Elsevier Journal Finder: Recommending Journals for your Paper

Ning Kang
Research Management, Elsevier
Radarweg 29, 1043 NX Amsterdam, The Netherlands
n.kang@elsevier.com

Marius Doornenbal
Research Management, Elsevier
Radarweg 29, 1043 NX Amsterdam, The Netherlands
+31 20 4583255
m.doornenbal@elsevier.com

Bob Schijvenaars
Research Management, Elsevier
Radarweg 29, 1043 NX Amsterdam, The Netherlands
b.schijvenaars@elsevier.com

ABSTRACT
Rejection is the norm in academic publishing. One of the main reasons for rejections is that the topics of the submitted papers are not relevant to the scope of the journal, even when the papers themselves are excellent. Submission to a journal that fits well with the publication may avoid this issue. A system that is able to suggest journals that have published similar articles to the submitted papers may help authors choose where to submit. The Elsevier journal finder, a freely available online service, is one of the most comprehensive journal recommender systems, covering all scientific domains and more than 2,900 per-reviewed Elsevier journals. The system uses natural language processing for feature generation, and Okapi BM25 matching for the recommendation algorithm. The procedure is to paste text, such as an abstract, and get a list of recommend journals and relevant metadata. The website URL is http://journalfinder.elsevier.com.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Clustering; I.2.7 [**Natural Language Processing**]: Text analysis; I.5.3 [**Clustering**]: Similarity measures

General Terms
Algorithms, Measurement, Experimentation, OKAPI BM25.

Keywords: Natural language processing, Recommender system, Noun phrase, TF-IDF, Okapi BM25

1. INTRODUCTION
Finding the right journal to submit a paper is one of the most important steps during the process of paper publishing. For most authors, this job is difficult because many journals have a very wide diversity of topics, and many articles involve several academic disciplines or professional specializations.

RecSys '15, September 16 - 20, 2015, Vienna, Austria

ACM 978-1-4503-3692-5/15/09…$15.00
DOI: http://dx.doi.org/10.1145/2792838.2799663

According to the records from the Scopus database [2], from 1992 to 2002, 12 million peer-reviewed papers have been published. This number has doubled between 2003 and 2013. With the rapid growth of new journals and papers each year, the task to select a correct journal to submit a paper becomes more and more difficult.

In this study, we present the Elsevier journal finder, a comprehensive journal recommender system that covers all major scientific domains and more than 2,900 peer-reviewed Elsevier journals, to help authors easily find relevant journals for their paper.

2. BACKGROUND
Recommender systems have become quite common in recent years, and are applied in a variety of applications [10][1].

In the field of journal recommendation, there are already some systems that search for similar articles [7]. For example, PubMed [6] offers the function to search similar records from Medline records, but only existing Medline records can be used as queries. eTBLAST [3] accepts full abstracts search for journal recommendation, and Jane [12] also provides similar functions. However, these systems only cover the biomedical domain. Some cross domain tools such as Mendeley [9] search for similar articles based on articles that have already been published, but do not recommend journals.

3. METHOD
3.1 Source of journals and papers
The Scopus database [2] is used as the source of journals and papers for the Elsevier journal finder. The Scopus database is the largest abstract and citation database of peer-reviewed literature from scientific journals, books and conference proceedings. It contains more than 55 million records and 5000 publishers, and covers all major scientific domains: Agriculture, Chemistry, Economics, Geo-Sciences, Humanities and Arts, Life and Health Sciences, Materials Science and Engineering, Mathematics, Physics, and Social Sciences.

For our system, we only use the papers that are published after 2008, for the reason that the scope of the journals may change over time, and newer papers reflect the current scope of journals more accurately. Also, the system's response time improves when it includes less sample papers. We also filter out all papers from non-Elsevier journals because our system only recommends Elsevier journals. The remaining 1.98 million paper records from the Scopus database are used as the sample papers set.

3.2 Noun phrase annotations

The Elsevier journal finder uses noun phrases [5] as features for the paper matching and journal ranking algorithm. The noun phrases are annotated and normalized by the Elsevier Fingerprint Engine [14]. The Elsevier Fingerprint Engine (EFE) applies a variety of Natural Language Processing (NLP) techniques to mine the input text and generates all relevant annotations, including sentence boundaries, tokenization, part-of-speech tags, and phrase chunking. Noun phrases are extracted based on a relatively simple pattern of part-of-speech (POS) tag sequences. We employed a simple noun phrase syntax, sketched in Backus-Naur-form:

<NP> ::= <Pre> <NN> | <NN> | <NP> "in" <NP>
<Mod> ::= "jj" | "nn" | "nn$" | "np"
<Pre> ::= <Mod> | <Pre> <Mod>
<NN> ::= "nn" | "np" | "nns"

In this noun phrase grammar, POS tags are used as terminals (*jj* is 'adjective', *nn* is 'noun', *np* is 'proper noun', *nn$* is 'possessive noun', *nns* is 'plural noun', and *in* is 'preposition'). To improve feature generation however, we made the algorithm to select sub-phrases of full noun phrases, in order to avoid a very sparse vector space containing only very specific noun phrases. The feature set consists of noun phrases in normalized form, meaning that plural forms translate to singular variants, and spelling variations, e.g. British to American, are normalized away.

We preprocess the 1.98 million sample papers from the Scopus database to generate normalized noun phrases as weighted features (vector-dimensions) for each paper published in the target journals. For each query input text submitted, we use the EFE to generate normalized noun phrases as a query vector for the paper matching algorithm. We remove all noun phrases that occur only once and the top 300 noun phrases that occur most frequently (e.g., study, method, data, analyse, paper, conclusion, model, system, etc. ….). These noun phrases are too commonly used to contribute to the ranking algorithm. By testing the accuracy, these optimization parameters give the highest accuracy.

3.3 Ranking algorithm

The journal recommendation ranking algorithm is divided into two parts. The first part is matching the submitted query to existing papers in the database. For this purpose, we use the Okapi BM25 algorithm [11]. The Okapi BM25 algorithm is widely used in the domain of information retrieval. It ranks matching documents according to their relevance to a given search query. Normally, the input is a bag-of-words, and the output is a set of documents with scores and ranks based on the query words appearing in each document, regardless of the inter-relationship between the query terms within a document (e.g., their relative proximity).

In our case, instead of using the whole text as input for the retrieval function, we use the normalized noun phrase annotations of the input text as input for the algorithm. By our estimate, this feature selection may account for part of the improved accuracy of our system relative to other recommender systems.

The Okapi BM25 can be described as below: Given an input text Q, containing noun phrases q_1, …, q_n, the BM25 score of a paper D is:

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{f(q_i,D) \cdot (k_1+1)}{f(q_i,D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})}$$

where $f(q_i,D)$ is q_i's frequency in paper D, |D| is the length of the paper D in noun phrases, and *avgdl* is the average paper length in noun phrases in the sample paper set. The parameters k_1 and b allow for adapting the algorithm to different use cases. In our case, we used 1.5 for k_1, 0.6 for b (experimentally), and measured 68 for *avgdl* as the average document length.

IDF q_i is the IDF (inverse document frequency) [12] weight of the noun phrase q_i. It is usually computed as:

$$IDF(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}$$

where N is the total number of papers in the sample paper set, and $n(q_i)$ is the number of papers containing q_i.

After the first step, we get a ranked paper list with a BM25 score for each paper that has already been published in a journal [5]. The top paper in this list is the paper most similar to the input text.

The second part of the journal recommendation ranking algorithm translates the scores for individual papers to scores for journals. This step is divided into the following sub-steps:

1. Keep the top 1 million papers with the highest BM25 score from the ranked paper list, and find the journal and the journal's scientific domains of the journal that each paper belongs to. Given the size of the data set we have no reason to expect that articles outside the top 1 million will contribute to the aggregated score per journal (see below, 3);

2. If the end-user has already selected a domain, then remove all documents that do not belong to this domain. This step is skipped if the end-user did not select a domain for the input text. (See the section of system overview for more information about the input from end-users);

3. Compute an average BM25 score per journal by averaging the scores of all papers published in the same journal:

$$score(J,Q) = \frac{\sum_{i=1}^{N_J} score(D_i,Q)}{N_J}$$

Where N_J is the number of papers published in journal J, $score(D_i,Q)$ is the BM25 score of paper D_i in journal J. We take the average to correct for journal size.

3.4 System overview

In Figure 1, we show the system overview of the Elsevier journal finder. When an end-user inputs an abstract, the EFE first generates the normalized noun phrases, which are then used by the paper matching algorithm to find the related papers from the Scopus database, and then these papers are used by the journal ranking algorithm to get the recommended journals list.

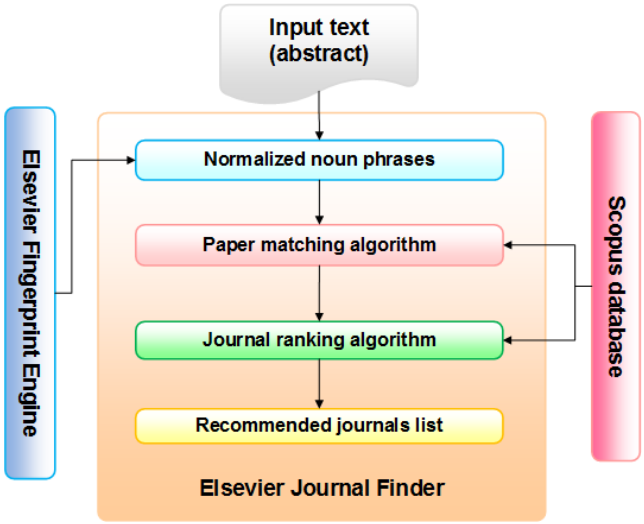

Figure 1: System overview of the Elsevier journal finder

From Figure 2, we can see the input interface of the Elsevier journal finder. The end users can simply input the paper title and abstract, or even just a few keywords, and then select one or more scientific domains that the input text belongs to (this step can also be skipped if the end-user is not sure about which domain(s) the input text belongs to).

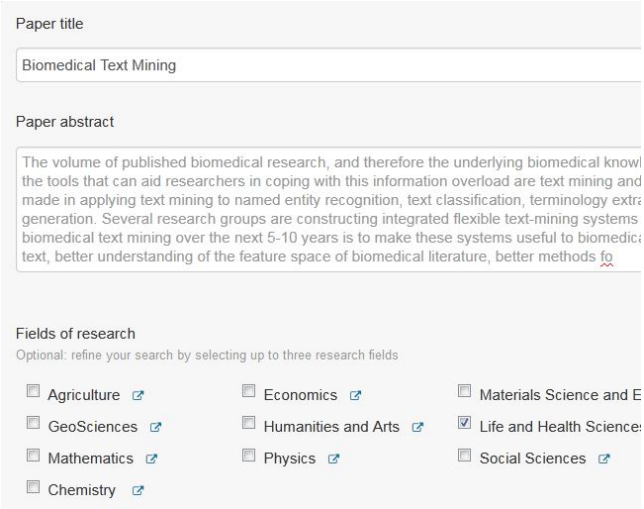

Figure 2: Screenshot of the input interface

Figure 3 shows the list of recommended Elsevier journals, together with some important metadata of the recommended journals, such as matching score, impact factor, open access, editorial times, acceptance rate, and production times. By clicking each journal title, user can also see the scope and more information about this journal. This information can help the authors to decide to which journal to submit their papers, and may reduce the probability of rejection.

Figure 3: Screenshot of the recommended journal list.

4. EXPERIMENTS AND RESULTS

To evaluate the accuracy of our system, we applied a strategy similar to leave-one-out cross-validation: we randomly selected 10 to 100 (depending on the number of papers published in each journal) already published papers from each Elsevier journal as the input documents, and removed these input documents from the source database. If the top three or top ten recommended journals contained the journal in which the input paper was published, then this is counted as a correct recommendation, otherwise it is counted as a false recommendation.

Table 1 shows the performance of each optimization. By changing the features from concepts to noun phrases, the performance is improved by more than 10%. By optimizing the noun phrases (normalize and filter the noun phrases) and the algorithm (tuning the parameters of the paper matching algorithm and the journal ranking algorithm), the performance is further improved by another 5%. The best performance is 42.6% for the top 3, and 64.6% for the top 10.

Table 1. The performance of each optimization step

Features	Performance	
	Top 3	Top 10
Concepts	26.7%	47.8%
Noun phrases	36.3%	59.8%
Optimized noun phrases	38.1%	61.2%
Optimized noun phrases and algorithm	42.6%	64.6%

5. DISCUSSION

We use normalized noun phrases as the features for our ranking algorithm, and do not use advanced annotations such as concepts as thesaurus-defined entities. Although the EFE can generate high quality concept annotations, these are not suitable for this use case. Using concept annotations results in a sparse feature set, particularly as a comprehensive, good-coverage thesaurus spanning all disciplines is not readily available. Furthermore, considering the diverse nature of the data (texts from multiple science domains), using noun phrase annotations as features is better than using concept annotations. For an extensive discussion of the best feature sets, cf. [4].

The ranking algorithm only works well if there are enough sample papers (at least more than 100) in each journal. However, for

some new journals, there are not enough published papers. To solve this problem, we asked the editors to select some papers from other journals that are relevant to the scope of the new journals, and then used these selected papers as the sample papers for the ranking algorithm.

The performance of the Elsevier journal finder is better than Jane (42% for top 3 and 58% for top 10) [13] and eTBLAST (35% for top 3 and 50% for top 10) [3] that use the same evaluation method of leave-one-out cross-validation. Besides that, the Elsevier journal finder is the only system that covers all major scientific domains (including the biomedical domain), whereas the other two systems only cover the biomedical domain.

However, the performance figures of these systems are based on different test document sets. They could be changed if we use the same test document set. This is difficult to do because the leave-one-out cross-validation method needs to change the training document set, which is impossible if we do not have the source code of Jane and eTBLAST.

Theoretically, the Elsevier journal finder can recommend any journal in the Scopus database. Although the recommended journals are limited to Elsevier journals only, the system can always recommend highly relevant journals to the authors for their papers, since Elsevier has more than 2900 peer-reviewed journals that cover almost all major scientific domains.

6. REFERENCES

[1] Bobadilla, J. et al. 2013. Recommender systems survey. *Knowledge-Based Systems*. 46, (2013), 109–132. DOI= http://dx.doi.org/10.1016/j.knosys.2013.03.012

[2] Burnham, J.F. 2006. Scopus database: a review. *Biomedical digital libraries*. 3, (2006), 1. DOI= http://dx.doi.org/10.1186%2F1742-5581-3-1

[3] Errami, M. et al. 2007. ETBLAST: A web server to identify expert reviewers, appropriate journals and similar publications. *Nucleic Acids Research 35(S2)*.

[4] Jimeno Yepes, A.J., Plaza, L., Carrillo-de-Albornoz, J., Mork, J.G., Aronson, A.R. Feature engineering for MEDLINE citation categorization with MeSH. *BMC Bioinformatics, 16 (1), 113* (2015), DOI=http://dx.doi.org/10.1186/s12859-015-0539-7

[5] Kang, N. et al. 2011. Comparing and combining chunkers of biomedical text. *J Biomed Inform*. 44, 2 (2011), 354–60. DOI=http://dx.doi.org/10.1016/j.jbi.2010.10.005

[6] Kantrowitz, M. et al. 2000. Stemming and its effects on TFIDF Ranking. *Proceedings of the 33st annual international ACM SIGIR conference on Research and development in information retrieval, (ACM SIGIR 2000)*, 357–359. DOI=http://dx.doi.org/10.1145/345508.345650

[7] Lu, Z. 2011. PubMed and beyond: A survey of web tools for searching biomedical literature. *Database*. (2011). DOI=http://dx.doi.org/10.1093/database/baq036

[8] McEntyre, J. and Lipman, D. 2001. PubMed: Bridging the information gap. *Canadian Medical Association Journal. 164, 1317-1319*.

[9] Reiswig, J. 2010. Mendeley. *Journal of the Medical Library Association*. 98, (2010), 193–194. DOI= http://dx.doi.org/10.3163%2F1536-5050.98.2.021

[10] Ricci, F. et al. 2011. Introduction to Recommender Systems Handbook. *Recommender Systems Handbook* 1-35

[11] Robertson, S.E. 1990. On term selection for query expansion. *Journal of Documentation*. *46(4)*, 359-364 DOI= http://dx.doi.org/10.1108/eb026866

[12] Roelleke, T. and Wang, J. 2008. TF-IDF uncovered: a study of theories and probabilities. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (2008), 435–442.

[13] Schuemie, M.J. and Kors, J.A. 2008. Jane: Suggesting journals, finding experts. *Bioinformatics*. 24, (2008), 727–728. DOI=http://dx.doi.org/10.1093/bioinformatics/btn006

[13] Vestdam, T.V. et al. 2014. Black magic meta data - Get a glimpse behind the scene. *Procedia Computer Science* (2014), 239–244. DOI=http://dx.doi.org/10.1016/j.procs.2014.06.038