is (necessarily) used but string matching relates the subjects to the documents [2,7,8]. In this paper, we focus on data-driven text categorisation, and it is still an open research question how this problem should best be approached.

Extreme Multi-label Text Classification. The goal of automatic subject prediction is to assign a small subset of relevant subject labels (subjects) to a document, taken from tens or hundreds of thousands target labels. This remains a difficult problem and is a form of Extreme Multi-label Text Classification (XMTC) [3,17,23], where the prediction space often consists of hundreds of thousands to millions of labels. Data sparsity and scalability are the major challenges. Unlike traditional binary or multi-class classification problems, in multi-label classification the target labels are neither independent nor mutually exclusive, thus making the modelling of the relationship between the documents and the labels challenging.

There are four categories of solutions: (1) 1-vs-All classification [22], (2) Embedding-based [3,25], (3) Tree-based [23], and (4) Deep learning methods [12,17].[1]. Each of these approaches has pros and cons. In 1-vs-all classification, a separate classifier is trained for each subject and is used to decide whether that particular subject is applicable for a given document: such an approach ignores the correlation between subjects and does, therefore, not make optimal use of the available data. Furthermore, this is not practical for extreme multi-label classification, because of the number of classifiers that need to be trained and evaluated for prediction is too high. Tree-based methods and Deep-learning methods can model the correlation between labels and can be extremely powerful given enough training data, but they are hard to interpret and suffer from the inevitable lack of data for the rare labels that make up so much of the heavy tail of the label distribution.

Embedding-based approaches [3,25] aim to address the data sparsity issue by projecting the high-dimensional label vectors to their low-dimensional embeddings during training. When classifying a new sample, a decompression process

[1] http://manikvarma.org/downloads/XC/XMLRepository.html.

is used to map the predicted embedding back to the original high-dimensional space. These methods are powerful, relatively fast and well understood, but they either rely on low-rank assumptions on the inter-subject distance matrix [27] or require clustering of the data which, in turn, depends on an arbitrarily chosen number of clusters and their initial cluster parameters [3].

In contrast, our approach does not make any assumptions on the rank of the inter-subject distance matrix. Instead, we use random projection to embed the co-occurrence matrix of the terms that constitute the documents and the subjects we try to predict. By modeling terms in function of their relationship with subjects, we obtain a very flexible embedding that also allows us to embed single documents (by considering which terms occur in them), directly incorporate the statistical relationship between terms and subjects and avoid needing to arbitrarily cluster the documents to resolve the high rank of the global inter-subject distance matrix.

3 Method

We propose to apply a Random Projection based embedding method to embed both terms and subjects in a semantic space as described in Sect. 3.1. This allows us to compute a vector representation of any document, seen or unseen. When classifying an unseen document, we propose two methods: first to compute its vector representation and compute that vector's similarity to the vector representations of all subjects, thus allowing us to rank subject candidates in order of decreasing similarity, to find the most appropriate ones for that document, as described in Sect. 3.2. Second, we propose computing similarities between the query document and documents from the training set: these are annotated with subjects and provide us with a way to better assess the validity of a subject to the query document, as explained in Sect. 3.3.

3.1 Ariadne Semantic Embedding

Let a document be a set of words for which term co-occurrence is relevant[2] and which can be meaningfully annotated with a subject. Let n_D be the total number of documents in the training set, n_S the number of subjects, n_V the number of *frequent* terms,[3]. A term is considered frequent when it occurs in more than k documents in the corpus, where K is flexible depending on the size of the corpus. In addition, let n_E be the total number of entities we want to embed (in our case $n_E = n_S + n_V$) and D the chosen dimensionality of the embedding vectors.

Building on our previous work [13–15], we embed the relevant entities by Random Projection [1,11] of their weighted co-occurrence:

$$\mathbf{C}'_{[n_E \times D]} = \mathbf{C}_{[n_E \times n_S]} \mathbf{R}_{[n_S \times D]} \tag{1}$$

[2] In general, a document could therefore be a sentence, a paragraph, a fixed-size window, a bibliographic record, *etc.*; in our case, documents are scientific publications.

[3] Terms could be words, n-grams or phrases. In our work, common phrases are automatically detected using the method described in [19].

where $\mathbf{C}'$ is the matrix of embedding vectors, $\mathbf{C}$ is the weighted co-occurrence matrix of different terms and $\mathbf{R}$ is a random matrix. In this work, we focus on subjects, and in this particular use case we observe that it is useful to use co-occurrence of the entities with the subject labels only. In general, term-term co-occurrences are more common and well-suited, in which case we would have n_V columns to the matrix C and n_V rows to $\mathbf{R}$.

Weighted Co-occurrence Counts. To improve the robustness of the approach, we weight the co-occurrence matrix $\mathbf{C}$ to reduce the effect of terms that are extremely common in certain documents and of terms that occur in the vast majority of documents. We use the terms' average TFIDF score in the training documents, modified as follows. Let each element c_{t_i,s_j} of C be the weighted co-occurrence count of entity t_i and subject s_j. For notational simplicity, we will use $s \in d$ to indicate that document d is annotated with subject s and $t \in d$ to indicate the document contains word t. Further, let d_t be the total number of documents that term t occurs in; $\boldsymbol{r}_s$ a D-dimensional "random vector" for subject s, *i.e.*, $\boldsymbol{r}_s$ is a row of $\mathbf{R}$ (In our implementation, this vector is binary and contains an equal number of $+1$ and -1, thus making computations very efficient [13]). Experimentally, we verified that the traditional IDF weighting factor of $\log \frac{N}{d_t}$ suppresses frequent terms too much, and replace it by a factor of $\sqrt{N/d_t}$, which has a similar effect but a longer tail and can also be seen as the normalisation constant of the t-test statistic [18]. For the TF factor, we use a factor of $1 + \log c_t(d)$, where $c_t(d)$ is the term count of term t in document d, and ignore the constant N which cancels out in the subsequent normalisation. The co-occurrence counts $c_{t,s}$ are, therefore, replaced with weighed counts so that the elements of $\mathbf{C}$ become:

$$c_{t,s} = \sum_d I(t \in d)\, I(s \in d)\, \frac{1 + \log c_t(d)}{\sqrt{d_t}} \tag{2}$$

where $I(t \in d)$ is an indicator function which is 1 if entity t occurs in document d, and zero otherwise. After projection, each row of $\mathbf{C}'$, denoted $\boldsymbol{v}_t$ in the remainder of this document, is a vector embedding of term t.

Orthogonal Projection. Traditional models discard both very infrequent words (because they are too rare for the model to be able to capture their semantics from the training data) and very frequent words (so-called "stop words" because they do not provide any semantically useful information). In our approach, we give a continuous weight to terms based on how frequently they occur and compute the average "language vector" of the corpus, $\boldsymbol{v}_a$, the sum of all the rows of $\mathbf{C}'$. Unsurprisingly, this vector is very similar to the average vector of stop words. Intuitively, words are increasingly more informative as they differ more from the average vector. By this reasoning, we project[4] word vectors on the

[4] We use projection rather than subtracting $\boldsymbol{v}_a$ to prevent orthogonal vectors from gaining undue importance.

orthogonal hyperplane to $\boldsymbol{v}_a$: $\boldsymbol{v}_t^* = \boldsymbol{v}_t - (\boldsymbol{v}_t \cdot \boldsymbol{v}_a)\boldsymbol{v}_a$, resulting in a representation where the uninformative component of terms is eliminated, and normalise the vectors to have unit length. When computing document vectors, we down-weight terms according to their similarity to $\boldsymbol{v}_a$ (see Eq. 3). This step is crucial to get distinctive document embeddings.

As a nice side effect, projection makes it possible to handle multilingual corpora. The vocabulary of one language tends to be largely orthogonal to that of other languages (since words of one language tend to co-occur almost exclusively with words of the same language), so that projection using one language's average vector does not have much effect on the terms in other languages. This makes it possible to handle different languages effectively, within the same vector space.

Term Weight Assignment and Document Embedding. Using the projection described above, the component that differentiates a term from the average vector is kept as its final embedding. Similarly, how different a term is from $\boldsymbol{v}_a$ also indicates how much that term contributes to the semantics of a document it is part of. In fact, we can interpret the cosine similarity as a lower bound on the mutual information (MI) between the two vectors [5]. In order to give a higher weight to the most informative terms, we assign a higher weight to words with lower MI to $\boldsymbol{v}_a$ by setting the final weight of each term to be $w_t = 1 - \cos(\boldsymbol{v}_t, \boldsymbol{v}_a)$.

With the frequent terms' embedding vectors and their proper weights, we can compute document embedding as the weighted average of its component terms' embeddings. For a document d, we obtain a set of normalised vectors $\boldsymbol{v}_{t_1}^*, \ldots, \boldsymbol{v}_{t_n}^*$, where n is the number of terms in document d and $\boldsymbol{v}_{t_i}^*$ is the final embedding vector for term t_i. The embedding of document d is calculated as follows:

$$\boldsymbol{v}_d = \frac{\sum_{i=1}^{n} w_{t_i} \cdot \boldsymbol{v}_{t_i}^*}{\sum_{i=1}^{n} w_{t_i}}. \tag{3}$$

where w_{t_i} is the weight for term t_i and out-of-vocabulary words are ignored. Note how term and document vectors all have unit length, making similarity computations elegant and effective.

3.2 Prediction by Subject-Document Similarity

Once subjects and documents are embedded in the same semantic space, it is straightforward to calculate the similarity between any subject and any document. Notice how we can interpret $w_{t_i} / \sum_t w_t$, as the document-conditional probability distribution over the terms, and $\boldsymbol{v}_d$ as the expectation of the embedding of the query document by marginalising out its component terms, while the subject embedding corresponds to the empirical mean of the training documents that were annotated by that subject. If we assume an isotropic distribution for documents annotated by a given subject, then the most related subjects to a document are simply the ones closest to the document, *i.e.*, the subjects with the highest cosine similarities to the document itself.

Algorithm 1. Non-Parametric Subject Prediction (NPSP)

```
1: function SUBJECT_PREDICTION(training documents 𝒟, unseen document u, k)
2:     D ← sort_{S(v_d, v_u)}(𝒟)   ▷ Order the document embeddings by decreasing similarity to the unseen document u
3:     ∀s : w_s ← S(v_s, v_u) ▷ Initialise the weight w_s with the similarity between the subjects and u
4:     for all documents d ∈ D_{1...k} do   ▷ For the k documents closest to u
5:         for all subjects s of document d do
6:             w_s ← w_s + S(v_d, v_u)   ▷ Add the similarity of the documents to w_s
7:     return sort_{w_s}({s})   ▷ Return a ranked list of subjects according to their weights
```

3.3 NPSP: Non-Parametric Subject Prediction

In practice, the distribution of documents annotated by a subject is quite complex, and we can discard the assumption we made in Sect. 3.2 by computing similarities between the query document and training documents only, and using these as supporting evidence for their subjects.

Algorithm 1 describes how we rank the subjects for a given new document u. The algorithm returns a ranked list of subjects, where the subjects are sorted according to a summation of (1) the similarity of each subject to the document and (2) the similarity of those of the k most similar documents from the training set which are annotated with the subject. This combination provides us with a robust ranking measure, which combines the direct embedding of the subject in the semantic space where the documents also live and an extra component which lets the k nearest neighbour documents of the new document vouch for the validity of the subject. The idea is that the embedding of each document is more precise than the embedding of the subjects (since that is done based on a combination of many documents), making the similarity computation more trustworthy and the subjects those documents are annotated with reflect more likely to fit the target document.

4 Datasets and Experiments

Our experiments were carried out on a subset of the MEDLINE database (10^6 articles for training and 10^4 for testing), randomly selected from WorldCat.[5] Written in English and published between 1984 and 2012, each article has a title and an abstract, to ensure sufficient textual information for computing the word embeddings. This dataset is interesting as it contains an above-average proportion of technical terms and jargon. Very rare terms carry critical meaning and make the task of word and document embedding particularly challenging.

There are in total 324,619 unique MeSH subjects in the training set, and in average each article is indexed by 16 subjects. These subjects are used in an extremely unbalanced way. On one hand, 222,135 subjects are used to index less than 10 articles, among which 95,218 subjects are used to index only one

[5] http://www.worldcat.org/.

single article. On the other hand, 7 subjects are each used to index more than 100K articles, and the most frequently used subject "Humans" indexes nearly 58% of the whole corpus (see Table 1). Similar statistics of the subject headings in the testing set is shown in Fig. 2.

We first computed embeddings for frequent terms[6] and MeSH subjects using the training set. For each article in the training set, we computed its document embedding based on the terms in its title and abstract. When classifying an unseed article in the testing set, we first computed its document embedding and then either looked for the most similar subjects directly or applied the NPSP method to predict the subjects (here, we took the top 25 closest neighbours, i.e., $k = 25$ in Algorithm 1). The actual MeSH subjects of this article were used as the target subjects for the evaluation.

We also applied fastText [12] which is a state-of-the-art multi-label text classifier to our dataset, and compared our predictions with those from fastText.

5 Evaluation Metrics

The goal of subject prediction is to provide a shortlist of potentially relevant subjects to describe the document at hand. It is important to present a ranked shortlist of candidate subjects and to evaluate the quality of the prediction with an emphasis on the relevance of the top portion of such lists. Therefore, we use rank-based evaluation metrics against the existing human annotations.

For a test document, let $\mathbf{y} \in \{0,1\}^L$ be its *annotated* ground truth label vector and $\hat{\mathbf{y}} \in \mathrm{R}^L$ be the predict score vector. Traditionally, we compute the precision, recall and the normalised Discounted Cumulative Gain (nDCG) up to the top n predictions

$$\mathrm{P@}n = \frac{1}{n} \sum_{l \in r_n(\hat{\mathbf{y}})} \mathbf{y}_l, \tag{4}$$

$$\mathrm{R@}n = \frac{1}{\|\mathbf{y}\|_0} \sum_{l \in r_n(\hat{\mathbf{y}})} \mathbf{y}_l, \tag{5}$$

$$\mathrm{DCG@}n = \sum_{l \in \mathrm{r}_k(\hat{\mathbf{y}})} \frac{\mathbf{y}_l}{\log(l+1)}, \tag{6}$$

$$\mathrm{nDCG@}n = \frac{\mathrm{DCG@}k}{\sum_{l=1}^{\min(k, \|\mathbf{y}\|_0)} \frac{1}{\log(l+1)}} \tag{7}$$

where $r_n(\hat{\mathbf{y}})$ is the set of rank indices of the annotated relevant labels among the top-n portion of the predicted ranked list for a document, and $\|\mathbf{y}\|_0$ counts the number of labels in the annotated ground truth label vector $\mathbf{y}$. P@n, R@n and nDCG@n are calculated for each test document and then averaged over all the documents.

[6] We extracted terms from titles and abstracts and removed those that occurred in less than 10 articles.

However, the complete set of ground truth labels is not obtainable, and infrequently occurring tail labels might be more informative and rewarding, Jain et al. proposed to use propensity scored measures to avoid the popularity bias and in favour rare/novel labels [10]:

$$\text{PSP@}n = \frac{1}{n} \sum_{l \in r_n(\hat{\mathbf{y}})} \frac{\mathbf{y}_l}{p_l}, \tag{8}$$

$$\text{PSR@}n = \frac{1}{1^T \mathbf{y}^*} \sum_{l \in r_n(\hat{\mathbf{y}})} \frac{\mathbf{y}_l}{p_l}, \tag{9}$$

$$\text{PSDCG@}n = \sum_{l \in r_n(\hat{\mathbf{y}})} \frac{\mathbf{y}_l}{p_l \log(l+1)} \tag{10}$$

$$\text{PSnDCG@}n = \frac{\text{PSDCG@}n}{\sum_{l=1}^{n} \frac{1}{\log(l+1)}} \tag{11}$$

where p_l is the propensity score for label l, which could be modelled as a sigmoidal function of the frequency of label l:

$$p_l = P(y_l = 1 | y_l^* = 1) = \frac{1}{1 + Ce^{-A \log(d_l + B)}} \tag{12}$$

where d_l is the number of documents that are indexed with the label l in the training set of size N. A and B depend on the specific dataset and $C = (\log(N) - 1)(B + 1)^A$. Jain et al. suggested $A = 0.55$ and $B = 1.5$. $\mathbf{y}^*$ is the complete (but unobtainable) ground truth label vectors and $1^T \mathbf{y}^*$ can be approximated as the sum of the propensity of the labels in the annotated ground truth, that is, $\sum_{l \in \mathbf{y}} \frac{\mathbf{y}_l}{p_l}$.

6 Evaluation Results

Previous studies [13,26] have shown that Ariadne semantic embedding is highly efficient and competitive with the state-of-the-art word and document embedding methods, such as Word2Vec [19], Doc2Vec [16], GloVe [21], fastText [4] and Sent2Vec [20]. Figure 1 shows the comparison of our two subject prediction methods to the state-of-the-art fastText method in terms of Precision, Recall and nDCG for varying values of n. In these graphs, *Ariadne* represents the straightforward predictions based on subject-document similarities. If we look at the standard precision, recall and nDCG (leftmost graphs), we can see that the quality of the predicted subjects from our similarity-based prediction are comparable with those generated by fastText. The precision of fastText is higher than our Ariadne method for low values of n while it quickly decreases to be worse than ours. Up to top 20 candidates, the recall for both Ariadne and fastText are more or less the same, but our method is able to predict more actual subjects at lower ranks, where the recall outperforms fastText. This is reflected in the propensity-weighted metrics (rightmost graphs), where even the basic Ariadne method outperforms fastText for all but the lowest values of n.

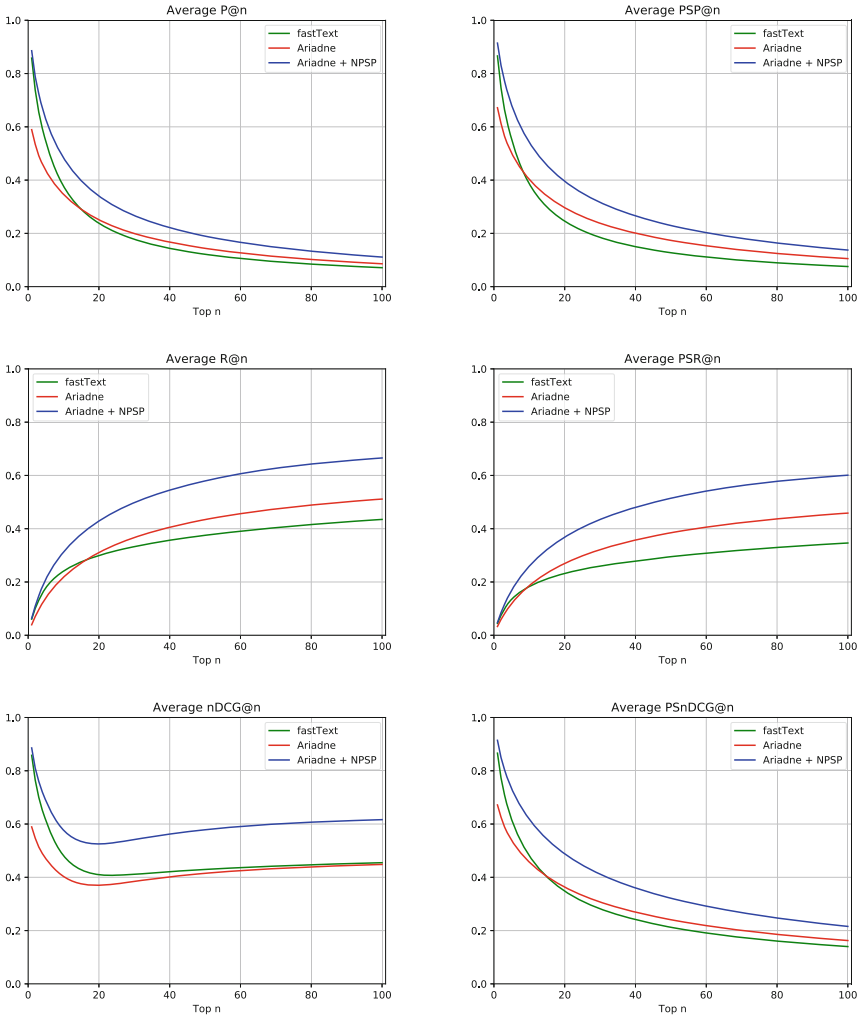

Fig. 1. Precision, Recall, nDCG comparison with and without propensity

The clear winner is the NPSP method. The precision and recall are both consistently higher than the other two methods. At $n = 100$, the recall is nearly 20% higher than the fastText predictions. More correct subjects are predicted at lower ranks, which explains the much slower decrease of precision with increasing rank.

Overall, we can note how well our method performs in the tail of the distribution of the subjects. This is somewhat reflected in the propensity-weighted metrics, but can also be observed directly. Figure 2 shows a histogram of subjects, binned according to the number of documents they are assigned to in the training set. For example, there were 849 unique subjects in the training data

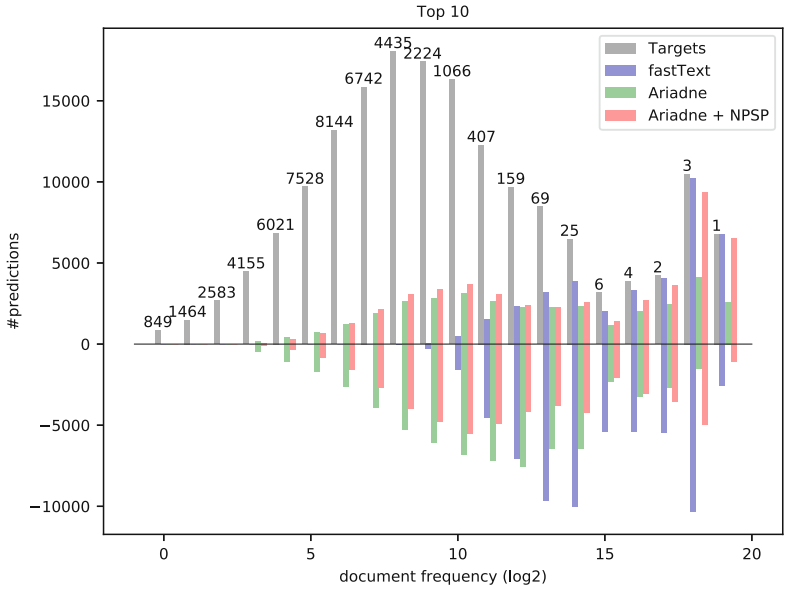

Fig. 2. Predictions vs subjects' document frequencies

that were assigned to exactly one (that is, 2^0) document, 1464 subjects that were assigned to two documents, etc. We then made label predictions for the documents in the test set, and report the true positive predictions (positive bars) and false positive predictions (negative bars) grouped by the assignment frequency of those subjects in the training set. The total number of documents of the test set that were annotated with each category of subjects (the annotated ground truth) are indicated by the grey bars, while the predictions are indicated by the coloured bars: the number of true positive predictions as positive bars and the number of false positive predictions as negative bars, so that the false negatives are indicated by the difference between the grey and the positive coloured bars.

As we can see from this graph, the subjects that are infrequently assigned are harder to predict (since there is less training data to train the model on). It is also noteworthy that commonly assigned subjects are broad terms that cover many documents and very much at the forefront of the annotators' mind, and so we can be confident that these subjects are correctly assigned: their presence may not be very informative but should be trusted. Their absence, on the other hand, is much more meaningful and should also be treated as trustworthy. In other words: for these subjects, false negatives are rare and have little practical relevance, while false positives are much more problematic. Conversely, when they are present, very rare subjects are much more informative and should be treated as both important and trustworthy, but predicting them is much harder: the true positives of rare subjects are very valuable. At the same time, when they are absent, we should allow for the fact that these subjects are often overlooked

by the annotators even when they are relevant: the false negatives for these subjects should be considered with suspicion. With this in mind, we can see how much better our method performs when compared to FastText: it has far fewer false positives for the common subjects, far more true positives for the uncommon subjects, and although it does have more false positives for uncommon subjects, these may well still be relevant subjects.

Notice how, unlike in retrieval where recall is of limited interest in practice because only the few most relevant documents would be actually looked at, in the case of XMTC recall is the more important metric, especially for the more obscure subjects for which lack of training data makes it hard to build comprehensive models. A high recall is important as it would greatly reduce the search space and also provides opportunities for cataloguers to find more suitable subjects which they may otherwise not have considered.

Also notice how, by the same token, higher values of n are also important. Highly ranked subjects tend to be very frequent terms: at best, they are relatively uninformative, at worst they are incorrect. Lower-ranked subjects which tend to be less frequently used terms, by contrast, are at best informative, at worst incorrect and more likely relevant but unannotated.

A Closer Look. To illustrate these points, Table 1 lists the 23 actual MeSH subjects for an arbitrary article, titled "Cumulative probability of neodymium: YAG laser posterior capsulotomy after phacoemulsification.", which is about laser-based eye surgery.[7] The MeSH terms that reflect the major subjects of this article, as annotated by the indexers, are marked with an asterisk (*). The 25 most relevant MeSH subjects predicted by our two methods and fastText are also listed.

It is not surprising that subjects such as "Humans" and "Female" are predicted first by fastText, because they are the most frequent in the dataset. In fact, many of the subjects predicted by fastText are very common (see their document counts in Table 1), which leads to higher precision and recall at the low values of n. However, as argued above, a P@5 of 100% is actually uninformative about the real topic of this article, therefore, less valuable. FastText has trouble finding subjects which describe the articles more precisely (also illustrated in Fig. 2).

The raw subject-document similarity is able to rank infrequent actual subjects such as "Phacoemulsification," "Lens Capsule, Crystalline/Surgery" high in the list. We believe these infrequent subjects are more informative and valuable in terms of subject indexing. Common subjects such as "Female" and "Male" tend to be ranked lower though. The NPSP method effectively boosts these common subjects to the front (such as "Human" is recovered and ranked at the top), while the correct specific subjects still stay relatively high in the list. The previously-missed subjects such as "Acrylic Resins" and "Silicone Elastomers" get into the top 25. Unfortunately the highly relevant but extremely infrequent subject "Capsulorhexis" drops out of the top 25 list now.

[7] https://www.ncbi.nlm.nih.gov/pubmed/14670424.

Table 1. An example of actual MeSH subjects versus the top 25 predicted ones by our two methods and fastText, where the ones in bold match the actual subjects. The raw document counts (d_t) of the MeSH subjects in the training dataset are also given.

Actual MeSH subjects (alphabetical order)	d_t	Ariadne	d_t	Ariadne + NPSP	d_t	fastText	d_t
Acrylic Resins	920	Lenses, Intraocular	434	**Humans**	579975	**Humans**	579975
Aged	118655	**Lens Implantation, Intraocular**	317	**Male**	336647	**Female**	328885
Aged, 80 and over	40642	**Phacoemulsification**	225	**Female**	328885	**Middle Aged**	168714
Capsulorhexis	24	**Visual Acuity**	2026	**Aged**	118655	**Male**	336647
Female	328885	**Lens Capsule, Crystalline/Surgery**	79	Lenses, Intraocular	434	**Risk Factors**	34538
Humans	579975	**Lens Capsule, Crystalline/Pathology**	65	**Lens Implantation, intraocular**	317	Adult	194200
Laser therapy*	847	Visual Acuity/Physiology	949	**Visual Acuity**	2026	**Aged**	118655
Lens Capsule, Crystalline/Pathology	65	Lens Implantation, Intraocular/Methods	90	**Phacoemulsification**	225	**Retrospective Studies**	32642
Lens Capsule, Crystalline/Surgery*	79	Cataract Extraction/Methods	136	**Middle Aged**	168714	Follow Up Studies	27911
Lens Implantation, Intraocular	317	Phacoemulsification/Methods	106	Prospective Studies	25714	**Aged 80 and over**	40642
Male	336647	**Retrospective Studies**	32642	**Lens Capsule, Crystalline/Surgery**	79	Incidence	11468
Middle Aged	168714	**Aged**	118655	Follow Up Studies	27911	Adolescent	75361
Phacoemulsification*	225	Prospective Studies	25714	**Acrylic Resins**	920	Young Adult	27991
Polymethyl Methacrylate	392	**Aged 80 and Over**	40642	**Lens Capsule, Crystalline/Pathology**	65	Prospective Studies	25714
Postoperative Complications/Pathology	351	**Middle Aged**	168714	Postoperative Complications	3961	Time Factors	50339
Postoperative Complications/Surgery*	823	**Male**	336647	**Aged 80 and Over**	40642	Logistic Models	7258
Probability	2914	Cataract Extraction	350	Cataract Etiology	218	Case Control Studies	13306
Retrospective Studies	32642	Lenses, Intraocular/Adverse Effects	62	Adult	194200	Cohort Studies	12275
Risk Factors	34538	Adolescent	75361	**Retrospective Studies**	32642	Treatment Outcome	40496
Sex Factors	10203	**Female**	328885	Visual Acuity/Physiology	949	Child	53738
Silicone Elastomers	294	Intraocular Pressure	712	Treatment Outcome	40496	Survival Rate	7975
Survival Analysis	7046	**Capsulorhexis**	24	Cataract Extraction	350	Prognosis	17160
Visual Acuity	2026	Pseudophakia Physiopathology	56	Lens Implantation, Intraocular/Adverse Effects	45	Risk Assessment	9271
		Adult	194200	**Silicone Elastomers**	294	Reoperation	3355
		Risk Factors	34538	Prosthesis Design	2096	Proportional Hazards Models	3403

We realise that this evaluation has its limitations. As shown in Table 1, highly related MeSH subjects such as "Lenses Intraocular" and "Phacoemulsification Methods" are predicted as good candidates for this article, both of which are reasonable and potentially useful, but since they are not the subject headings that the professional taxonomists have chosen, their value cannot be easily assessed. As discussed previously, such false negatives should be treated critically.

That being said, we believe our predictions are useful in practice and can be presented to cataloger as candidate subjects to choose from. Consider, for example, "Cataract Extraction," "Intraocular Pressure," etc. in our example. Again, we need to get subject specialists involved to conduct such qualitative evaluations.

7 Conclusion

In this paper, we have shown that a similarity-based subject prediction based on a suitable semantic space that allows for the embedding of both subjects and documents is very competitive with the state-of-the-art subject-prediction method based on a classifier. We have described such an embedding and have shown how effective this specific semantic space really is, both with quantitative and with qualitative evaluations. In addition, we have shown how our embedding-based method is particularly effective at correctly predicting very specialised subjects, which are associated with few documents in the training set and are more problematic for a classifier, as is reflected in our method's much slower decrease of precision with increasing rank. In addition, we proposed a novel, non-parametric, similarity-based method with the documents instead of the subjects. We have shown that this method substantially improves the quality of the predictions, both in comparison to the state-of-the-art and to the bare similarity-based method.

References

1. Achlioptas, D.: Database-friendly random projections: Johnson-Lindenstrauss with binary coins. J. Comput. Syst. Sci. **66**(4), 671–687 (2003). https://doi.org/10.1016/S0022-0000(03)00025-4
2. Arash, J., Abdulhussain, E.M.: Classification of scientific publications according to library controlled vocabularies: a new concept matching-based approach. Libr. Hi Tech **31**, 725–747 (2013). https://doi.org/10.1108/LHT-03-2013-0030
3. Bhatia, K., Jain, H., Kar, P., Varma, M., Jain, P.: Sparse local embeddings for extreme multi-label classification. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28, pp. 730–738. Curran Associates, Inc. (2015)
4. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017). https://doi.org/10.1162/tacl_a_00051
5. Foster, D.V., Grassberger, P.: Lower bounds on mutual information. Phys. Rev. E **83**, 010101 (2011). https://doi.org/10.1103/PhysRevE.83.010101

6. Frommholz, I., Abbasi, M.K.: Automated text categorization and clustering. In: Golub, K. (ed.) Subject Access to Information: An Interdisciplinary Approach: An Interdisciplinary Approach, pp. 117–131. ABC-CLIO (2014)
7. Godby, J., Reighart, R.: The wordsmith indexing system. J. Libr. Adm. **34**(3–4), 375–385 (2001). https://doi.org/10.1300/J111v34n03_18
8. Godby, J., Smith, D.: Scorpion. https://www.oclc.org/research/activities/scorpion.html. Accessed Apr 2019
9. Golub, K.: Automatic subject indexing of text. In: ISKO Encyclopedia of Knowledge Organization. http://www.isko.org/cyclo/automatic. Version 07 Mar 2019
10. Jain, H., Prabhu, Y., Varma, M.: Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016, pp. 935–944. ACM, New York (2016). https://doi.org/10.1145/2939672.2939756
11. Johnson, W., Lindenstrauss, J.: Extensions of Lipschitz mappings into a Hilbert space. Contemp. Math. **26**, 189–206 (1984)
12. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759 (2016)
13. Koopman, R., Wang, S., Englebienne, G.: Fast and discriminative semantic embedding. In: Proceedings of the 13th International Conference on Computational Semantics - Long Papers, Gothenburg, Sweden, 23–27 May 2019, pp. 235–246. ACL (2019)
14. Koopman, R., Wang, S., Scharnhorst, A.: Contextualization of topics: browsing through the universe of bibliographic information. Scientometrics **111**(2), 1119–1139 (2017). https://doi.org/10.1007/s11192-017-2303-4
15. Koopman, R., Wang, S., Scharnhorst, A., Englebienne, G.: Ariadne's thread: interactive navigation in a world of networked information. In: Proceedings of the ACM Conference Extended Abstracts on Human Factors in Computing Systems, pp. 1833–1838 (2015)
16. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: International Conference on Machine Learning - ICML 2014, vol. 32, pp. 1188–1196, March 2014. https://doi.org/10.1145/2740908.2742760
17. Liu, J., Chang, W.C., Wu, Y., Yang, Y.: Deep learning for extreme multi-label text classification. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2017, pp. 115–124. ACM, New York (2017). https://doi.org/10.1145/3077136.3080834
18. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, NIPS 2013, pp. 3111–3119. Curran Associates Inc., USA (2013)
20. Pagliardini, M., Gupta, P., Jaggi, M.: Unsupervised learning of sentence embeddings using compositional n-gram features. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pp. 528–540. Association for Computational Linguistics (2018). https://doi.org/10.18653/v1/N18-1049
21. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014). https://doi.org/10.3115/v1/D14-1162

22. Prabhu, Y., Kag, A., Harsola, S., Agrawal, R., Varma, M.: Parabel: partitioned label trees for extreme classification with application to dynamic search advertising. In: Proceedings of the International World Wide Web Conference, April 2018
23. Prabhu, Y., Varma, M.: FastXML: a fast, accurate and stable tree-classifier for extreme multi-label learning. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2014, pp. 263–272. ACM, New York (2014). https://doi.org/10.1145/2623330.2623651
24. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002). https://doi.org/10.1145/505282.505283
25. Tagami, Y.: AnnexML: approximate nearest neighbor search for extreme multi-label classification. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2017, pp. 455–464. ACM, New York (2017). https://doi.org/10.1145/3097983.3097987
26. Wang, S., Koopman, R.: Semantic embedding for information retrieval. In: Proceedings of the Fifth Workshop on Bibliometric-enhanced Information Retrieval, pp. 122–132 (2017)
27. Weston, J., Bengio, S., Usunier, N.: WSABIE: Scaling up to large vocabulary image annotation. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Three, IJCAI 2011, pp. 2764–2770. AAAI Press (2011). https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-460