Investigating the Use of Semantic Relatedness at Document and Passage Level for Query Augmentation

Ghulam Sarwar* and Stephen Bradshaw*
Department of Information Technology, National University of Ireland, Galway, Ireland

Keywords: IR Theory and Practice, Query Expansion, Content Representation and Processing, Passage Level Retrieval and Evidence.

Abstract: This paper documents an approach that i) uses graphs to capture the semantic relatedness between terms in text and ii) augmenting queries with those terms deemed to be semantically related to the query terms. In building the graphs we use a relatively straightforward approach based on term locations; we investigate approaches that aid query improvement by capturing the semantic relatedness that is extracted at passage level as well as the complete document level. Semantic relatedness between is represented on a graph, where the terms are stored as nodes and the strength of their connection is recorded as an edge weight. In this fashion, we recorded the degree of connection between terms and use this to suggest possible additional words for improving the precision of a query. We compare the results of both approaches to a traditional approach and present a number of experiments at passage and document level. Our findings are that the approaches investigated achieve a competitive standard against a well known baseline.

1 INTRODUCTION

In natural language, the same word will often be used to confer different meanings, and when presented in different contexts can embody different concepts. In many information retrieval systems, queries tend to be short and comprise a few indicative terms. Employing additional processes can improve results but at a computational cost. Standard approaches involve assigning a value to each term and returning documents that score highly in relation to the query terms submitted. The most informative terms are typically those that feature highly in a document, but not across the corpus (Salton and Buckley, 1988). Many IR systems consider the frequency of terms and adopt a term independence assumption. In doing so, many potentially useful indicators in the text are often overlooked. Approaches that do attempt to incorporate additional inputs include, among others: part of speech tagging (POS) (Brill, 2000) probabilistic frequency (Blei et al., 2003) and semantic dependencies (Lund and Burgess, 1996). Capitalising on additional indicators found in the text can offset the adverse effect of polysemy in IR systems.

Different query expansion approaches have been used in the past. One state of the art approach is to move the query towards the terms that are most related to it while keeping it away from the terms that could result in decreasing the performance of the system. This approach was first introduced by Rocchio (Rocchio, 1971). However, in Rocchio's method, one major concern is the problem of query drift. The expanded query might contain terms that could appear frequently in the documents but it does not accurately capture the search topic; hence an improperly expanded query is formed, that leads toward the poor performance. Similarly, If a query contains a word that has many different usages in the corpus, identifying the instance that relates to their information need can be difficult. In this paper, we propose the use of a graph approach to capture the semantic dependencies of terms and use those findings to reformulate the query. With the graph approach, while considering the relevant documents, we can pick the number of words that we find are the most suitable to expand the query. This is very beneficial in terms of understanding the behavior of the system whilst the selection of expanded terms in the query. In section 4 we discuss this issue in more detail.

We investigate the use of semantic dependencies to see if appropriate additional terms can be identified and used in augmenting the queries. To ensure that our approach is robust, we will investigate varying

*Both authors contributed equally to this paper

Sarwar, G. and Bradshaw, S.
Investigating the Use of Semantic Relatedness at Document and Passage Level for Query Augmentation.
DOI: 10.5220/0006935902370244
In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2018) - Volume 1: KDIR*, pages 237-244
ISBN: 978-989-758-330-8

a number of parameters to our model; as well as by exploring different document preprocessing steps. Another important feature we employed in this paper is the usages of passages over that of entire documents. The motivation behind the adoption of passages for query expansion was to hopefully reduce noise that could occur lead to the topic drift in the resulting query. Moreover, retrieving the indexed passages over documents from the IR engine shortened the amount of text to be processed in our graph approach.

The paper outline is as follows: section 2 presents an overview of the previous work in query expansion with the use of different language modeling approaches and also highlights how passage-level evidence is used to extract the semantic relatedness and query expansion to improve the effectiveness of an IR system. Section 3 gives an overview of the methodology used, that outlines the details of the graph-based approach and its application with the document and passage level retrieval. It also highlights the different similarity functions employed to extract the top passages for query augmentation with the brief overview of the test collection used for the experiments. Section 5 reports the experimental results obtained while comparison of query augmentation approaches at document and passage level. Lastly, we provide the brief overview of the main conclusion and outline future work.

2 RELATED WORK

Many approaches have been explored in previous research to augment user queries to better reflect the user's intended information needed, thereby improving the accuracy of the system. One of the original approaches was proposed by Rocchio (Rocchio, 1971) which attempts to augment a query to better distinguish between relevant and non-relevant documents. An ideal query is one that returns all of the relevant documents while avoiding all of the irrelevant ones. To estimate this ideal query, Rocchio suggests an iterative feedback process whereby the positive feedback and negative feedback provided by a user is used to guide the query modification. To achieve this, the author suggests giving each word a weight relative to its presence in either the relevant or irrelevant document set.

Zhang et al (Zhang et al., 2005) attempt to improve upon search results using metadata found within the corpus. They designated two features found within the documents as indicators of how to rank the documents; **information richness** and **diversity**. Information richness is the extent to which a document relates to a particular topic, and diversity is the number of topics found within the corpus. In addition to determining these scores for the documents, they assign each document to a graph where the node represents the document and the surrounding nodes are determined by the inter similarity of the documents. Using this approach they improved the overall ranking of information gain and diversity by 12% and 31% respectively.

Hyperspace Analogue to Language (HAL) was proposed by Lund and Burgess in a theoretical analysis on the concept of capturing interdependencies in terms (Lund and Burgess, 1996). In this work, they applied a window of 10 to their document corpus and measured the co-occurrence of terms. Yun et al (Yan et al., 2010) use an approach that is relatable to HAL and apply it to three TREC datasets for query augmentation. They identify the drawbacks to a standard bag of word approaches and apply HAL to capture the semantic relationship between terms. Additionally, they model the syntactic elements of terms around a target event to better inform which words to use in augmenting a query.

Similarly, Kotov and Zhai (Kotov and Zhai, 2011) propose to use HAL to provide alternative senses for words. His dataset comprised of three TREC collections: AP88-89, ROBUST04 and ACQUAINT. He applied a mutual information measure and HAL to the dataset to ascertain the semantic strength between terms. He used these values and selected the strongest alternative candidate terms. Six participants were asked to input the queries as found in the respective datasets and were offered the option of using the alternative terms if they felt the search results were not sufficient. By combining these two methodologies he was able to improve the overall performance of the system. His conclusion was that 20 was the ideal window size when computing the HAL score.

Passage level retrieval has been used in the past for multiple purposes. Callan (Callan, 1994) has used passage level evidence to improve the document level ranking. Similarly Jong and Buckley (Jong et al., 2015), and Sarwar et al (Sarwar et al., 2017) followed the same concept and considered some alternative passage evidence i.e. passage score, the summation of passage score, inverse rank, and evaluation functions score etc. to retrieve the documents more effectively. Moreover, to choose the best passage boundaries several techniques have been used. Callan (Callan, 1994) proposed the bounded passages and overlapping window based approach. Similarly text-titling, usage of arbitrary passages and

language modeling approach was also considered (Hearst, 1997; Kaszkiel and Zobel, 2001; Liu and Croft, 2002).
In addition to that, blind relevance feedback(Mitra et al., 1998a) has been used before for automatic query expansion at the document and at passage level. Previously, it is shown that taking passages from the relevant documents for query expansion can be more effective than the document itself (Gu and Luo, 2004; Mitra et al., 1998b; Liu and Croft, 2002). Different query expansion approaches have been used in the past where the information extracted from passages was employed. Ferhat et al (Aydın et al., 2017) used two different methods to expand their queries for passage retrieval in the biomedical domain that can help to identify the protein interaction (PPI). At first, they used a supervised approach which uses the combination of term frequency-relevance frequency to identify the added terms. They subsequently used an unsupervised approach where they used a medical ontology to get the expanded terms. Similarly, for passage level retrieval, Wei Zhou et al (Zhou et al., 2007) used the domain-specific knowledge (Synonyms, Hypernyms, and Hyponyms) in the biomedical literature (information about concepts and their relationships in a certain domain) to improve the effectiveness of an IR system. Additionally, they used a variation of pseudo-feedback approach to add new terms in the query. The results show that utilizing the information from the domain knowledge leads to significant improvements.

3 METHODOLOGY

In classical IR, the documents are usually taken as single entities. However, an alternative approach has been proposed by Callan (Callan, 1994), which involves splitting the documents into several passages. This is done because a document may contain a highly relevant passage amongst large tracts of irrelevant text resulting in an overall poor relevance score. We consider passages as pseudo-documents where in general a passage could be defined as a sentence, number of sentences or a paragraph. Several identifiers, such as paragraph markings ($< p >$), new line tags($/n$) etc. can be used in the text to split the document into passages.

In this paper, we report our results using different query expansion approaches at both the document and pseudo-document level. Using evidence from relevance judgments present in the document collection is known as simulated feedback. To measure the retrieval performance and the quality of our query augmentation approach, Mean Average Precision (MAP) was used. We generated results using both the document and passage level evidence derived from different representation functions (discussed in section 3.3).
We use the Ohsumed collection as the test collection in our experiments. The dataset consists of a set of queries and an associated set of documents labeled as relevant for that document. The relevant documents for a query (limited to a fixed number) were converted to a vector space model representation (VSM) and stop words were removed. We then placed the VSM representations into a graph (described in more detail in this section). Using this graph we augmented the original query with additional terms as found there. We applied both our graph method and Song's variant on the HAL approach for query augmentation to each representation of the corpus and documented the results.

A directed graph was used to capture the semantic relatedness inherent in the text. Each word in the text was assigned a node in the graph. A sliding window of varying length was run over the text and co-occurrences were observed by incrementing the strength of the edge weight between the target term node and every proceeding term node within the range of the window size. The window size varied from one to ten. So when the window size was set to four the proceeding four terms edge weights from the target node were incremented by one. In this paper, We used the term 'graph approach' to refer to this process. Figure 1 is a graphical representation of the graph approach with the window size of 2, whereas every node (i.e term) in the graph is connected to 2 following and preceding terms. In addition to experimenting with the size of the window used, we varied the number of terms used to augment the query. Again we used values between one and ten to determine the number of additional terms to add for each term in query augmentation.

Figure 1: Recording the connection between terms.

3.1 Creating Passage Level Pseudo-documents

In this work we divided each document into number of passages with an overlapping window based passage boundary approach (Callan, 1994) and considered each passage as a pseudo-document i.e $d^{'} = \{p_1, p_2, \ldots p_n\}$. To augment the queries, we only used the relevant passages from the top 2000 results that were returned from the original queries of the Ohsumed collection. The number of relevant documents (and passages) for each query varies in number. We augmented the queries starting from level 1 to level 10. The level reflects the size of the co-occurring term window used on the returned passages. Depending on the approach used, a single term is selected and used to augment each word of the query. We explored different levels to determine what the optimal size in the sliding window should be. Figure 2 illustrates the basic flow of the complete system.

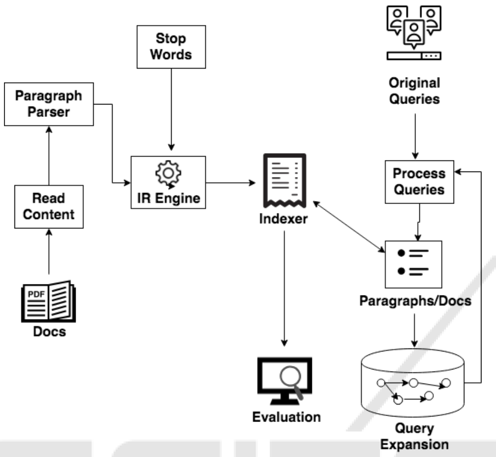


Figure 2: Basic flow diagram.

To consider documents at passage level, different passage representation functions can be used to re-rank the results as well as to filter the returned text for query augmentation. It is worth noting that the similarity between the passage or the document with the given query is interpreted here as the Lucene score. In Lucene, a vector space model is adopted with a weighting scheme based on the variation of tf-idf and Boolean model (BM) (Lashkari et al., 2009) to measure the similarity between the query and the index documents. We used that score to re-rank the documents based on the following passage similarity function.

- Max Passage (SF1): In this approach, the passage that has the highest similarity score from each document is chosen and then the results are re-ranked accordingly.

$$sim(d^{'}, q) = max(sim(p_i, q))$$

- Sum of passages (SF2): This approach differs from the SF1 approach because instead of just taking one passage with the highest score, top k passages are considered and then the similarity scores and the text is summed up and concatenated.

$$sim(d^{'}, q) = \sum_{i=1}^{k}[sim(p_i, q)]$$

We performed the experiments for $k = 1, 2, 3, 4, 5$ and in this paper, we reported results for $k = 2$ due to the better performance as compared to other values.

3.2 Rocchio Algorithm

We implemented the Rocchio formula is as follows.

$$\vec{q}m = a\vec{q} + \frac{\beta}{|D_r|}\sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \frac{\lambda}{|D_n|}\sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

Where $a\vec{q}$ represents the initial query, $\frac{\beta}{|D_r|}\sum_{\forall \vec{d}_j \in D_r} \vec{d}_j$ represents the value of the word as determined in the related document set, and $\frac{\lambda}{|D_n|}\sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$ are the values for each word from the non related results. The query will expand to a length equal to the number of all unique words present. The parameters α β and λ can be tuned before the process begins; we used 16, 8 and 0 respectively. Lambda was set to 0 as query syntax for Solr in Lucene does not support negative weights, and the assigned alpha and beta values were shown to produce the best results.

3.3 Test Collection and Experiment Setup

The Ohsumed collection was used in the experiments as it is substantial in size and initially performed poorly in terms of Mean Average Precision at document and passage level. The collection comprises a list of abstracts and titles from 270 Medline journal articles. It has 348,566 articles along with 106 queries in total. Of the 106 queries, 97 of them had relevant documents identified in the relevance judgment file. Therefore, we only used these queries to report the augmentation results in this paper. Furthermore, not all the articles in the collection contain the abstract. Thus, for the retrieval task, we indexed only the 233,445 documents to which abstract text was available.

To index all the documents and passages Solr 5.2.1[1] was used. Solr is a lucene[2] based IR system that uses a vector space model with the variation of TF-IDF and Boolean model for its weighting scheme. We used an overlapping window of 30 terms to divide the documents into passages which generated 1.4 million pseudo-documents. In this paper, the 'document' collection is referred to as collection 1 and the 'passage' collection is referred to

[1] http://lucene.apache.org/solr/5_2_1/index.html

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

as collection 2.
We used different passage selection functions to choose passages to augment the queries. We applied it to the returned passages and generated the related document text i.e. for max passage, by taking the highest scoring passage text from each document and for the sum of passages approached, by taking the two highest scoring passages from each document. And then later we used the score of these functions to re-rank our results to do the evaluation.

4 RESULTS

In this section, we present the results of two query expansion algorithms and compare them against the baseline approach. Scenarios using documents as the basic unit and scenarios using passage level evidence are considered. We will explain how the performance is changed when we perform query augmentation.
Four different representations of the data were used in the query expansion process:

1. Document Level (DL): Here we used the original document text that is retrieved from the relevant documents. In this one, we sent the queries to collection 1.
2. Passage Level (PL): Normal passages are used to expand queries without any further processing on them.
3. Max Passage Level (MPL): Once the passages are retrieved (just like in PL) we processed them and applied the SF1 to filter out the text for the expansion.
4. Sum of Passages Level (SOPL): It is similar to the MPL, but we use the other function i.e. SF2 in this scenario. Since we used $k = 2$ as a parameter, therefore each relevant document had a combined text of two top passages.

4.1 Varying Sized Windows

For the baseline, without using any expansion approach the recorded performance for a normal document level MAP was **13.50** and for the Max passage approach the MAP achieved was **13.09**. We used the Max passage similarity function to report our passage level results as was giving overall the best results after we applied the query expansion in comparison to document level results.

All the results discussed with the query expansion are compared by these baseline results. The results are also compared to classic Rocchio expansion. The MAP value recorded for Rocchio at passage level was **31.45**%.
Moreover, as shown in Table 1, the DL results give slightly better MAP as compared to the passage level. However, for the top results, passage level evidence was giving a better performance during our analysis (shown in figure 3(a) and 3(b)) when compared with the document level results, which reflects the significance of passages over the documents. In addition to that, by considering the variation of results at different levels, we took the best value in each query expansion approach i.e for the graph approach (i.e. EC) it is 18.16 and for Song's NEC it is 18.74. We compared the results at this position with baseline at different MAP levels to check the significance of the improved results. To do that, we performed the paired student's T-test for the MAP at 5 to 50 with the difference of 5 in each iteration and calculated the p-value. Both p-values are less than 0.05, therefore, for the Ohsumed collection, both graph-based algorithms significantly outperformed the baseline.

4.2 Increased Number of Terms

In Table 2, we show the results for the graph approach for all four representations of the document sets. There is a marked difference between the results for including terms to augment the query over the increase in the sliding window. The results improve and continue to improve in a linear fashion as the number of augmented terms is increased. This indicates that every newly added term had a positive influence on the overall results. Table 3 contains the results for Song's normalized HAL approach. Interestingly the graph approach makes noticeable improvements for the first three iterations, before increasing at a much slower rate. Song's approach only does so on the first two iterations.

The normal document length shows the strongest results for both approaches. Presumably, because there is more evidence from which to capture the semantic relatedness in the terms. MPL, SOPL, and PL all show results that are very near one another for both approaches. This indicates that both algorithms perform similarly when applied to smaller bodies of text. However, SOPL outperformed MPL and PL nearly at all levels, which supports our intuition behind using the passage representation function to isolate significant tracts of text. Song's approach, however, does not show the same level of improvement as graph approach on the larger datasets, suggesting that it does not capitalize on the extra information. We believe that the reason for this is that the normalization smooths out some of the

Table 1: MAP(%) of SF1 for the Ohsumed Collection at Different Query Expansion Approaches Using the **Varying Sized Windows** Approach.

Window Size	PL Edge Count	PL Normalised Edge Count	PL Rochio	DL Rochio	DL Edge Count	DL Normalised Edge count
Level 0	N/A	N/A	**31.45**	**31.54**	N/A	N/A
Level 1	17.01	17.15	N/A	N/A	17.12	17.62
Level 2	18.13	17.10	N/A	N/A	17.00	18.17
Level 3	17.85	16.98	N/A	N/A	17.50	18.44
Level 4	17.50	16.64	N/A	N/A	18.27	17.44
Level 5	**18.16**	17.12	N/A	N/A	**18.71**	**18.74**
Level 6	18.07	17.33	N/A	N/A	18.48	17.47
Level 7	17.42	**17.55**	N/A	N/A	18.47	17.33
Level 8	17.32	17.36	N/A	N/A	17.27	17.24
Level 9	18.00	17.07	N/A	N/A	17.49	17.28
Level 10	17.22	17.10	N/A	N/A	17.46	17.53

Table 2: MAP(%) of SF1 for Graph Approach at Different Query Expansion Approaches Using the **Increased Number of Terms** Approach Per Query Word.

Additional terms	DL	MPL	SOPL	PL
Level 1	17.17	18.41	17.36	18.85
Level 2	21.22	20.62	21.06	21.03
Level 3	24.23	21.92	22.49	21.13
Level 4	25.64	22.56	23.58	21.80
Level 5	26.56	22.90	24.36	21.77
Level 6	26.51	23.20	24.86	22.85
Level 7	27.25	23.59	25.18	23.12
Level 8	27.27	23.90	25.33	22.95
Level 9	27.59	23.92	25.80	22.96
Level 10	**27.97**	**24.25**	**25.86**	**23.22**

Table 3: MAP(%) of SF1 for Song's Normalised HAL approach at Different Query Expansion Approaches Using The **Increased Number of Terms** Approach Per Query Word.

Additional Terms	DL	MPL	SOPL	PL
Level 1	18.80	18.03	18.00	18.02
Level 2	21.14	20.76	20.59	20.59
Level 3	22.61	21.48	22.29	22.05
Level 4	23.54	22.30	23.65	22.29
Level 5	24.72	22.55	24.08	22.34
Level 6	25.00	22.97	24.22	22.83
Level 7	25.26	23.43	24.93	23.09
Level 8	25.17	23.50	25.16	23.26
Level 9	25.46	23.97	25.42	23.22
Level 10	**25.80**	**24.27**	**25.58**	**23.45**

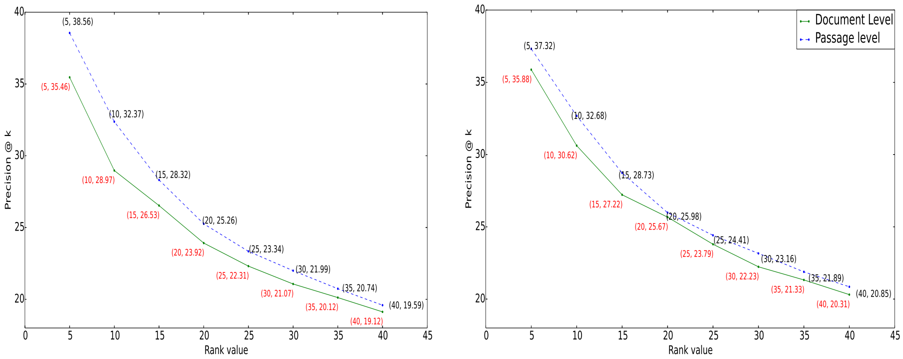
distinctions between terms. While this is a positive characteristic when grouping documents, it shows to return poor results when determining the highest distinguishing associative term. In future work, we aim to confirm this hypothesis by applying the same procedures to group documents and seeing if this maxim holds true.

5 CONCLUSIONS

In this paper, we have undertaken an analysis of approaches to capturing semantic relatedness between terms in the text. While the approach fell short of the baseline used (Rocchio), we did make significant improvements over the basic retrieval performance. With regards to setting the window size, our results are closest to Song's setting of six; we found that five provided better results. The difference might be explained by using different datasets. These figures differ from Burgess's and Kotov's assertions that 8 and 20 respectively were optimal window sizes.

Secondly, we found that increasing the terms added to the query produces better results. It is important to note that the number of additional terms used was only 10 per query term. This is dramatically less than used in the Rocchio approach which uses every term in documents for which feedback was given . Moreover, we are taking evidence from relevant documents only; the Rocchio method also takes evidence from unrelated documents which can help generate a very suitable query.

A third feature of note was the use of passages as pseudo-documents over entire documents. Our intuition was that the use of passages would aid the graphing of concepts because it would remove elements of noise found in a text document which contains a number of topics. To a degree, this intuition proved feasible as the results at passage level were quite competitive. The advantage here is that by applying this preprocessing step it reduces the amount of text needing to be processed. Future work will focus on this pre-processing step. We feel that we can aid the graphing of terms by improving the relatedness of the text to the target term. To achieve this we propose applying Latent Dirichlet Allocation to the corpus and using the results to



(a) Graph Approach - Edge Count (EC) (b) Song's Normalised Edge Count (NEC)

Figure 3: Precision at K for Document and Max passage level.

inform on where best to segment the documents into passages. Secondly, we aim to use the other datasets that contain larger size documents, to see what effect the document size in the collection had on the final results.

ACKNOWLEDGEMENTS

The first author is supported in his research by the Irish Research Council. The second author is supported by Irelands Higher Education Authority through the IT Investment Fund and ComputerDISC in the National University of Ireland, Galway

REFERENCES

Aydın, F., Hüsünbeyi, Z. M., and Özgür, A. (2017). Automatic query generation using word embeddings for retrieving passages describing experimental methods. *Database*, 2017(1).

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Brill, E. (2000). Part-of-speech tagging. *Handbook of natural language processing*, pages 403–414.

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

Gu, Z. and Luo, M. (2004). Comparison of using passages and documents for blind relevance feedback in information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, pages 482–483, New York, NY, USA. ACM.

Hearst, M. A. (1997). Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

Jong, M.-H., Ri, C.-H., Choe, H.-C., and Hwang, C.-J. (2015). A method of passage-based document retrieval in question answering system. *arXiv preprint arXiv:1512.05437*.

Kaszkiel, M. and Zobel, J. (2001). Effective ranking with arbitrary passages. *Journal of the American Society for Information Science and Technology*, 52(4):344–364.

Kotov, A. and Zhai, C. (2011). Interactive sense feedback for difficult queries. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 163–172. ACM.

Lashkari, A. H., Mahdavi, F., and Ghomi, V. (2009). A boolean model in information retrieval for search engines. In *Information Management and Engineering, 2009. ICIME'09. International Conference on*, pages 385–389. IEEE.

Liu, X. and Croft, W. B. (2002). Passage retrieval based on language models. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 375–382. ACM.

Lund, K. and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, 28(2):203–208.

Mitra, M., Singhal, A., and Buckley, C. (1998a). Improving automatic query expansion. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214. ACM.

Mitra, M., Singhal, A., and Buckley, C. (1998b). Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 206–214, New York, NY, USA. ACM.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. *The Smart retrieval system-experiments in automatic document processing*.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Sarwar, G., O'Riordan, C., and Newell, J. (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*, pages 83–90.

Yan, T., Maxwell, T., Song, D., Hou, Y., and Zhang, P. (2010). Event-based hyperspace analogue to language for query expansion. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 120–125. Association for Computational Linguistics.

Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., Chen, Z., and Ma, W.-Y. (2005). Improving web search results using affinity graph. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 504–511. ACM.

Zhou, W., Yu, C., Smalheiser, N., Torvik, V., and Hong, J. (2007). Knowledge-intensive conceptual retrieval and passage extraction of biomedical literature. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 655–662. ACM.