What Were People Searching For? A Query Log Analysis of An Academic Search Engine

Shaurya Rohatgi
Pennsylvania State University
University Park, PA, USA
szr207@psu.edu

Jian Wu
Old Dominion University
Norfolk, VA, USA
jwu@cs.odu.edu

C. Lee Giles
Pennsylvania State University
University Park, PA, USA
clg20@psu.edu

Abstract

Academic search engines have served the research community for years, yet there is little work done on understanding the taxonomy of query semantics. In this work, we present our findings of analyzing the query log of an academic search engine in the past four years. We study the distribution of query intents to understand the information requested by users. We classify query strings by topics using shallow and latent features captured using a customized word embedding model. To this end, we create a dataset that has scientific keywords and titles labeled with fields of study. This dataset is later used to train a classifier that discriminates query logs by topics. Our work will help to train better learning-based ranking functions that improve user experiences for an academic search engine. In addition, we anonymize our 78 million query logs and make them available for the research community for further exploration.

CCS Concepts: • **Information systems → Query log analysis**.

Keywords: information retrieval, query log analysis, query processing

ACM Reference Format:
Shaurya Rohatgi, Jian Wu, and C. Lee Giles. 2020. What Were People Searching For? A Query Log Analysis of An Academic Search Engine . In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3395027.3419597

1 Introduction

Academic search engines with full text search capabilities like Google Scholar, Semantic Scholar, and Microsoft Academic Search hold special positions in academia. They provide the researchers quick and easy access to the plethora of research articles on the web. In spite of their usefulness, there is limited work in the area of academic search query understanding. The increasing online traffic for academic search makes it necessary to study the information needs in academic search, which is important in guiding the development of ranking models. Query understanding models have been developed to probe users' information needs, including tasks such as query classification, intent understanding, segmentation, suggestion, and rewriting [1]. Query intent understanding for academic search engines aims at understanding how users search for a particular page or an article, such as by title, keyword, or author names. One goal of query topic classification is to assign a topic to a search query. In the context of an academic search engine, these topics are usually research-related.

Existing work has analyzed search engine logs to understand academic queries [5]. Previous work has classified queries into navigational and informational categories [4, 7]. This high-level categorization may not be sufficient to understand users' information need. This motivates us to examine the semantics of queries in a fine granularity by classifying them by intent and topics. Previous work has demonstrated that if a topic can be identified for a search query, the search results are significantly improved [1]. Research Subject classification using scientific abstracts has been well studied in a recent work [6]. This task is challenging due to vocabulary overlap between similar domains. However, topic classification on short text such as search queries can also be challenging due to the limited and ambiguous information.

In this work we focus on topic classification and intent understanding of search logs for an academic search engine, using CiteSeerX as a case study [3]. We introduce the concept of academic query intent understanding and research topic classification. We release a dataset containing millions of anonymized log records across 4 years. To the best of our knowledge, this is the first work that studies academic query logs at such a scale.

Conference'17, July 2017, Washington, DC, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8000-3/20/09…$15.00
https://doi.org/10.1145/3395027.3419597

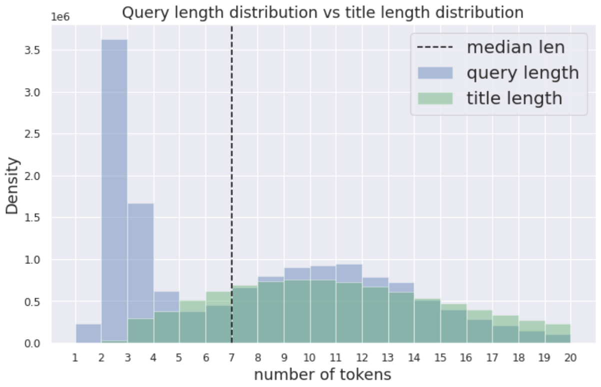

Figure 1. The length of queries peaks at length=2, which are bigrams and mostly author mentions. The title distribution from MAG is very close to our query log distribution.

2 Data Acquisition and Preprocessing

CiteSeerX is a digital library search engine, currently indexing over 10 million open-access scientific documents [9]. The search engine uses Apache Solr as the indexer and provides full-text search. We analyze queries generated by the search engine between January 2017 and January 2021. The total number of raw search queries for this period is 78,124,884. The original query logs are generated by Tomcat (version 6.x) across three web servers. We select logs recording users' attempts to query using search interfaces. There are several possible interfaces the users may input a text search. The search box for CiteSeerX[1] is available on the homepage, search engine result page (SERP) and a paper's summary page, or the advanced search page. We merge queries from all interfaces on all web servers.

For each search query log, we extract the following information. (1) Raw query string. (2) IP address of the request. (3) User-agent e.g., "Baiduspider 2.0" and "Chrome 45.0.2454". (4) User-agent Type e.g., "Spider", "PC", and, "Other". To anonymize the logs, we hash all IP addresses, ensuring the same hash for the same IP, which allows us to group requests by IP for further analysis.

We further remove logs matching the following patterns.

1. Duplicate raw query logs (14,759,852 unique queries).
2. Queries containing obscene words in a controlled list[2].
3. Queries with characters other than alphanumeric and special characters, e.g., Chinese characters.
4. Queries containing more than 40 tokens.

Figure 1 shows the query length distribution. The peak appearing at bi-gram and tri-gram queries are predominantly contributed by keywords and author names.

Table 1. Tasks and their respective datasets used in this paper. Abstracts from SciDocs were used to extract keywords which helped us augment data for each task.

Task	Dataset	Remarks
Title classification	SciDocs	19k samples for each class
Query Topic classification	SciDocs	8k noun phrases and titles for each research topic
Word representation learning (Scientific fastText)	MAG	500k samples for each research topic

3 Methods

3.1 Query Intent Classification

Identifying the intents of user queries is important for improving the ranking quality of search engines. For example, if a user searches a paper title, incorporating an exact title-match would greatly improve the ranking. Similarly, if the users are searching for author names, the ranker should include author-name disambiguation and recognition.

Academic search has 3 major intents. Users sometimes mix basic intents and query author names with keywords. We consider these cases as author searches.

1. Concept/keyword search: Research concept e.g., "relational reinforcement learning".
2. Title search: Exact title of a paper, e.g., "Acknowledgement Entity Recognition in CORD-19 Papers".
3. Author search: Searching a specific author, e.g., "Chris Manning".

We use a supervised model to classify queries into three intents above. To build the ground truth, we obtain 48,472 titles from SciDocs [2], a benchmark dataset created for research subject classification, which includes an equal number of samples from diverse research subjects. We generate keyword samples by automatically extracting noun phrases from abstracts of papers in this dataset using a grammar-based chunking method. We classify author queries using a pre-trained named entity recognition (NER) model.

3.2 Query Topic Classification

Our goal is to understand what research topics have been searched and the distributions. We adopt a supervised machine learning model to classify the queries into research subjects. In SciDocs, each abstract is associated with a research subject label. We extract noun phrases from the abstracts and label each phrase with the research subject of the abstracts where they are extracted.

4 Experiments and Results

4.1 Query Intent Classification

We use a 2-step filtering method for intent identification. In the first step, we identify the queries that contain author

[1] http://citeseerx.ist.psu.edu
[2] https://www.cs.cmu.edu/~biglou/resources/

Table 2. Intent distribution and their samples form logs

Intent (%)	Log Samples
Author Search (13.3%)	Michael J. Schöning
	Mohammed Petiwala
	Youngkil Choi
Title Search (37%)	Automatic induction of FrameNet lexical units
	next-to-leading order perturbative qcd correction
	N-body spacetime constraints
Keyword Search (49.7%)	rapid event
	capture kinetics scalable compiler framework
	4-bit reversible circuit

mentions and filter them out. The second step uses a classifier based on basic features [8] to identify if a string is a title or a keyword.

Step 1: Author Mention Identification To identify author mentions in queries, we use the implementation of named-entity resolution by spaCy[3], which features a sophisticated word embedding strategy using subword features and "Bloom" embeddings, a deep convolutional neural network with residual connections, and a novel transition-based approach to named entity parsing. A preliminary manual examination on 100 random queries indicates the accuracy of the NER on our corpus is 93%. We exclude the queries which have author mentions in them. The remaining queries are passed to the title classifier.

Step 2: Title Classifier The classifier extracts seven features from each query string, including document length, stopword count, punctuation count, number of words starting with uppercase letters, the minimum TF-IDF, the maximum TF-IDF, and the median TF-IDF of words. To calculate the TF-IDF of the queries and the train/test data, we sampled 500k abstracts and titles from each of the 19 level-1 research topics defined in the Microsoft Academic Graph 1 so we have coverage of the scientific vocabulary across the research topics.

We use the SciDocs dataset to build the training data. We extract noun phrases from abstracts in this dataset and label them as keywords. The paper titles from this dataset are directly adopted as query samples labeled as "title". We randomly down-sample noun-phrases extracted to match the size of samples labeled as "title". Finally, we obtained a dataset containing a total of 96,944 titles and keywords. We use K-fold validation while training our classifiers. We use an 8:1:1 split for our train, validation, and test data. Hyperparameters include 100 trees in the forest and the Gini index to measure the quality of the split. Extracting basic features [8] and training a random forest classifier on it yields F1=0.96.

Query Intent Distribution Using this classifier we classified 14 million unique CiteSeerX queries obtained in Section 2. The distribution of title, keyword, and author categories is shown in Table 2. Keywords dominate the search queries, which is consistent with the observation in Figure 1. It is interesting to see that author search constitutes about 14% of the search queries which indicates the importance of author name recognition and disambiguation in an academic search engine. Exact title matches take about 37% of all queries, which can be identified as a "navigational" query [7], where the users know what they want. Our results indicate that exact title search and author name search should be considered when designing a ranking function for an academic search engine. Table 2 shows some examples from the actual logs.

4.2 Query Topic Classification

Unlike query intent classification, in which the syntactical and lexical information is sufficient, classifying queries by topics requires the model to understand the semantics represented by latent features. We created a dataset consisting of 150k samples from the SciDocs dataset. We extracted noun phrases from each abstract and labeled them with research topics. Then we removed noun phrases that occurred more than once and in abstracts across multiple research topics, e.g., "state of the art", "the elements", and "engineers". We retained noun phrases that occurred in only one research topic, e.g., "polynomial sequences" from mathematics, and "the medieval style" from art. The extracted dataset includes titles and keywords from 19 research topics.

Training Scientific FastText We reused the 500k abstracts and titles extracted for query intent classification to train a language model called SciFastText. A skipgram model such as FastText trained on general text usually cannot correctly produce dense vector representations for all scientific text because of the large number of domain-specific out of vocabulary tokens. To overcome this limit, it is necessary to retrain the word embedding model using scientific text. In fact, we compared a FastText trained on the Common Crawl data with SciFastText on the classification task and found that the F1 score of the latter is at least 2% higher. We do not use SciBERT as it has been trained only on Biology and Computer Science papers.

We compare three classifiers using the SciFastText embedding. The settings of each classifier are below. (1) **KNN**: The number of neighbors is set to 10. (2) **CNN**: The architecture contains 2 layers of 1-D convolutions. The number of

Table 3. Classification results for query topic classification.

Model	Precision	Recall	Macro Avg, F1
KNN	0.42	0.42	0.41
CNN	0.48	0.47	0.47
BiLSTM	**0.53**	**0.52**	**0.52**

[3] https://spacy.io/universe/project/video-spacys-ner-model

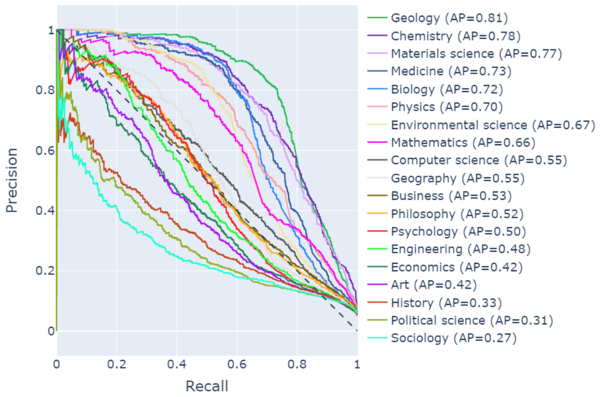

Figure 2. Precision-Recall curves for query topic classification sorted by AP for each research topic.

filters is set to 64. The dropout rate is 0.2. The max sequence length is 20. The model adopted the batch normalization, the cross-entropy loss with the Adam optimizer. (3) **BiLSTM**: The architecture contains 2 layers of Bidirectional LSTM. The number of units in each layer is 256. The dropout rate is 0.2. The max sequence length is 20. The model adopted batch normalization, the cross-entropy loss with Adam optimizer.

We tracked our validation loss and used early stopping to make sure that the models do not overfit. BiLSTM outperforms the other two models, achieving a macro average F1 score of 0.52 (Table 3). We held 10% of the data as the test set. We found that identifying certain research topics seemed more difficult than the others. For example, Geology achieved an F1 of 0.72 but Sociology achieved an F1 of 0.23. We believe this was because of the lack of domain-specific scientific vocabulary used in these domains.

The query topic classifier performed well in terms of average precision (AP), which summarizes a precision-recall curve as the weighted average of precision achieved at each threshold, for several research topics, such as Geology, Chemistry, Material Science, Medicine, Biology, and Physics (Figure 2). We analysed the confusion matrix of the predictions for our test set and found that the classifier does not perform well for topics with significant overlap across queries in other topics. For example, Economics and Business, Art and History are often confused.

Using the trained model we predict the research topics of the query logs in CiteSeerX. We find that Mathematics is the most searched research topic (Figure 3). It should be noted that our classifier often confuses Computer Science and Mathematics short-text.

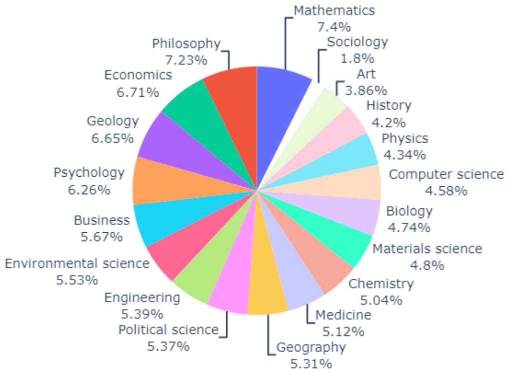

Figure 3. Research topic distribution of the query logs

5 Conclusion

In this work, we present a preliminary effort of understanding queries of an academic search engine in two aspects: query intent and topics. Our results indicate it is important to incorporate author names and exact titles as features while designing a ranking function. We demonstrate the challenges with academic query topic classification and showcase preliminary results by our classifiers on the CiteSeerX data. This log dataset has been anonymized and shared with the research community for further exploration.

6 Acknowledgements

We gratefully acknowledge partial support from the National Science Foundation.

References

[1] Y. Chang and H. Deng. 2020. *Query Understanding for Search Engines.* Springer International Publishing AG.

[2] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S. Weld. 2020. SPECTER: Document-level Representation Learning using Citation-informed Transformers. In *ACL*.

[3] C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. 1998. CiteSeer: An Automatic Citation Indexing System. In *Proceedings of the 3rd ACM International Conference on Digital Libraries*. 89–98.

[4] Jorge R. Herskovic, Len Y. Tanaka, William Hersh, and Elmer V. Bernstam. 2007. A Day in the Life of PubMed: Analysis of a Typical Day's Query Log. *J Am Med Inform Assoc* 14, 2 (03 2007), 212–220.

[5] Emilia Kacprzak, Laura M Koesten, Luis-Daniel Ibáñez, Elena Simperl, and Jeni Tennison. 2017. A query log analysis of dataset search. In *International Conference on Web Engineering*. Springer, 429–436.

[6] Bharath Kandimalla, Shaurya Rohatgi, Jian Wu, and C. Lee Giles. 2021. Large Scale Subject Category Classification of Scholarly Papers With Deep Attentive Neural Networks. *Front. Res. Metr. Anal.* 5 (2021), 31.

[7] Madian Khabsa, Zhaohui Wu, and C Lee Giles. 2016. Towards better understanding of academic search. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)*. IEEE, 111–114.

[8] Athar Sefid, Jian Wu, C Ge Allen, Jing Zhao, Lu Liu, Cornelia Caragea, Prasenjit Mitra, and C Lee Giles. 2019. Cleaning noisy and heterogeneous metadata for record linking across scholarly big datasets. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9601–9606.

[9] Jian Wu, Kyle Williams, Hung-Hsuan Chen, Madian Khabsa, Cornelia Caragea, Alexander Ororbia, Douglas Jordan, and C. Lee Giles. 2014. CiteSeerX: AI in a Digital Library Search Engine. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*. 2930–2937.