Poznań Contribution to TREC-PM 2019 *

Artur Cieślewicz[1], Jakub Dutkiewicz[2], and Czesław Jędrzejek[2]

[1] Poznań University of Medical Sciences, Collegium Maius, Fredry 10, 61-701 Poznań, Poland
[2] Poznań University of Technology, Plac Marii Skłodowskiej-Curie 5, 60-965 Poznań, Poland

Abstract. This paper describes Poznań contribution to the Precision Medicine track of the TREC 2019. In this submission we present several novelties. We cover the motivation for the hand-picked values of the weights assigned to the expanded query terms. We propose a result fusion method – slightly modified version of Borda Count algorithm. Additionally we use a learning to rank environment, we analyze the effectiveness of such an approach in combination with our other methods and analyze the achieved results. We also discuss our dedicated document processing methods. We achieve an improvement of up to 0.02 (infNDCG measure) over the baseline for Clinical Trials with our proposed methods, however the evaluation value of our baseline is much lower than the median of all contributions. The reverse effect happens in the Scientific Abstracts task, the baseline we propose is much stronger than the median, but the default setting of learning to rank proposition lowers the overall evaluation score.

Keywords: Information Retrieval · Query Expansion · Word Embedding · Learning to rank.

1 Introduction

The Information Retrieval(IR) task in this setting is defined as follows: given a set of documents, return a list of documents sorted by relevance of each document to the given query. TREC-PM[5][6] is a specific track, which evaluates the systems performing the IR tasks on a specific sets of documents and queries. Here, each query is defined as a description of a potential patient. It is divided into several fields: disease, related gene and its properties, such as variant or function; the query also contains the demographical description of a patient given by his age and gender. There are two document collections, one which consists of scientific abstracts of medical publication, and a second one which consists of formalized descriptions of clinical trials. We propose several improvements to the classical approach if executing an IR task. Namely, we propose a method of expanding a query with terms, which are similar to the query terms. We calculate the similarity between various terms upon the word embedding space. We use a

* Supported by the PUT DS grant no 04/45/DSPB/0197 and 04/45/DSMK/0200.

Borda Count based method for combining retrieved sets of documents by various systems. In this submission we also test the default setting of the learning to rank environment.
We describe the system architecture and discuss the steps we take in order to produce the retrieved lists of documents. We specifically focus on our dedicated document processing methods. We provide a description of the query processing. We also describe the settings we use in the Terrier tool, which performs the IR task. Next, we define and describe the runs we submitted and finally discuss the evaluation values we received.

2 System Description

The TREC-PM task consists of two types of documents - Scientific Abstracts (SA) and Clinical Trials (CT). Each type has its unique features and we take advantage of it. Scientific Abstracts is a set of PubMed article abstracts. This is a large set, it consists of 27564896 documents, which we deemed valid and a total size of 235GB. We use an efficient C++ code[3] in order to process the abstracts, dedicated scripts for specialized analysis and Terrier engine in order to perform the IR task[4].

As the document collection for Clinical Trials is fairly smaller, we use a classical XML parser. As we process the documents, we put parts of its contents into an SQL database. We use a word embedding space in order to expand the queries with specific terms. We use the learning to ranking environment in order to enhance the quality of the system. Finally, various results are aggregated with a Borda count method.

2.1 Scientific Abstracts

This section contains details of implementation of retrieval setup used for the Scientific Abstracts task. It consists of three major parts:

- Document processing
- Query processing
- Retrieval

Document processing. In order to process the set of documents, we use a fast, dedicated XML processor. The program starts by creating a specified number of processing threads. Each thread receives an output file name, a set of input file names and a list of key tags it is supposed to store. Program iteratively progresses trough the file and seeks the tags with names, which are on the list of key tags. Once it encounters the ⟨`document`⟩ tag, it creates an empty map, which

[3] https://github.com/dudenzz/myindex/
[4] terrier.org

consists of pairs (key feature, empty string). The program starts to retrieve data from this specific document. When a key tag is encountered, it starts to save its content into a proper map field until it encounters the same closing tag. Once the parser encounters closing ⟨`/document`⟩ tag, it stores the contents of processed document into a specified file. It allows us to process documents fast, but we omit the syntax checking feature of the parser - we assume all of the files have correct syntax. We use the following key tags:

- PMID
- ArticleTitle
- AbstractText
- Keyword
- NameOfSubstance
- DescriptorName

If a document contains at least two of the mentioned tags (PMID and one additional tag) it is deemed valid. Optionally we can assign the parser to skip documents with specific tags. In this particular case, we skip documents which contain ⟨`CommentsCorrectionsList`⟩ tag, as it indicates that a document directly references another document (e.g. it is a comment, erratum or a technical note on another document).

Query processing There is one set of queries for both the Scientific Abstracts task and Clinical Trials task. Each query consists of three fields - disease name, related genes and their description (such as variant or gene function) and patient demographics (gender and age). In hope to expand the queries for the Scientific Abstracts task we have retrieved the word embeddings using a classical Word2Vec [4] approach and a collection of Scientific Abstracts documents. We use a relatively simple idea of finding a query expansion candidate terms. For each term in the query we look for the most similar vectors in the word embedding space to the vector of this term. We use a cosine definition of similarity given by (1).

$$sim(w_1, w_2) = \frac{\sum_i v_{1i} v_{2i}}{\sqrt{\sum_i v_{1i}^2} \sqrt{\sum_i v_{2i}^2}} \tag{1}$$

In (1) w_1 and w_2 relate to the words, which are being compares. Word embeddings collection $\Omega \in \mathcal{R}^N$ contains vectors $v_1 \in \Omega$ and $v_2 \in \Omega$ which are embeddings for the w_1 and w_2 words. In this setup we are able to find very interesting, new queries, however, the newly generated queries are not equivalent to the original query. This is due to the nature of a replaceability of a term. The word embedding space we use is built upon the word contexts. Thus, words which appear in similar contexts are supposed to be similar. We wish those words to be semantically related as synonyms, but it's often not the case. In most cases the words are semantically parallel (e.g. name of a different type of cancer, name of gene which relate to a very similar, but different type of disease, name of gene which could be related to this disease, but the query does not ask for this gene).

Table 1 consists of several examples of this issue. After a qualitative analysis we decided to use only the original version of queries in the Scientific Abstracts task. However, using this technique, we expanded the queries for the Clinical Trials task.

Query Term	MS	2nd MS	3rd MS
meningioma	ependymoma (0.867)	astrocytoma (0.858)	chordoma (0.81)
KRAS	BRAF(0.881)	PIK3CA(0.874)	TP53(0.847)
V600E	BRAFV600E(0.830)	V600(0.790)	p.V600(0.789)
melanoma	SCCHN(0.719)	tumor(0.717)	cSCC(0.707)
E545K	H1047R(0.861)	G12V(0.788)	PI3KCA(0.754)

Table 1. Examples of the similarity check for the query terms. MS stands for the most similar, similarity value is given in brackets. In the examples we omit various inflection forms of the original term (usually it is a plural form of the same words).

Retrieval In order to perform the Retrieval we use the Terrier tool. We follow the standard procedure of indexing the processed documents. We also employ Terrier to generate the default learning to rank environment. In this environment we use the following features as input:

- BM25 calculated for each field in the document,
- Length of each field in the document,
- Proximity features: DFR dependence score and MRF dependence score.

We train the model on the TREC-PM 2018 data. We use this setup along with the divergence from randomness retrieval models:

- LGD [1],
- DPH [1],

in order to create an output ranking. We use the created ranking lists to test our data Borda count based fusion method.

2.2 Clinical Trials

We test a slightly different approach for the Clinical Abstracts task. This section contains the details of implementation of our method for this task.

Document Processing As the document collection for Clinical Trials is fairly smaller, we use a classical XML parser. Thus, we examine the syntax of each document within the set. As we process the documents, we put parts of its contents into an SQL database. Specifically, we use the summary and description fields as the body of a document. We also take use of the demographics data stored within the documents. A complete list of processed tags is as follows:

- brief title
- official title
- brief summary
- detailed description
- primary outcome
- secondary outcome
- condition
- arm group
- condition browse
- intervention browse
- keyword
- criteria

The documents to be indexed are generated with use of an automatic script. The script puts contents of each field into respectively body and the title of a document. We additionally split the criteria tag into two parts - exclusion and inclusion criteria. The inclusion criteria is put into the document body, while the exclusion criteria is put into a special "negative" tag. We additionally replace every term "KIT" in the collection with "gene_kit" string. The difference between gene KIT and kits used for isolation of DNA or ELISA is easily recognizable with use of the upper cased letters. As the case is lost during the indexing and generation of the word embedding space, we differentiate "kit" from "KIT" at this level.

Query Processing We start the query processing by calculating the word embedding space. We use a corpus generated upon the Clinical Trials documents collection by concatenating of all the processed fields. Instead of searching for an expansion for each term in the query separately, we aggregate the similarities before judging whether a term is a valid expansion candidate. The aggregation is an average of similarities of all query terms to the potential expansion term. We choose the expansion terms with computed similarity above the threshold of 0.6. In the query processing section for Scientific Abstracts, we've shown that the expansion terms might distort the results as they change the sense of a query. We hypothesise, that adding terms to the query with much lower weight would improve the quality of a retrieval. This is due to the fact that terms added with much lower weight potentially wouldn't change the order of ranking for the top scoring documents, they would however add an information to the tail of the retrieved list. For example, if we look for documents about the Meningioma, and we find no such document within the entire collection, it's better to retrieve a document related to Ependymoma, than it is to retrieve a randomly selected document. Thus we add the new terms with the following weights:

- Expanded query term weight : 1; Original query term : 120 (used for the Borda Count fusion method)
- Expanded query term weight : 1; Original query term : 140 (this is a default setting)

- Expanded query term weight : 1; Original query term : 160 (used for the Borda Count fusion method)
- Expanded query term weight : 1; Original query term : 180 (used for the Borda Count fusion method)

Retrieval We use the Terrier system in order to create a document index. We retrieve the documents with BB2 retrieval model. We also use Terrier to implement the learning to rank environment. We train the model on data from TREC-PM 2018 and TREC-PM 2017. We use the same set of features as in the Scientific Abstracts task:

- BM25 calculated for each field in the document,
- Length of each field in the document,
- Proximity features: DFR dependence score and MRF dependence score.

Finally, we exclude the trials with inadequate description of demographics from the lists of retrieved documents,

2.3 Results fusion

One can observe, that systems performance vary on the query it processes. If we took only the best system for each query, the averaged result would be much better than an average of a best system. Thus we believe it is reasonable to fuse the results obtained by various systems. We implement the Borda Count [3] function in order to retrieve the final ranking of documents. The Borda Count method we use is given by

$$s_{C,Q}(D) = \sum_{t \in T} \frac{k - r_{t,C,Q}(D)}{log_2(r_{t,C,Q}(D) + 1)} \tag{2}$$

Final score s of a document D given collection of documents C and a query Q, is a sum of components produced by various systems $t \in T$. Here, the $r_{t,C,Q}(D)$ is a rank of a document D retrieved by system t given a query Q and a document collection C, and k is a size of the retrieved list.

3 Results Analysis

TREC-PM currently uses three evaluation measures:

- inferred Normalized Discounted Cumulative Gain (infNDCG) - an inferred version of the normalized cumulative information gain. This measure takes into account the position of a document on the retrieved list [7].
- Precision at 10 (P@10) - proportion of relevant documents within the top ten retrieved documents[2].
- R-precision (R-prec) - precision at R, where R is a number of relevant documents within the collection for a given query[2].

Often, the evaluation values are aggregated over the set of queries as an average. We believe this is a necessary operation when comparing systems head to head in order to pick a better one. However, we also believe, that this aggregation is a cause of an information loss. By averaging we lose an information of how the system performs on a various queries. We have prepared a set of tables, in which we can observe our system performs much better than the median on a subset of queries; as well as a subset of queries in which our system performed much worse. This issue is specifically disturbing for our version of Clinical Trials system.

3.1 Submitted runs

We submit a total of nine runs, four runs for Scientific Abstracts and five runs for Clinical Trials. The purpose of the Scientific Abstracts runs is to test our document processing methods as well as to test learning to rank environment and the Borda Count results fusion method:

1. SAsimpleLGD - a default setting retrieval. The purpose of this run is to test the quality of the document processing. We hypothesise that the LGD retrieval model is a very strong baseline. Thus we employ it to perform the retrieval.
2. SA_LGD_letor and SA_DPH_letor - these two runs use the default learning to rank environment, purpose of these runs is to test the quality of learning to rank with the default setting as well as create comparable data for the results fusion method.
3. SA_bc - a run, which uses the Borda Count method in order to concatenate two retrieval rankings. The Borda Count is calculated upon the SA_LGD_letor and SA_DPH_letor runs.

We submit five runs for the Clinical Trials task. We use the BB2 retrieval model in all of them In addition to the goals described in the above section, the purpose of these runs is to test our hypothesis of improving the quality of retrieval by adding word embedding based terms with much lower weights.

1. simple - a default setting retrieval. We use its as a comparison. We hope to achieve results better than the ones generated by this run.
2. simple_letor - a default setting run with learning to rank employed. We use no word embedding based expansion here.
3. w2v_noletor - a run with word embedding based expansion terms.
4. w2v_letor - a combination of learning to rank environment with word embedding based expansions.
5. bc - a run, which uses a default setting run without learning to run and four different runs with word embedding based expansion terms. A final result is a combination of results with the Borda Count function.

The aggregated results of those runs are presented in the table 2. We can see that for the Scientific Abstracts task, the baseline proved to be very strong.

We believe that our method of document processing, did well and it is one of the reasons of a decent evaluation value. Unfortunately the learning to rank environment did not work properly. Whether it is due to a wrong selection of the features - it requires further investigation. However, we observe that the Borda Count method did particularly well for this task.

As for the Clinical Trials, we can see that the baseline we chose is, contrary to the one picked for Scientific Abstracts, relatively weak. However, every enhancement to the baseline method we propose seems to improve the evaluation values. In particular, the word embedding based query expansion in combination with learning to rank environment seems to improve results by the largest margin.

3.2 Further analysis

We also provide a detailed "per-query" results. Figures 1-6 in the Appendix A illustrate specific results for each run.
We observe that our proposal for the Scientific Abstracts correlates well with an a median submission. The runs perform well for easy queries and perform worse for harder queries. Runs with learning to rank implemented perform better for some queries (e.g. query no 12, 15, 25) but on average the simple version gives the best results.
The Clinical Trials set of results is very different. We observe that the system we propose works really well for majority of the queries, however there are some queries for which it returns nonsensical lists of documents. In particular queries no 4,5,12,13,18,27,36 and 38 are problematic. If those queries were excluded the overall evaluation value would drastically go up. It is an issue which requires emergent investigation.

Note: how to read the figures The top row (labeled as 0) of each figure is an average of the remainder of results. Each figure is split into three parts. The middle part - a column labeled as Trec median - is a reference point. It is a median evaluation value for all submitted runs. The lower this value gets (the more it is red), the harder the query for an average system is. The left part of the figure illustrates how well did our system do on that query. The right part of the figure illustrates how well did our system do on that query compared to an average system.

4 Conclusions

This submission highlights an important issue. Some systems vastly underperform in specific settings. There are two solutions to this problem. We could either formulate the description of such settings, so we know not to use those systems when these specific conditions are met. We suppose that the conditions would be easily described with use of the annotated set, which is not very useful, but we also believe there is a correlation between the features of annotated set (such as number of annotated relevant documents)

Run name	infNDCG	P@10	Rprec
SA_DPH_letor	0,45	0,50	0,28
SA_LGD_letor	0,45	0,51	0,27
SA_bc	0,47	0,52	0,31
SA_simple_LGD	0,48	0,54	0,31
Trec Median	0,46	0,55	0,28
bc	0,47	0,44	0,34
simple	0,47	0,44	0,33
simple_letor	0,48	0,44	0,35
w2v_letor	0,48	0,42	0,35
w2v	0,47	0,44	0,33
Trec Median	0,51	0,47	0,35

Table 2. Aggregated results for all submitted runs.

and a shape of the retrieval score distribution. We plan to analyze this issue. The second way of solving this issue is to use a combination of methods. In particular, the proposed Borda Count method works fairly well and it improves the overall evaluation value.

We have also examined two of our methods of document processing. The document processing proposed for Scientific Abstracts seem to work fairly well. The processing we proposed for the Clinical Trials requires further investigation, as it might be a cause of the low baseline score.

References

1. Clinchant, S., Gaussier, É.: Information-based models for ad hoc IR. In: Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010. pp. 234–241 (2010). https://doi.org/10.1145/1835449.1835490, https://doi.org/10.1145/1835449.1835490
2. Craswell, N.: R-Precision, pp. 2453–2453. Springer US, Boston, MA (2009). https://doi.org/10.1007/978-0-387-39940-9_486, https://doi.org/10.1007/978-0-387-39940-9_486
3. Lin, S.: Rank aggregation methods. Wiley Interdisciplinary Reviews: Computational Statistics **2**(5), 555–570 (2010). https://doi.org/10.1002/wics.111, https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.111
4. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Bengio, Y., LeCun, Y. (eds.) 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings (2013), http://arxiv.org/abs/1301.3781
5. Roberts, K., Demner-Fushman, D., Voorhees, E.M., Hersh, W.R., Bedrick, S., Lazar, A.J.: Overview of the TREC 2018 precision medicine track. In: Voorhees, E.M., Ellis, A. (eds.) Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018, Gaithersburg, Maryland, USA, November 14-16, 2018. vol. Special Publication 500-331. National Institute of Standards and Technology (NIST) (2018), https://trec.nist.gov/pubs/trec27/papers/Overview-PM.pdf
6. Roberts, K., Demner-Fushman, D., Voorhees, E.M., Hersh, W.R., Bedrick, S., Lazar, A.J., Pant, S.: Overview of the TREC 2017 precision medicine track. In: Voorhees, E.M., Ellis, A. (eds.) Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017, Gaithersburg, Maryland, USA, November 15-17, 2017. vol. Special Publication 500-324. National Institute of Standards and Technology (NIST) (2017), https://trec.nist.gov/pubs/trec26/papers/Overview-PM.pdf
7. Yilmaz, E., Kanoulas, E., Aslam, J.A.: A simple and efficient sampling method for estimating AP and NDCG. In: Myaeng, S., Oard, D.W., Sebastiani, F., Chua, T., Leong, M. (eds.) Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008. pp. 603–610. ACM (2008). https://doi.org/10.1145/1390334.1390437, https://doi.org/10.1145/1390334.1390437

A Query specific results

	Labels	simple_letor	w2v_noletor	w2v_letor	simple	bc	Trec median	diff1	diff2	diff3	diff4	diff5
0	0	0.4807	0.4669	0.481	0.4676	0.4743	0.513692	-0.0329921	-0.0467921	-0.0326921	-0.0460921	-0.0393921
1	1	0.8814	0.7291	0.8763	0.724	0.7716	0.7349	0.1465	-0.0058	0.1414	-0.0109	0.0367
2	2	0.9846	0.9872	0.9846	0.9872	0.9896	0.859	0.1256	0.1282	0.1256	0.1282	0.1306
3	3	0.6778	0.5851	0.6778	0.5851	0.5807	0.4687	0.2091	0.1164	0.2091	0.1164	0.112
4	4	0.2882	0.249	0.2882	0.2489	0.2863	0.2119	0.0763	0.0371	0.0763	0.037	0.0744
5	5	0.1545	0.1499	0.1545	0.1499	0.1447	0.1158	0.0387	0.0341	0.0387	0.0341	0.0289
6	6	0.3761	0.3039	0.3761	0.3039	0.3521	0.7078	-0.3317	-0.4039	-0.3317	-0.4039	-0.3557
7	7	0.3355	0.339	0.3355	0.339	0.3613	0.8336	-0.4981	-0.4946	-0.4981	-0.4946	-0.4723
8	8	0.5862	0.5782	0.5862	0.5782	0.5775	0.5526	0.0336	0.0256	0.0336	0.0256	0.0249
9	9	0.7339	0.7655	0.7422	0.7567	0.7627	0.7021	0.0318	0.0634	0.0401	0.0546	0.0606
10	10	0.6104	0.6219	0.5237	0.6764	0.6353	0.5968	0.0136	0.0251	-0.0731	0.0796	0.0385
11	11	0.725	0.6833	0.725	0.6833	0.7101	0.6015	0.1235	0.0818	0.1235	0.0818	0.1086
12	12	0	0	0	0	0	0.3707	-0.3707	-0.3707	-0.3707	-0.3707	-0.3707
13	13	0.0131	0	0.0131	0	0	0.5677	-0.5546	-0.5677	-0.5546	-0.5677	-0.5677
14	14	0.3719	0.3667	0.3719	0.3667	0.3647	0.316	0.0559	0.0507	0.0559	0.0507	0.0487
15	15	0.5458	0.4808	0.5458	0.4808	0.5039	0.5331	0.0127	-0.0523	0.0127	-0.0523	-0.0292
16	16	0.4554	0.4777	0.4554	0.4777	0.4765	0.439	0.0164	0.0387	0.0164	0.0387	0.0375
17	17	0.363	0.4216	0.363	0.4216	0.4604	0.3045	0.0585	0.1171	0.0585	0.1171	0.1559
18	18	0	0	0	0	0	0.2074	-0.2074	-0.2074	-0.2074	-0.2074	-0.2074
19	19	0.1863	0.2144	0.1863	0.2144	0.2131	0.1863	0	0.0281	0	0.0281	0.0268
20	20	0.3927	0.2683	0.3927	0.2698	0.2981	0.2817	0.111	-0.0134	0.111	-0.0119	0.0164
21	21	0.7544	0.7992	0.7806	0.7872	0.7874	0.7748	-0.0204	0.0244	0.0058	0.0124	0.0126
22	22	0.5591	0.5678	0.5591	0.5678	0.5842	0.4919	0.0672	0.0759	0.0672	0.0759	0.0923
23	23	0.3361	0.3253	0.3361	0.3253	0.3309	0.0767	0.2594	0.2486	0.2594	0.2486	0.2542
24	24	0.3128	0.2971	0.3128	0.2971	0.3412	0.2971	0.0157	0	0.0157	0	0.0441
25	25	0.8315	0.8352	0.8315	0.8344	0.8354	0.7715	0.06	0.0637	0.06	0.0629	0.0639
26	26	0.7916	0.7873	0.7916	0.7873	0.7427	0.5958	0.1958	0.1915	0.1958	0.1915	0.1469
27	27	0.1593	0.1477	0.1593	0.1477	0.1418	0.5695	-0.4102	-0.4218	-0.4102	-0.4218	-0.4277
28	28	0.4939	0.5393	0.4939	0.5391	0.5294	0.4975	-0.0036	0.0418	-0.0036	0.0416	0.0319
29	29	0.555	0.4735	0.4816	0.4735	0.4745	0.4386	0.1164	0.0349	0.043	0.0349	0.0359
30	30	0.31	0.4011	0.31	0.4011	0.3934	0.3125	-0.0025	0.0886	-0.0025	0.0886	0.0809
31	31	0.8597	0.8597	1	0.8597	0.8597	0.7602	0.0995	0.0995	0.2398	0.0995	0.0995
34	34	0.9239	0.8956	0.9239	0.8956	0.8956	0.4796	0.4443	0.416	0.4443	0.416	0.416
35	35	0.6954	0.7061	0.6954	0.7061	0.7339	0.5214	0.174	0.1847	0.174	0.1847	0.2125
36	36	0	0	0	0	0	0.8012	-0.8012	-0.8012	-0.8012	-0.8012	-0.8012
37	37	0.9903	0.9944	0.9903	0.9944	0.9944	0.8437	0.1466	0.1507	0.1466	0.1507	0.1507
38	38	0	0	0	0	0	1	1	1	1	1	1
39	39	0.3196	0.3196	0.3196	0.3196	0.3196	0.1266	0.193	0.193	0.193	0.193	0.193
40	40	0.6934	0.5706	0.6934	0.5706	0.5706	0.5706	0.1228	0	0.1228	0	0

Fig. 1. Query specific results for Clinical Trials. Evaluation measure: infNDCG. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus simple_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.

	Labels	simple_letor	w2v_noletor	w2v_letor	simple	bc	Trec median	diff1	diff2	diff3	diff4	diff5
0	0	0.4421	0.4421	0.4237	0.4395	0.4395	0.465789	-0.0236895	-0.0236895	-0.0420895	-0.0262895	-0.0262895
1	1	0.9	0.5	0.7	0.4	0.5	0.8	0.1	-0.3	-0.1	-0.4	-0.3
2	2	1	1	1	1	1	1	0	0	0	0	0
3	3	0.3	0.3	0.3	0.3	0.3	0.2	0.1	0.1	0.1	0.1	0.1
4	4	0.4	0.4	0.4	0.4	0.4	0.4	0	0	0	0	0
5	5	0.1	0.2	0.1	0.2	0.2	0.1	0	0.1	0	0.1	0.1
6	6	0.6	0.4	0.6	0.4	0.4	0.8	-0.2	-0.4	-0.2	-0.4	-0.4
7	7	0.9	0.9	0.9	0.9	0.9	1	-0.1	-0.1	-0.1	-0.1	-0.1
8	8	0.7	0.7	0.7	0.7	0.7	0.7	0	0	0	0	0
9	9	1	1	1	0.9	1	0.9	0.1	0.1	0.1	0	0.1
10	10	1	1	0.6	1	0.9	0.9	0.1	0.1	-0.3	0.1	0
11	11	0.8	0.8	0.8	0.8	0.8	0.8	0	0	0	0	0
12	12	0	0	0	0	0	0.2	-0.2	-0.2	-0.2	-0.2	-0.2
13	13	0	0	0	0	0	0.6	-0.6	-0.6	-0.6	-0.6	-0.6
14	14	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1
15	15	0.6	0.7	0.6	0.7	0.7	0.7	-0.1	0	-0.1	0	0
16	16	0.6	0.6	0.6	0.6	0.6	0.6	0	0	0	0	0
17	17	0.4	0.4	0.4	0.4	0.4	0.2	0.2	0.2	0.2	0.2	0.2
18	18	0	0	0	0	0	0.2	-0.2	-0.2	-0.2	-0.2	-0.2
19	19	0.3	0.2	0.3	0.2	0.3	0.2	0.1	0	0.1	0	0.1
20	20	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0	0
21	21	0.8	0.8	0.7	0.9	0.8	0.8	0	0	-0.1	0.1	0
22	22	1	1	1	1	1	0.8	0.2	0.2	0.2	0.2	0.2
23	23	0.3	0.2	0.3	0.2	0.2	0.1	0.2	0.1	0.2	0.1	0.1
24	24	0.4	0.5	0.4	0.5	0.6	0.5	-0.1	0	-0.1	0	0.1
25	25	0.9	1	0.9	1	0.9	0.8	0.1	0.2	0.1	0.2	0.1
26	26	0.4	0.5	0.4	0.5	0.5	0.4	0	0.1	0	0.1	0.1
27	27	0.3	0.2	0.3	0.2	0.2	0.7	-0.4	-0.5	-0.4	-0.5	-0.5
28	28	0.6	0.9	0.6	0.9	0.8	0.8	-0.2	0.1	-0.2	0.1	0
29	29	0.2	0.2	0.2	0.2	0.2	0.2	0	0	0	0	0
30	30	0.3	0.4	0.3	0.4	0.4	0.4	-0.1	0	-0.1	0	0
31	31	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1
34	34	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1
35	35	0.4	0.4	0.4	0.4	0.4	0.3	0.1	0.1	0.1	0.1	0.1
36	36	0	0	0	0	0	0.5	-0.5	-0.5	-0.5	-0.5	-0.5
37	37	0.5	0.5	0.5	0.5	0.5	0.4	0.1	0.1	0.1	0.1	0.1
38	38	0	0	0	0	0	0.1	-0.1	-0.1	-0.1	-0.1	-0.1
39	39	0.1	0.1	0.1	0.1	0.1	0	0.1	0.1	0.1	0.1	0.1
40	40	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1

Fig. 2. Query specific results for Clinical Trials. Evaluation measure: P@10. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus simple_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.

	Labels	simple_letor	w2v_noletor	w2v_letor	simple	bc	Trec median	diff1	diff2	diff3	diff4	diff5
0	0	0.3503	0.3315	0.3481	0.3313	0.3386	0.347734	0.00256579	-0.0162342	0.000365789	-0.0164342	-0.00913421
1	1	0.6014	0.5942	0.6087	0.5797	0.6014	0.5435	0.0579	0.0507	0.0652	0.0362	0.0579
2	2	0.4533	0.514	0.4533	0.514	0.472	0.3832	0.0701	0.1308	0.0701	0.1308	0.0888
3	3	0.3	0.3	0.3	0.3	0.3	0.2	0.1	0.1	0.1	0.1	0.1
4	4	0.236	0.2135	0.236	0.2135	0.236	0.1685	0.0675	0.045	0.0675	0.045	0.0675
5	5	0.0577	0.0385	0.0577	0.0385	0.0577	0.0769	-0.0192	-0.0384	-0.0192	-0.0384	-0.0192
6	6	0.069	0.069	0.069	0.069	0.069	0.3197	-0.2507	-0.2507	-0.2507	-0.2507	-0.2507
7	7	0.0822	0.0856	0.0822	0.0856	0.0856	0.4212	-0.339	-0.3356	-0.339	-0.3356	-0.3356
8	8	0.5217	0.5652	0.5217	0.5652	0.5652	0.5217	0	0.0435	0	0.0435	0.0435
9	9	0.7101	0.6377	0.6957	0.6377	0.6812	0.6087	0.1014	0.029	0.087	0.029	0.0725
10	10	0.6	0.6286	0.4857	0.6286	0.6286	0.5143	0.0857	0.1143	-0.0286	0.1143	0.1143
11	11	0.6	0.6	0.6	0.6	0.6	0.6	0	0	0	0	0
12	12	0	0	0	0	0	0.2222	-0.2222	-0.2222	-0.2222	-0.2222	-0.2222
13	13	0	0	0	0	0	0.45	-0.45	-0.45	-0.45	-0.45	-0.45
14	14	0.1538	0.1538	0.1538	0.1538	0.1538	0.1538	0	0	0	0	0
15	15	0.4098	0.3989	0.4098	0.3989	0.4262	0.3388	0.071	0.0601	0.071	0.0601	0.0874
16	16	0.2857	0.2619	0.2857	0.2619	0.2619	0.3095	-0.0238	-0.0476	-0.0238	-0.0476	-0.0476
17	17	0.2778	0.2222	0.2778	0.2222	0.2778	0.1667	0.1111	0.0555	0.1111	0.0555	0.1111
18	18	0	0	0	0	0	0.1818	-0.1818	-0.1818	-0.1818	-0.1818	-0.1818
19	19	0.1429	0.1429	0.1429	0.1429	0.1429	0.1143	0.0286	0.0286	0.0286	0.0286	0.0286
20	20	0.15	0.2	0.15	0.2	0.2	0.2	-0.05	0	-0.05	0	0
21	21	0.7308	0.7564	0.7692	0.7308	0.7436	0.6923	0.0385	0.0641	0.0769	0.0385	0.0513
22	22	0.4833	0.4333	0.4833	0.4333	0.4667	0.4	0.0833	0.0333	0.0833	0.0333	0.0667
23	23	0.25	0.25	0.25	0.25	0.25	0	0.25	0.25	0.25	0.25	0.25
24	24	0.2188	0.2188	0.2188	0.2188	0.2812	0.25	-0.0312	-0.0312	-0.0312	-0.0312	0.0312
25	25	0.5455	0.5152	0.5455	0.5455	0.5455	0.5152	0.0303	0	0.0303	0.0303	0.0303
26	26	0.5714	0.5714	0.5714	0.5714	0.5714	0.4286	0.1428	0.1428	0.1428	0.1428	0.1428
27	27	0.0652	0.0326	0.0652	0.0326	0.0543	0.4457	-0.3805	-0.4131	-0.3805	-0.4131	-0.3914
28	28	0.5229	0.5138	0.5229	0.5138	0.5138	0.4312	0.0917	0.0826	0.0917	0.0826	0.0826
29	29	0.25	0.25	0.25	0.25	0.25	0.25	0	0	0	0	0
30	30	0.2667	0.2667	0.2667	0.2667	0.2667	0.2667	0	0	0	0	0
31	31	1	1	1	1	1	0.5	0.5	0.5	0.5	0.5	0.5
34	34	0.5	0.5	0.5	0.5	0.5	0	0.5	0.5	0.5	0.5	0.5
35	35	0.4545	0.3636	0.4545	0.3636	0.3636	0.2727	0.1818	0.0909	0.1818	0.0909	0.0909
36	36	0	0	0	0	0	0.6667	-0.6667	-0.6667	-0.6667	-0.6667	-0.6667
37	37	0.8	0.8	0.8	0.8	0.8	0.6	0.2	0.2	0.2	0.2	0.2
38	38	0	0	0	0	0	1	-1	-1	-1	-1	-1
39	39	0.5	0.5	0.5	0.5	0.5	0	0.5	0.5	0.5	0.5	0.5
40	40	0.5	0	0.5	0	0	0	0.5	0	0.5	0	0

Fig. 3. Query specific results for Clinical Trials. Evaluation measure: Rprec. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus simple_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.

	Labels	SA_LGD_letor	SA_bc	SAsimpleLGD	SA_DPH_letor	Trec median	diff1	diff2	diff3	diff4
0	0	0.4469	0.4707	0.4755	0.4542	0.455902	-0.0090025	0.0147975	0.0195975	-0.0017025
1	1	0.5386	0.6061	0.6102	0.5926	0.6768	0.1382	-0.0707	-0.0666	-0.0842
2	2	0.9461	0.9568	0.9611	0.9207	0.8556	0.0905	0.1012	0.1055	0.0651
3	3	0.3462	0.376	0.3702	0.2624	0.2426	0.1036	0.1334	0.1276	0.0198
4	4	0.4865	0.5078	0.4639	0.3857	0.3307	0.1558	0.1771	0.1332	0.055
5	5	0.4366	0.4829	0.4825	0.3276	0.3276	0.109	0.1553	0.1549	0
6	6	0.3599	0.3344	0.3275	0.4193	0.4788	-0.1189	-0.1444	-0.1513	-0.0595
7	7	0.94	0.908	0.9096	0.9307	0.8883	0.0517	0.0197	0.0213	0.0424
8	8	0.3995	0.469	0.4737	0.4661	0.4875	-0.088	-0.0185	-0.0138	-0.0214
9	9	0.7897	0.7682	0.7476	0.7686	0.6211	0.1686	0.1471	0.1265	0.1475
10	10	0.3707	0.4105	0.573	0.3773	0.5074	-0.1367	-0.0969	0.0656	-0.1301
11	11	0.6706	0.6507	0.7005	0.662	0.6299	0.0407	0.0208	0.0706	0.0321
12	12	0.3853	0.3689	0.2455	0.3841	0.2754	0.1099	0.0935	-0.0299	0.1087
13	13	0.2932	0.2856	0.3683	0.3136	0.3585	-0.0653	-0.0729	0.0098	-0.0449
14	14	0	0	0	0	0.1304	-0.1304	-0.1304	-0.1304	-0.1304
15	15	0.4843	0.4926	0.4949	0.6048	0.6272	-0.1429	-0.1346	-0.1323	-0.0224
16	16	0.8381	0.8113	0.7424	0.8551	0.8015	0.0366	0.0098	-0.0591	0.0536
17	17	0.1727	0.2383	0.3303	0.1945	0.3303	-0.1576	-0.092	0	-0.1358
18	18	0.6453	0.6443	0.65	0.615	0.5571	0.0882	0.0872	0.0929	0.0579
19	19	0.1812	0.1674	0.1624	0.0604	0.0918	0.0894	0.0756	0.0706	-0.0314
20	20	0.1662	0.125	0.1291	0.2218	0.1659	0.0003	-0.0409	-0.0368	0.0559
21	21	0.2448	0.3279	0.3441	0.2879	0.3854	-0.1406	-0.0575	-0.0413	-0.0975
22	22	0.2677	0.2727	0.2244	0.2379	0.3222	-0.0545	-0.0495	-0.0978	-0.0843
23	23	0.304	0.28	0.2591	0.1708	0.2339	0.0701	0.0461	0.0252	-0.0631
24	24	0.2717	0.3516	0.3762	0.163	0.3103	-0.0386	0.0413	0.0659	-0.1473
25	25	0.7656	0.8053	0.7852	0.7878	0.8515	-0.0859	-0.0462	-0.0663	-0.0637
26	26	0.5244	0.5617	0.6658	0.5439	0.5617	-0.0373	0	0.1041	-0.0178
27	27	0.4179	0.473	0.5584	0.3846	0.507	-0.0891	-0.034	0.0514	-0.1224
28	28	0.7704	0.8014	0.7887	0.7266	0.7711	-0.0007	0.0303	0.0176	-0.0445
29	29	0.4238	0.5293	0.5238	0.5008	0.4401	-0.0163	0.0892	0.0837	0.0607
30	30	0.6357	0.704	0.8268	0.481	0.6739	-0.0382	0.0301	0.1529	-0.1929
31	31	0.3543	0.5545	0.5622	0.53	0.4087	-0.0544	0.1458	0.1535	0.1213
32	32	0.3414	0.4386	0.4493	0.4137	0.3923	-0.0509	0.0463	0.057	0.0214
33	33	0.0717	0.0591	0	0.1223	0.128	-0.0563	-0.0689	-0.128	-0.0057
34	34	0.2791	0.3129	0.2155	0.3379	0.2373	0.0418	0.0756	-0.0218	0.1006
35	35	0.6633	0.6615	0.6403	0.7127	0.5712	0.0921	0.0903	0.0691	0.1415
36	36	0.338	0.3408	0.3035	0.4642	0.3688	-0.0308	-0.028	-0.0653	0.0954
37	37	0.7602	0.7736	0.8037	0.8391	0.806	-0.0458	-0.0324	-0.0023	0.0331
38	38	0.3793	0.3673	0.3243	0.4041	0.2968	0.0825	0.0705	0.0275	0.1073
39	39	0.5705	0.5334	0.504	0.5894	0.5001	0.0704	0.0333	0.0039	0.0893
40	40	0.0405	0.0766	0.1214	0.109	0.0854	-0.0449	-0.0088	0.036	0.0236

Fig. 4. Query specific results for Scientific Abstracts Trials. Evaluation measure: infNDCG. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus SA_LGD_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.

	Labels	SA_LGD_letor	SA_bc	SAsimpleLGD	SA_DPH_letor	Trec median	diff1	diff2	diff3	diff4
0	0	0.505	0.515	0.54	0.5	0.545	-0.04	-0.03	-0.005	-0.045
1	1	0.6	0.8	0.7	0.7	0.8	-0.2	0	-0.1	-0.1
2	2	1	1	1	1	1	0	0	0	0
3	3	0.3	0.3	0.4	0.1	0.1	0.2	0.2	0.3	0
4	4	0.4	0.6	0.7	0.6	0.6	-0.2	0	0.1	0
5	5	0.8	0.7	0.8	0.7	0.8	0	-0.1	0	-0.1
6	6	0.4	0.4	0.4	0.3	0.4	0	0	0	-0.1
7	7	1	0.9	0.9	1	0.9	0.1	0	0	0.1
8	8	0.5	0.4	0.6	0.4	0.6	-0.1	-0.2	0	-0.2
9	9	0.9	0.9	0.9	1	0.9	0	0	0	0.1
10	10	0.6	0.6	0.6	0.5	0.7	-0.1	-0.1	-0.1	-0.2
11	11	0.7	0.8	1	0.9	0.9	-0.2	-0.1	0.1	0
12	12	0.3	0.3	0.3	0.3	0.2	0.1	0.1	0.1	0.1
13	13	0.1	0.2	0.2	0.2	0.3	-0.2	-0.1	-0.1	-0.1
14	14	0	0	0	0	0.2	-0.2	-0.2	-0.2	-0.2
15	15	0.9	0.8	0.9	0.7	0.9	0	-0.1	0	-0.2
16	16	0.9	1	1	0.9	0.9	0	0.1	0.1	0
17	17	0.1	0.4	0.4	0.3	0.5	-0.4	-0.1	-0.1	-0.2
18	18	0.6	0.7	0.8	0.7	0.7	-0.1	0	0.1	0
19	19	0.3	0.3	0.2	0	0.2	0.1	0.1	0	-0.2
20	20	0	0	0.1	0.1	0.1	-0.1	-0.1	0	0
21	21	0.6	0.5	0.6	0.5	0.6	0	-0.1	0	-0.1
22	22	0.6	0.6	0.7	0.3	0.6	0	0	0.1	-0.3
23	23	0.3	0.1	0.1	0.2	0.2	0.1	-0.1	-0.1	0
24	24	0.7	0.7	0.7	0.5	0.8	-0.1	-0.1	-0.1	-0.3
25	25	0.8	0.8	0.5	0.9	0.9	-0.1	-0.1	-0.4	0
26	26	0.6	0.6	0.7	0.8	0.8	-0.2	-0.2	-0.1	0
27	27	0.6	0.7	0.9	0.5	0.7	-0.1	0	0.2	-0.2
28	28	0.8	0.8	1	0.7	0.8	0	0	0.2	-0.1
29	29	0.3	0.6	0.5	0.6	0.6	-0.3	0	-0.1	0
30	30	0.7	0.8	0.9	0.7	0.7	0	0.1	0.2	0
31	31	0.1	0.3	0.3	0.3	0.2	-0.1	0.1	0.1	0.1
32	32	0	0.2	0.3	0.2	0.2	-0.2	0	0.1	0
33	33	0	0	0	0	0	0	0	0	0
34	34	0.4	0.1	0.1	0.2	0.2	0.2	-0.1	-0.1	0
35	35	0.8	0.8	0.6	0.7	0.7	0.1	0.1	-0.1	0
36	36	0.4	0.2	0.3	0.4	0.4	0	-0.2	-0.1	0
37	37	0.8	0.8	1	1	0.9	-0.1	-0.1	0.1	0.1
38	38	0.6	0.3	0	0.3	0.3	0.3	0	-0.3	0
39	39	0.6	0.5	0.4	0.7	0.5	0.1	0	-0.1	0.2
40	40	0.1	0.1	0.1	0.1	0	0.1	0.1	0.1	0.1

Fig. 5. Query specific results for Scientific Abstracts Trials. Evaluation measure: P@10. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus SA_LGD_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.

	Labels	SA_LGD_letor	SA_bc	SAsimpleLGD	SA_DPH_letor	Trec median	diff1	diff2	diff3	diff4
0	0	0.2714	0.3051	0.3092	0.2808	0.28062	-0.00922	0.02448	0.02858	0.00018
1	1	0.2181	0.3173	0.3541	0.2266	0.3484	-0.1303	-0.0311	0.0057	-0.1218
2	2	0.1864	0.3305	0.3192	0.2232	0.2429	-0.0565	0.0876	0.0763	-0.0197
3	3	0.1667	0.2222	0.2222	0.1111	0.1111	0.0556	0.1111	0.1111	0
4	4	0.296	0.2915	0.2915	0.2511	0.2152	0.0808	0.0763	0.0763	0.0359
5	5	0.2857	0.2946	0.3304	0.1964	0.1964	0.0893	0.0982	0.134	0
6	6	0.1558	0.1688	0.1532	0.1818	0.2	-0.0442	-0.0312	-0.0468	-0.0182
7	7	0.3673	0.3863	0.372	0.3768	0.346	0.0213	0.0403	0.026	0.0308
8	8	0.2857	0.3857	0.4	0.3857	0.4571	-0.1714	-0.0714	-0.0571	-0.0714
9	9	0.4463	0.4587	0.438	0.4711	0.4174	0.0289	0.0413	0.0206	0.0537
10	10	0.3373	0.4096	0.4819	0.3855	0.4337	-0.0964	-0.0241	0.0482	-0.0482
11	11	0.3286	0.3357	0.3786	0.3357	0.3357	-0.0071	0	0.0429	0
12	12	0.2121	0.1818	0.1515	0.1818	0.1515	0.0606	0.0303	0	0.0303
13	13	0.274	0.2877	0.2877	0.2603	0.2877	-0.0137	0	0	-0.0274
14	14	0	0	0	0	0.1154	-0.1154	-0.1154	-0.1154	-0.1154
15	15	0.263	0.3006	0.289	0.2977	0.2919	-0.0289	0.0087	-0.0029	0.0058
16	16	0.5201	0.5348	0.5311	0.5275	0.4579	0.0622	0.0769	0.0732	0.0696
17	17	0.1795	0.2436	0.3077	0.2179	0.2949	-0.1154	-0.0513	0.0128	-0.077
18	18	0.3947	0.4035	0.4474	0.386	0.3509	0.0438	0.0526	0.0965	0.0351
19	19	0.1449	0.1304	0.1449	0.087	0.058	0.0869	0.0724	0.0869	0.029
20	20	0.102	0.1224	0.102	0.1633	0.1429	-0.0409	-0.0205	-0.0409	0.0204
21	21	0.1136	0.1534	0.1534	0.1364	0.2045	-0.0909	-0.0511	-0.0511	-0.0681
22	22	0.1691	0.1618	0.1103	0.1838	0.2647	-0.0956	-0.1029	-0.1544	-0.0809
23	23	0.16	0.2	0.24	0.16	0.16	0	0.04	0.08	0
24	24	0.2308	0.3846	0.4359	0.1282	0.2564	-0.0256	0.1282	0.1795	-0.1282
25	25	0.4071	0.4664	0.4625	0.4625	0.4229	-0.0158	0.0435	0.0396	0.0396
26	26	0.5345	0.5517	0.5517	0.5172	0.5345	0	0.0172	0.0172	-0.0173
27	27	0.1923	0.2308	0.2692	0.2198	0.2582	-0.0659	-0.0274	0.011	-0.0384
28	28	0.3625	0.4094	0.4156	0.3781	0.3812	-0.0187	0.0282	0.0344	-0.0031
29	29	0.3235	0.4118	0.4706	0.4706	0.3824	-0.0589	0.0294	0.0882	0.0882
30	30	0.6	0.6333	0.7	0.4	0.5	0.1	0.1333	0.2	-0.1
31	31	0.1875	0.25	0.1875	0.25	0.1875	0	0.0625	0	0.0625
32	32	0.16	0.24	0.24	0.28	0.2	-0.04	0.04	0.04	0.08
33	33	0	0	0	0	0	0	0	0	0
34	34	0.1972	0.1972	0.169	0.1972	0.1972	0	0	-0.0282	0
35	35	0.4	0.4238	0.3952	0.4476	0.3429	0.0571	0.0809	0.0523	0.1047
36	36	0.2857	0.2857	0.2449	0.2857	0.2857	0	0	-0.0408	0
37	37	0.5667	0.6267	0.6133	0.5933	0.5933	-0.0266	0.0334	0.02	0
38	38	0.3281	0.2969	0.2344	0.3438	0.25	0.0781	0.0469	-0.0156	0.0938
39	39	0.3484	0.3484	0.3484	0.3871	0.3484	0	0	0	0.0387
40	40	0.125	0.125	0.125	0.125	0	0.125	0.125	0.125	0.125

Fig. 6. Query specific results for Scientific Abstracts Trials. Evaluation measure: Rprec. The first row, labeled as 0, is an average over all queries. Column labeled as diff1 is equal to Trec median minus SA_LGD_letor. The following columns labeled as diff represent the difference between Trec median and the following runs.