Passage Based Answer-Set Graph Approach for Query Performance Prediction

Ghulam Sarwar
National University of Ireland, Galway
Galway, Ireland
g.sarwar1@nuigalway.ie

Colm O'Riordan
National University of Ireland, Galway
Galway, Ireland
colm.oriordan@nuigalway.ie

ABSTRACT

Approaches involving the use of post-retrieval information for a given query have been adopted in a variety of ways in the past for query performance prediction (QPP) tasks. Researchers have utilized information via document retrieval as well as passage retrieval approaches for QPP. We present a novel approach of representing the top returned passages (answer-set) as a graph where each node represents a passage and an edge weight indicates the similarity score between these passages. By examining the answer-set graph we developed new predictors that utilizes graph features such as cohesion and minimum spanning tree. Based on the empirical evaluation, we show that our answer-set graph predictors are very effective and perform even better (for Cranfield and Ohsumed Collection) than the current state-of-the-art QPP approaches.

KEYWORDS

passage retrieval, passage similarity graph, Query Performance Prediction, weighted graph, query difficulty, post-retrieval prediction

ACM Reference Format:
Ghulam Sarwar and Colm O'Riordan. 2021. Passage Based Answer-Set Graph Approach for Query Performance Prediction. In *Australasian Document Computing Symposium (ADCS '21), December 9, 2021, Virtual Event, Australia.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3503516.3503534

1 INTRODUCTION

Finding a way to distinguish between an easy and a difficult query is a challenging task for Information Retrieval (IR) systems. Several indicators have been developed in the past which are categorised into two main approaches: Pre-retrieval and Post-retrieval. Pre-retrieval indicators focus on the features of the query terms [11], whereas the Post-retrieval indicators take the answer set for a given query into account[15]. The main purpose of all QPP indicators is to differentiate and formulate strategies to improve the performance of the IR system with the gained knowledge of a given query. To check the effectiveness of a given predictor, a correlation of average precision (which depicts the performance of the system) is used against the resulting score of the predictor.

ADCS '21, December 9, 2021, Virtual Event, Australia

ACM ISBN 978-1-4503-9599-1/21/12…$15.00
https://doi.org/10.1145/3503516.3503534

In general, the QPP task is studied by extracting the information from a document retrieval system. However, there is very little work done so far where researchers have utilized passage information as a measure for predicting query performance. The passage based QPP is either used in QA tasks [10, 26] or a passage evidence approach like MaxPassage [7, 40] is combined with document score to formulate a query performance predictor[37]. Utilizing passages rather than documents is beneficial especially where the documents in the collection are diverse in nature [48]. Problems arising due to the presence of bias in documents used in document-based relevance can be potentially overcome with the usage of passages [22]. Passages are also effective in reducing the additional noise caused by the topic drift within the document as well as computationally less expensive to compare. In this paper, we use the term 'answer-set' to represent the top ranked k passages as a retrieved list for a given query q where k is a parameter that can be changed as per the need. We used a completely novel approach by composing an answer-set graph where each node represents a passage and their similarity score with each other forms an edge between them. The structure of this graph captures valuable information about the nature of the answer-set. Our approach is based on the cluster hypothesis [28], which states that similar documents within a cluster represent the same relevance to information need [50]. Let us consider two extreme scenarios where an easy query returns a coherent answer-set or a difficult query exhibiting ambiguity results in a diverse answer-set covering a large number of potentially unrelated topics. The motivation behind considering the graph of the answer-set is that by considering the cluster hypothesis, we would see a significant difference in the graphs of the aforementioned scenarios; in the first case, we would expect to see one large cluster of related topics with edges with a relatively high similarity score. For the second scenario, we would expect to see several small clusters (highly connected within) that are loosely connected to each other with edges of lower similarity weight. In order to check how well connected are these answer sets, different graph features can be used that determines the quality of the graph and the relatedness of its nodes. Furthermore, if a query is 'difficult' due to certain terms dominating the query, or certain terms having several meanings, then we would expect different documents to be returned by virtue of different aspects in the query, and hence a graph of these documents would exhibit substantially different features. As we are using the relation between passages within the answer-set (top k passages), our approach is computationally economical compared to applying the similar approach and generating graph by using the documents (due to the small text size i.e. passages compare to full-length documents), and provide flexibility in terms of formulating the strength of edges within the graph by utilizing numerous similarity measures (term frequency, semantic approaches etc.) as per the need.

Consider an undirected weighted graph $G(V, E)$ where each node $n_i \in V$ represents a passage p_x. An edge $e_{x,y} \in E$ represents the similarity (cosine similarity) between vertices x and y. The weight of an edge represents the strength of similarity between passage nodes p_x and p_y. Initially every node in the answer-set graph is connected to all other nodes, forming a complete graph. In this paper, we used an edge count *ec* as a threshold representing the top number of edges (with highest weight) to form a sub graph where we discard all the remaining edges with weights below *ec*.

In this paper, we present a novel approach that utilizes the existing graph measures and apply it to answer-set of retrieved passages for Query Performance Prediction task. We observed that answer-set graph approaches are comparable and even perform better than the existing QPP approaches.

This paper's outline is as follows: related work in QPP and passage retrieval using graphs is discussed in section 2. We detail our answer-set graph-based prediction approach in section 3. Experimental setup and results are discussed in section 4, and section 5 concludes the paper and outline future work.

2 RELATED WORK

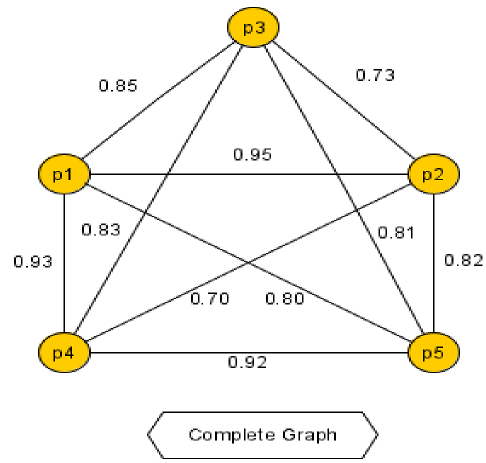
For the Query Performance Prediction task, the main objective is to estimate the effectiveness of retrieved results against a given query without any relevance judgements [8, 35]. The main body of work in QPP is divided into two areas. Pre-retrieval predictors consider query and the corpus information before the retrieval process [11, 16, 17, 31, 52]. Post-retrieval predictors also include the ranked result list of retrieval process [12, 15, 44, 45, 51, 53]. Our approach falls under the post-retrieval predictors. Major work in QPP post-retrieval work has been more focused on using the document retrieval evidence and very little work has been done to use the passage based information for the QPP task. Passage retrieval has been used in the past to improve the retrieval performance [1, 2, 7, 20, 40] thoroughly. Callan [7] presented the passage boundary approaches like bounded passages and overlapping window-based approach. Other approaches like text-tiling and usage of language models were also studied to extract passages[18, 30]. However, overlapped and non-overlapped window-based approaches are most commonly used to extract passages [7, 54].

Roitman [37] used passage information for QPP task. Similar to our work, they also hypothesized that the passage evidence from a retrieved results can provide a valuable evidence with respect to the effectiveness of the retrieval. They used the max passage approach [7] to obtain the best passages from the retrieved document list. However, rather than retrieving documents (and extracting passage information from them), we index and retrieved passages as answer-set to measure the effectiveness of retrieval because the top ranked results in the answer-set provide better contextual information around the topic of the query. Moreover, A. Khwileh et al.[24] utilized QPP to select the suitable passage type against each query. They also introduced a new QPP approach called Weighed Expansion Gain (WEG) and compared it with existing approaches like *WIG* [53] and *NQC* [45] to select the best passage based evidence approach [7, 18, 30] for a given query. Once the most suited passage is selected, they formulated an adaptive Query Expansion (QE) technique to improve the performance of the system. Semantic-based post retrieval predictors have also been presented in the past for QPP task [9, 19, 23]. Recently, Jafarzadeh et al. proposed a semantic-based approach that utilizes the graph model which exploits the topological features (represented by document entities) to capture the semantic similarities between a given query and top-returned documents. We used cohesion (defined in Section 3.1) of the answer-set based on the retrieved passages. They employed the similar approach to measure the cohesion of the answer-set but they considered documents and not the passages. Also, their approach to measure the similarity between the top returned documents was build upon a graph-based semantic model and not a traditional approach like cosine similarity (adopted in this paper). Other than cohesion, they also introduced a semantic based query-drift predictor (similar to the model presented by Shtok et al. [45]). They have shown that their proposed predictors were effective compared to other existing QPP approaches.

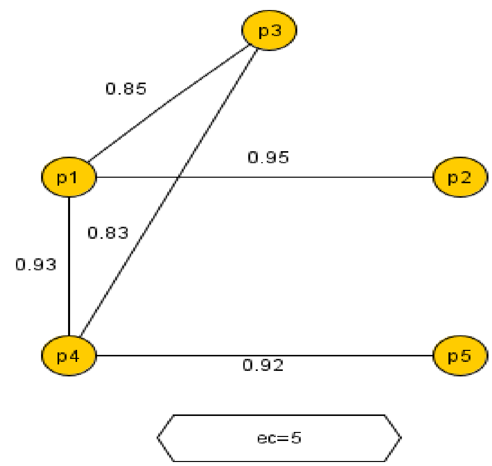
Graphs have been used in the past to represent text for ad-hoc information retrieval tasks [4, 39, 46, 47]. For question answer (QA) task, Li et al. [29] introduced a graph-based model to re-rank the retrieved results (passages) by using their similarity with each other. We generated the passage graph in a similar way and then apply an edge count as a threshold on it to measure different features of the graph for QPP task. Similarly, graph approaches like the HITS algorithm [25] or PageRank [5] have also been adopted for passage retrieval by Dkaki et al. [14] where they identified the relevance between passage by considering the related terms that are shared amongst the returned passage. As our intuition of utilizing the similarity between passage is motivated by the cluster hypothesis, a learning to rank approach based on a similar intuition has recently been employed by Eilon et al. [43] for passage retrieval task. They have also shown that cluster hypothesis holds for returned documents as well as for returned passages against a given query.

3 METHODOLOGY

This section introduces our passage answer-set graph approach where we discuss different graph measures for predicting the query performance. We represent each document as a set of pseudo-documents (passages) i.e. $d' = \{p_1, p_2, \ldots p_n\}$. We assume that the set of all documents (corpus) C, a query q and retrieval approach (R) remain constant. The main purpose of our work is to utilize passage evidence from the answer-set and formulate a predictor to evaluate the quality of the ranked results given by the retrieval approach R. We generated a weighted undirected graph $G = (V, E)$ where each node p_x represents a passage from the answer-set and E is an edge-weight function that is based on the similarity between passage nodes p_x, p_y. As mentioned previously that at first, the undirected graph is constructed by taking all the passages within the answer-set as nodes and calculating the similarity between them as edges; formulating a complete graph and then an edge count (ec) threshold is applied to generate a subgraph. An undirected complete graph of 100 nodes will have a maximum of $n(n-1)/2$ i.e. 4950 edges. So if $ec = 250$, the remaining edges after the top 250 edges from the aforementioned graph will be discarded. We used



(a) Passage Answer-Set Complete Graph



(b) Answer-Set Graph; Edge Count (ec) =5

Figure 1: Answer-Set Graph Before and After Applying Edge Count (ec) as a Threshold

$ec = \{100, 250, 500, 1000\}$ and the motivation of setting different ec value is to remove the noise from the answer-set graph and only consider the relationship between passages that are forming close clusters based on cluster hypothesis. Figures 1(a) and 1(b) illustrate an example of an initial answer-set graph constructed for a given query q with top 5 returned passages ($k = 5$) and the state of the graph after the edge count ec parameter is applied i.e. $ec = 5$.

Below we explain two different graph measures that we adopted with the passage answer-set graph for the QPP task.

3.1 Cohesion

In graph structure, cohesion is used to measure the interconnection between the nodes of the graph. One way to capture cohesion in a given text is to look at the term distribution [36, 49] or by analysing different clusters generated from a returned documents graphs [21, 32]. Recently Sarwar et.al [41] defined cohesion by considering the similarity of passages within the same document. Rather than taking only the passages in a returned list from the same document (Sarwar et.al approach), we applied a similar approach on the passage answer-set that measured the inter-connectivity of the answer-set. We take cohesion as a divergence of topics within the answer-set. If the returned answer-set of passages discuss a similar topic, the cohesion of the answer-set will be high and vice-versa. As per the cluster hypothesis, the nodes that belong to the same cluster convey similar relevance to the information need [3]. For an easy query, as the returned documents will largely be relevant, therefore, according to the cluster hypothesis these documents will be similar to each other. Hence, the cohesion score can be an effective measure to distinguish between the easy and hard query and can be used as a predictor of query performance.

For an undirected graph, the size of a graph is its number of edges $|E|$ where it contains n(n-1)/2 maximum number of edges. We denote the cumulative weight of the graph as $W = \sum_{\forall e} wt$ where wt is an edge weight and total number of edges are defined as $|E|$. The cohesion of an answer-set graph is denoted as follows:

$$C = \frac{W}{|E|} \tag{1}$$

3.2 Minimum Spanning Tree

The graph of the answer-set contains a node for every passage returned and an edge between each pair represents the similarity (above the ec threshold). The structure of this graph should capture some valuable information regarding the nature of the answer set.

One measure often used in graph theory is to find the minimal network of a given network. The resulting graph should be connected i.e. involve all nodes of the original network. This is been explored in traffic networks [13], communication networks [6] to find the fundamental 'cheapest' graph with the costly redundant edges removed. Prim[34] and Kruskal's[27] are two well-known algorithms that achieve this. The algorithms produce what is known as a Minimal Spanning Tree (MST). We can apply this idea to the graph representing the answer set. We are interested in finding the minimum spanning trees where the lower the edge, the lower the similarity.

For a set of loosely related set of topics, we will have low similarity between many of the nodes in the answer set; this will lead to a low weighted minimum spanning tree. One could argue that for easy queries we would expect larger clusters of related topics with high similarity scores. If that is the case, even the nodes with a relatively lower similarity score in the graph would have a higher edge weight compare to a graph formed with a hard query which will form loosely connected clusters in the answer-set graph. MST illustrates the overall connectedness of a graph, which is why it can exhibit different characteristics of the answer-set. By taking the lower score edges to generate MST, we are measuring a summary statistic of the answer-set graph which can capture the nature of the retrieved responses for a given q. Thus, we used MST as a predictor for the query performance task. We used Kruskal's implementation of MST and calculated the MST score by adding all the weights

Table 1: Document Collections Characteristics

	# Docs	# Passages	# Queries	Window Size
WebAp	6399	146000	150	250 words
Cranfield	1400	7722	225	30 words
Ohsumed	233,445	1404440	97	30 words

assigned to each edge of the generated minimum spanning tree from the answer-set graph.

4 EVALUATION

4.1 Experimental Setup

We performed our experiments using three different test collections. The characteristics of these test collections is specified in Table 1. To identify the passage boundaries, we employed a half-overlapping fixed-length window size approach [7]. To index all the passages, we used SOLR [42] as our retrieval model, which is a Lucene[1] based IR system that uses a vector space model with a weighting scheme from the family of tf-idf weighting schemes. For each test collection, we predicted the performance of each query based on its top 100 highly ranked documents and passages (retrieved from SOLR).

To extract the graph features specified in section 3, we used a sub-graph based on the top edge count *ec* value. We used *ec* ={100, 250, 500, 1000} to create four different graphs. One drawback of keeping more edges in the graph is that it increases the level of noise. Similarly by lowering the amount of edges, we may lose the context and some important features within the answer-set. Moreover, graph features such as MST for an easy query with higher number of edges will have a lower MST score, which in result form a tree with nodes that are not as strongly connected to each other. Hence, we will lose the information about the highest correlated nodes with in the graph. Therefore, setting the higher threshold for edge count will overall reduce the correlation performance. In this paper, we present our correlation results with $ec = 250$ as we found it to be the optimal one. We will further discuss the results on all four graphs with different edge count size in Experimental Results section. We employed Pearson's correlation as an evaluation measure since it is commonly used to evaluate the query performance prediction. It is computed by considering the correlation between the actual document level retrieval performance i.e. average precision (*AP*) for a given query *q* and the output generated for *q* by the specified query performance predictor.

4.2 Baselines

Oftentimes in IR, it is a common scenario where the retrieved documents do not pertain to the information required for a given query *q* but it still shows a high similarity to *q*. One way to quantify the variation of this similarity for *q* in the ranked list is to use the standard deviation of the query-document similarity scores [12, 38]. We choose our first baseline as a standard deviation on a fixed cut-off point i.e. at top 100 document σ_{100} as it has been shown as an effective predictor [12, 33]. We also considered other popular post-retrieval QPP approaches like Normalized Query Commitment(NQC) and Weighed Information Gain(WIG). Similar to the normal standard deviation approach like σ_{100}, NQC also utilise standard deviation by further normalizing it with the corpus (all documents in the collection) score. The motivation in NQC is to estimate the query-drift [45] and use that as a predictor for query performance. Furthermore, WIG measures the difference between the average score of top-ranked documents and of the corpus. For WIG and NQC, we generated the corpus by concatenating the content of all the documents (except any tags, boundary information etc.). So far, all these QPP approaches used document retrieval information. Approaches like $NQC(psg)$ and $WIG(psg)$ have been explored lately that utilize the same fundamental principles while using passage information [37]. Here we also include the σ_{100}(psg) to show a clear comparison of existing document vs passage-based approaches that have been explored in the past. To calculate the AP for the passage based approaches, we used the standard Max passage [7] estimation approach with top 100 ranked documents.

4.3 Experimental Results

In this section we present the experimental results to illustrate the evaluation of different QPP approaches. Table 3 summarizes the results of existing approaches against our passage answer-set graph methods (Cohesion and MST). The results are reported for three different test collections. First, we observe that the document level QPP approaches along with their passage based equivalent methods exhibit differing performance results. For example, we can see that for the WebAp, the passage-based counterparts of WIG i.e $WIG(psg)$ give a much better correlation result compared to its document level equivalents. Similarly, for all the test collections, the passage-based σ_{100}(psg) performed better than the document level NQC and WIG methods, which shows the significance of the usage of passage based evidence in QPP tasks.

Furthermore, we can see that our newly proposed graph-based approach give a positive correlation for all the test collections, which supports our intuition of exploiting the answer-set of passage graph for predicting the query performance. Our MST approach outperformed all the other document and passage-based baselines for the Cranfield and the Ohsumed collection. The standard deviation σ_{100} approach at document level performed the best for the WebAp, yet our MST and cohesion approach outperformed the NQC and WIG baselines. This indicates the significance of our answer-set passage graph approaches. The standard deviation is calculated based on the spread of query document similarity scores on a given ranked list. For the webAp collection, we noticed that there was a significant difference (calculated via student's t-test) between the Sd scores of the top 5 hard queries and easy queries. For MST we saw the difference as well, however it wasn't as strongly significant as the SD scores. One possible reason for it is that in our MST approach we are focused more on the relevance of the returned passages with themselves (passages in the answer-set) and not the query itself. Our MST work is based on the cluster hypothesis where we hope to find different MST score for easy and hard queries. One limitation here for MST can be that in some situations even if the returned passages are irrelevant, their answer-set graph can still exhibit a high MST score (similar to easy query) because those irrelevant passages are strongly connected to each other. And

[1] http://lucene.apache.org/

Table 2: Pearson Correlation Comparison of Passage Graph Approaches at Different Edge Count (ec). Boldface: Best Result Per test collection against multiple ec.

	Cranfield				WebAP				Ohsumed			
	100	250	500	1000	100	250	500	1000	100	250	500	1000
Cohesion	**0.09**	0.03	0.007	-0.005	0.14	**0.19**	0.17	0.15	**0.34**	0.24	0.18	0.13
MST	0.21	**0.25**	0.18	0.10	0.15	**0.24**	0.22	0.18	0.45	**0.48**	0.41	0.20

we saw that this was the case for WebAp collection; Therefore, SD approach σ_{100} gave better results compared to the MST approach.

In Table 3 we reported our results for edge count *ec* set to 250 top-scored edges in the answer-set graph. Adding more edges i.e. passage information can cause extra noise, which could result in the reduction of prediction performance. In Table 2 we show the results for different edge count to illustrate the implication of reducing or increasing more passage information on the overall performance. We can see that for MST when the *ec* was increased from 100 to 250, the performance is always improved against all test collections. Similarly, when we go above 250 edges in our graph, the performance tends to decrease gradually for 500 and 1000 edges. This shows that adding more passage information causes noise and as a result, reduce the predictor's performance.

Table 3: Prediction via Pearson Correlation for document and passage QPP baselines against Cohesion and MST. Boldface: best result per test collection.

	Cranfield	WebAp	Ohsumed
σ_{100}	-0.07	**0.63**	0.48
NQC	-0.113	0.17	-0.09
WIG	-0.32	-0.13	-0.07
σ_{100}(psg)	0.082	0.30	0.46
$NQC(psg)$	-0.074	0.062	-0.09
$WIG(psg)$	-0.30	0.19	-0.08
Cohesion	0.03	0.19	0.24
MST	**0.25**	0.24	**0.48**

5 CONCLUSION AND FUTURE WORK

In this paper, we introduced a novel approach to predict the query performance by using the features of the answer-set passage graph. We explained two graph features i.e. Cohesion and Minimum Spanning Tree (MST) and calculated their respective scores and used them as a measure to find its correlation against the retrieval performance (AP) for QPP task. We also compared different baselines that utilize not just the document relevance but also the passage information for QPP. The results show that our MST approach outperforms the existing QPP approaches against the Cranfield and the Ohsumed test collection. The MST approach also outperforms other popular methods like WIG and NQC against WebAp test collection as well. Our Cohesion and MST approach shows a positive correlation for all test collections whereas other document and passage-based counterpart exhibits mixed performance results. It signifies that graph features of the passage answer-set can be a useful measure for Query performance prediction. Moreover, features like cohesion and MST from generated graph of passage answer-set exhibits the connectedness of result list against a given query. In a scenario where the user wants to visualize the differentiation between the returned set in terms of topic drift or uniqueness, our approach can be useful there as well.

Our future work would involve applying the same approach to bigger size test collections such as GOV2, ROBUST04, MSMARCO, and ACQUAINT etc. and compared them against the state of the art approaches. We also intend to use the answer-set graph features to visualize the result and apply different clustering methods to find better ways to show the results to users based on query characteristics (easy vs hard query). Furthermore, in this paper we used the standard cosine similarity as a measure to define the edge weight between the graph nodes (passages). In future, we aim to extend our approach by applying other semantic based similarity measures such as, entity-based, word embeddings (BERT, Word2vec), or topic modelling (LDA) etc. to better capture the similarity of passages.

REFERENCES

[1] Qingyao Ai, Brendan O'Connor, and W Bruce Croft. 2018. A Neural Passage Model for Ad-hoc Document Retrieval. In *European Conference on Information Retrieval*. Springer, 537–543.

[2] Michael Bendersky and Oren Kurland. 2008. Utilizing passage-based language models for document retrieval. In *European Conference on Information Retrieval*. Springer, 162–174.

[3] David C. Blair. 1979. Information Retrieval, 2nd ed. C.J. Van Rijsbergen. London: Butterworths. *Journal of the American Society for Information Science* (1979).

[4] Roi Blanco and Christina Lioma. 2012. Graph-based term weighting for information retrieval. *Information retrieval* 15, 1 (2012), 54–92.

[5] Vincent D Blondel, Anahí Gajardo, Maureen Heymans, Pierre Senellart, and Paul Van Dooren. 2004. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM review* 46, 4 (2004), 647–666.

[6] Frank T Boesch, Frank Harary, and Jerald A Kabell. 1981. Graphs as models of communication network vulnerability: Connectivity and persistence. *Networks* 11, 1 (1981), 57–63.

[7] James P Callan. 1994. Passage-level evidence in document retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. Springer-Verlag New York, Inc., 302–310.

[8] David Carmel and Elad Yom-Tov. 2010. Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services* 2, 1 (2010), 1–89.

[9] Adrian-Gabriel Chifu, Léa Laporte, Josiane Mothe, and Md Zia Ullah. 2018. Query performance prediction focused on summarized letor features. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1177–1180.

[10] Charles LA Clarke, Gordon V Cormack, Thomas R Lynam, and Egidio L Terra. 2008. Question answering by passage selection. In *Advances in Open Domain Question Answering*. Springer, 259–283.

[11] Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft. 2002. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. 299–306.

[12] Ronan Cummins, Joemon Jose, and Colm O'Riordan. 2011. Improved query performance prediction using standard deviation. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 1089–1090.

[13] Hristo Djidjev, Gary Sandine, Curtis Storlie, and Scott Vander Wiel. 2011. Graph based statistical analysis of network traffic. In *Proceedings of the Ninth Workshop on Mining and Learning with Graphs*.

[14] Taoufiq Dkaki, Josiane Mothe, and Quoc Dinh Truong. 2007. Passage retrieval using graph vertices comparison. In *2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System*. IEEE, 71–76.

[15] Guglielmo Faggioli, Oleg Zendel, J Shane Culpepper, Nicola Ferro, and Falk Scholer. 2021. An enhanced evaluation framework for query performance prediction. In *European Conference on Information Retrieval*. Springer, 115–129.

[16] Claudia Hauff, Djoerd Hiemstra, and Franciska de Jong. 2008. A survey of pre-retrieval query performance predictors. In *Proceedings of the 17th ACM conference on Information and knowledge management*. 1419–1420.

[17] Ben He and Iadh Ounis. 2004. Inferring query performance using pre-retrieval predictors. In *International symposium on string processing and information retrieval*. Springer, 43–54.

[18] Marti A Hearst. 1997. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics* 23, 1 (1997), 33–64.

[19] Parastoo Jafarzadeh and Faezeh Ensan. 2022. A semantic approach to post-retrieval query performance prediction. *Information Processing & Management* 59, 1 (2022), 102746.

[20] Man-Hung Jong, Chong-Han Ri, Hyok-Chol Choe, and Chol-Jun Hwang. 2015. A Method of Passage-Based Document Retrieval in Question Answering System. *arXiv preprint arXiv:1512.05437* (2015).

[21] Vasileios Kandylas, S Phineas Upham, and Lyle H Ungar. 2008. Finding cohesive clusters for analyzing knowledge communities. *Knowledge and Information Systems* 17, 3 (2008), 335–354.

[22] Marcin Kaszkiel and Justin Zobel. 2001. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology* 52, 4 (2001), 344–364.

[23] Maryam Khodabakhsh and Ebrahim Bagheri. 2021. Semantics-enabled query performance prediction for ad hoc table retrieval. *Information Processing & Management* 58, 1 (2021), 102399.

[24] Ahmad Khwileh, Andy Way, and Gareth JF Jones. 2017. Improving the reliability of query expansion for user-generated speech retrieval using query performance prediction. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, 43–56.

[25] Jon M Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)* 46, 5 (1999), 604–632.

[26] Eyal Krikon, David Carmel, and Oren Kurland. 2012. Predicting the performance of passage retrieval for question answering. In *Proceedings of the 21st ACM International Conference on Information and Knowledge management*. 2451–2454.

[27] Joseph B Kruskal. 1956. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society* 7, 1 (1956), 48–50.

[28] Oren Kurland. 2014. The cluster hypothesis in information retrieval. In *European Conference on Information Retrieval*. Springer, 823–826.

[29] Xin Li and Enhong Chen. 2010. Graph-Based Answer Passage Ranking for Question Answering. In *Computational Intelligence and Security (CIS), 2010 International Conference on*. IEEE, 634–638.

[30] Xiaoyong Liu and W Bruce Croft. 2002. Passage retrieval based on language models. In *Proceedings of the eleventh international conference on Information and knowledge management*. ACM, 375–382.

[31] Josiane Mothe and Ludovic Tanguy. 2005. Linguistic features to predict query difficulty. In *ACM Conference on research and Development in Information Retrieval, SIGIR, Predicting query difficulty-methods and applications workshop*. 7–10.

[32] Raúl Abella Pérez and José Eladio Medina Pagola. 2010. An incremental text segmentation by clustering cohesion. *HaCDAIS 2010* (2010), 65.

[33] Joaquín Pérez-Iglesias and Lourdes Araujo. 2010. Standard deviation as a query hardness estimator. In *International Symposium on String Processing and Information Retrieval*. Springer, 207–212.

[34] Robert Clay Prim. 1957. Shortest connection networks and some generalizations. *The Bell System Technical Journal* 36, 6 (1957), 1389–1401.

[35] Fiana Raiber and Oren Kurland. 2014. Query-performance prediction: setting the expectations straight. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. 13–22.

[36] Benjamin Renoust, Guy Melançon, and Marie-Luce Viaud. 2013. Measuring group cohesion in document collections. In *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01*. IEEE Computer Society, 373–380.

[37] Haggai Roitman. 2018. Query performance prediction using passage information. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 893–896.

[38] Haggai Roitman, Shai Erera, and Bar Weiner. 2017. Robust standard deviation estimation for query performance prediction. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*. 245–248.

[39] François Rousseau and Michalis Vazirgiannis. 2013. Graph-of-word and TW-IDF: new approach to ad hoc IR. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM, 59–68.

[40] Ghulam Sarwar, Colm O'Riordan, and John Newell. 2017. Passage Level Evidence for Effective Document Level Retrieval. In *Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*. 83–90. https://doi.org/10.5220/0006502800830090

[41] Ghulam Sarwar. and Colm O'Riordan. 2021. A Graph-based Approach at Passage Level to Investigate the Cohesiveness of Documents. In *Proceedings of the 10th International Conference on Data Science, Technology and Applications - DATA,*. INSTICC, SciTePress, 115–123. https://doi.org/10.5220/0010619101150123

[42] Dikshant Shahi. 2016. *Apache Solr*. Springer.

[43] Eilon Sheetrit and Oren Kurland. 2019. Cluster-based focused retrieval. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2305–2308.

[44] Anna Shtok, Oren Kurland, and David Carmel. 2016. Query performance prediction using reference lists. *ACM Transactions on Information Systems (TOIS)* 34, 4 (2016), 1–34.

[45] Anna Shtok, Oren Kurland, David Carmel, Fiana Raiber, and Gad Markovits. 2012. Predicting query performance by query-drift estimation. *ACM Transactions on Information Systems (TOIS)* 30, 2 (2012), 1–35.

[46] Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1171–1181.

[47] Duangjai Thammasut and Ohm Sornil. 2006. A graph-based information retrieval system. In *2006 International Symposium on Communications and Information Technologies*. IEEE, 743–748.

[48] Jörg Tiedemann and Jori Mur. 2008. Simple is best: experiments with different document segmentation strategies for passage retrieval. In *Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering*. 17–25.

[49] Olga Vechtomova and Murat Karamuftuoglu. 2008. Lexical cohesion and term proximity in document ranking. *Information Processing & Management* 44, 4 (2008), 1485–1502.

[50] Ellen M Voorhees. 1985. The cluster hypothesis revisited. In *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*. 188–196.

[51] Oleg Zendel, Anna Shtok, Fiana Raiber, Oren Kurland, and J Shane Culpepper. 2019. Information needs, queries, and query performance prediction. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 395–404.

[52] Ying Zhao, Falk Scholer, and Yohannes Tsegay. 2008. Effective pre-retrieval query performance prediction using similarity and variability evidence. In *European conference on information retrieval*. Springer, 52–64.

[53] Yun Zhou and W Bruce Croft. 2007. Query performance prediction in web search environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 543–550.

[54] Justin Zobel, Alistair Moffat, Ross Wilkinson, and Ron Sacks-Davis. 1995. Efficient retrieval of partial documents. *Information Processing & Management* 31, 3 (1995), 361–377.