Information Retrieval Final Report

JORRICK BOSHOVE, Radboud University, Netherlands, s4259718
JAAP DIJKSTRA, Radboud University, Netherlands, s4793048
REZA SHOKRZAD, Radboud University, Netherlands, s1056369

Additional Key Words and Phrases: Information Retrieval, TREC, Graph background linking

ACM Reference Format:
Jorrick Boshove, Jaap Dijkstra, and Reza Shokrzad. 2020. Information Retrieval Final Report.

1 INTRODUCTION

Digital innovation has shifted the public's news habit such that around 68% of American adults follow online news in 2018 [1]. Online news consumption has made it easier to provide relevant background information to an article by providing easily accessible links to other articles. This background information allows the reader to gain more insight into the topic described by the article, and can provide the necessary context. However, automatically determining relevant background articles turns out to be a difficult problem. In order to help solve this problem, the Text Retrieval Conference (TREC) [2] established a news track in which researchers can test new solutions for the purpose background linking.

Most of the solutions used to solve the problem of background linking make use of regular ad-hoc search. In this ad-hoc search articles are transformed into queries, often by using a bag-of-words representation of an article and selecting the best, according to some criteria, N words of an article. The search then revolves about finding articles having similar queries. An alternative solution to the background linking problem is one which uses graphs to represent news articles. A possible advantage of this method is that certain relationships between nodes can be encoded in the graph.

In this project, we want to find out whether focusing on the title and first paragraph of an article can improve the performance in the background linking task. This is based on the assumption that the most important part of a news story is given in the beginning of a document, and that subsequent paragraphs contain mostly elaboration or repetition. We use the graph representation of news articles as used by Boers [3] as a basis for our research. We however alter this representation so that we only take the title and first paragraph of an article into account.

2 RELATED WORK

Several studies have researched the use of graphs in background linking. Essem et al. [4] used a graph of an article's text to extract it's most important keywords, and used those keywords as a query in an ad-hoc search. A weighted graph was constructed for an article and then decomposed using different decomposing methods resulting in a graph containing the co-occurrence, order and context similarities of certain terms. Essem et al. found that their method performed better than the median in TREC.

Our work relies heavily on the work done by Boers [3]. In his work, Boers constructed a graph using the 100 highest ranked tf-idf terms as its nodes, and letting edges denote relationships between terms such as co-occurrence in a paragraph or vector similarity. Furthermore different options could be toggled on and off to alter the graphs properties, such as the weights of the nodes and edges. Boers found that his best performing model did not significantly outperform the baseline model using BM25 + RM3.

Some research has been done in using the first N terms of an article in an ad-hoc search. This was done by the anserini team in 2018 [5] in which the first 1000 terms of an article were extracted and used to form a query. They did not look specifically at the first paragraph and did not use a graph-based representation.

3 METHODS

In order to obtain a graph representation of the news articles we used Boers' research as a basis. We deviated from his work by using the title and the first paragraph of an article as the nodes in the graph. In this section we will explain our work in more detail.

3.1 Obtaining the first paragraph

A definition of a paragraph found in the Cambridge Dictionary reads as follows: "A short part of a text, consisting of at least one sentence and beginning on a new line. It usually deals with a single event, description, idea, etc." A news article almost always consists of multiple paragraphs. During our research, we noticed that some news articles have a short first sentence as the first paragraph, meant to catch the readers attention, giving little information about the topic of the article [1]. For those articles we considered the paragraph after the short sentence as our actual first paragraph. This was done by skipping over the first paragraph if it consisted of less than a predefined amount of words.

3.2 Graph structure

In Boers' work a news article was represented by a graph consisting of nodes and edges. The top N tf-idf terms from a document were used to construct the nodes of a graph. Subsequently, nodes were given weights based on their tf-idf score, and edges between nodes were based on the relative positions of the nodes. In our work the selection of nodes for the graph representation differs compared to what Boers used. We will now explain the nodes and edges in more detail.

3.2.1 Nodes. As nodes in the graph we used all terms from the first paragraph and the title. Similar to Boers, some pre-processing was required to obtain better results. Tokens that did not contribute to a word, like ',' were removed. The full list of removed tokens can be found in the appendix. Furthermore stemming was performed such that words having the same stem (e.g early, earlier) were grouped in a single term. Additionally, stop words were removed. Stop words are words that provide little information about the topic of the article they occur in. The full list of removed stop words can be found in the appendix. The resulting terms after pre-processing were used as the nodes in the graph.

Each node in our graph was weighted based on the importance of that node in the graph. The weights of the node were set the in the same way as in Boers' research. This meant that the weight of a node was calculated according to the term frequency and inverse document frequency. Additionally there was an option to use the index of the title and paragraph to specify the weight. For example, words occurring in the title were given higher weights than words occurring in the first paragraph.

3.2.2 Edges. Edges in a graph are often used to signify a relationship between nodes. In Boers' thesis two strategies were used for drawing edges between nodes. One strategy connected nodes within the same paragraph to each other with edges having a weight of 1, and connected nodes appearing in consecutive paragraphs with edges having a weight

[1] https://www.washingtonpost.com/national/health-science/medical-mysteries-she-couldnt-stop-coughing-were-fragrances-to-blame/2015/07/27/7ffad2c2-0c57-11e5-95fd-d580f1c5d44e_story.html

of 0.5. The idea behind this strategy was that successive paragraph introduce each other and nodes in a paragraph describe the same subtopic. The second strategy used a vector representation of word embeddings to obtain a similarity score between vectors. We did not consider this strategy since we did not have the hardware to cope with the memory demands for the word embeddings.

4 EXPERIMENTAL SETUP

The data set on which we have tested our models is the Washington Post data set [6]. It consists of 608,810 news articles from the Washington Post newspaper. For running the experiments, we relied on the code written by Boers. For each of 60 query documents, around 80 candidate documents were ranked according to their relevance to the query document. The found results of the model were compared with predetermined relevance assessments. The metric used for evaluating the results is NDCG@5, the normalized discounted cumulative gain.

Our base model consisted of using only nodes, and no edges. As mentioned previously, the node weights were determined based on their tf-idf scores. Additionally, we tested a setup where the position of the term in the document (i.e. title or first paragraph) was taken into account for the node weight. Third, we used a model with weighted edges defined by the distance between words in a text. Lastly, we tested a model that combined the tf-idf and term position for nodes, and used the text distance for edge weights.

4.1 Reproduction

Our changes and additions to Boers' code can be found on our Github [2]. Our results can be reproduced by first cloning Boers' repository, setting up the resources, and then substituting the files in his repository with the corresponding files in our repository. After having done this, rebuild the image using the command: `docker build . -t blimg` in the background-linking directory. Finally, to run an experiment simply run one of the commands in Boers' README. The commands that match the four setups we tested in our experiment are 'Graph [100 terms, no edges], Graph [100 terms - weights based on term position], Graph [100 terms, edges based on text distance]', and the first command in 'Graph configurations combining: term position, text distance & word embedding' respectively. Note that the $N = 100$ keyword argument is unused, as we simply use all the terms in the first paragraph and title of the documents, instead of selecting the top 100 tf-idf terms from the documents.

5 RESULTS AND DISCUSSION

Model		No skip	**N < 16**	N < 18	N < 20
Boers' best performing graph model	**0.5326**				
Base graph (no edges)		0.3373	0.3376	0.3321	0.3365
Base + Term position		0.3617	0.3973	0.3795	0.3943
Base + Text distance		0.3562	0.3595	0.3520	0.3620
Base + Term position + Text distance		0.3814	**0.4219**	0.4078	0.4121

Table 1. Experimental results

[2] https://github.com/jdijkstra2/IRGroup21

The results of our experiments can be seen in Table 1. '$N < 16, N < 18, N < 20$' in the top row refers to the decision to skip over the first paragraph if it did not contain more than 16, 18 and 20 words respectively. The 'no-skip' scenario refers to the setting where we do not skip over these one-liners.

The base graph with no edges and weights based only on the terms tf-idf score obtained a score of 0.3376. Incorporating the positions of terms in the text improved the performance of the base model. This shows that words occurring in the title generally provide a better description of the document compared to words mentioned in the first paragraph. Consequently, placing higher weights on terms in the title improves performance.

Adding edges to the base model with weights based on the difference between two terms in the text also improved the performance across all chosen N. This shows that drawing connections between words in a document is a valuable addition to the graph representation, and that indeed the distance between the words in a text is a suitable implementation of this. However, placing weights on nodes based on their position in the text proved to result in a greater performance increase.

Our best performing model obtained an NDCG@5 score of 0.4219. This model defined node weights using their tf-idf and positions in the text, and edge weights based on the distance between the nodes in the text. Further decreasing or increasing N dit not improve performance. This model was outperformed by Boers' best performing model (0.5326). This shows that using only the first paragraph (and the title) of a document instead of using the entire document worsens the performance in a background-linking task. A possible explanation for this is that the average length of the first paragraph of an article was around 25 terms. Even after adding the terms of the title, this is still a significantly smaller number of terms in a graph than the amount of terms there would be in a graph using all terms from an article. Therefore it is likely that some information is lost by removing paragraphs other than the first paragraph from an article, information which could be used to distinguish between documents. This would obviously negatively impact the performance.

6 LIMITATIONS

As mentioned previously, we have skipped over the first paragraph if it did not contain more than N words. This was done with the idea that often articles contain attention-grabbing one liners that do not yet describe the specific article. Choosing a boundary for this check was necessary, as it was too time-consuming to check by hand whether a skipped first paragraph was really a one-liner, or instead an extremely short actual first paragraph. We have experimented with several values for N and taken the one that resulted in the best performance. However, it could be the case that we have skipped over legitimate first paragraphs in situations where we should not have, and as a result, affected the performance.

7 APPENDIX

7.1 List of removed tokens

(‘,’, ‘.’, ‘&’, ‘-’, ‘"’, “’, ‘ ’, ‘_’, ‘$’, ‘%’, ‘#’, ‘’, ‘*’, ‘(’, ‘)’, “‘’, ‘”’, ‘"’, ‘""’, ‘–’, ‘..’, ‘...’, ‘....’)

7.2 List of removed stop words

(‘ourselves’, ‘hers’, ‘between’, ‘yourself’, ‘but’, ‘again’, ‘there’, ‘about’, ‘once’, ‘during’, ‘out’, ‘very’, ‘having’, ‘with’, ‘they’, ‘own’, ‘an’, ‘be’, ‘some’, ‘for’, ‘do’, ‘its’, ‘yours’, ‘such’, ‘into’, ‘of’, ‘most’, ‘itself’, ‘other’, ‘off’, ‘is’, ‘s’, ‘am’, ‘or’, ‘who’, ‘as’, ‘from’, ‘him’, ‘each’, ‘the’, ‘themselves’, ‘until’, ‘below’, ‘are’, ‘we’, ‘these’, ‘your’, ‘his’, ‘through’, ‘don’, ‘nor’, ‘me’, ‘were’, ‘her’, ‘more’, ‘himself’, ‘this’, ‘down’, ‘should’, ‘our’, ‘their’, ‘while’, ‘above’, ‘both’, ‘up’, ‘to’, ‘ours’, ‘had’, ‘she’, ‘all’, ‘no’, ‘when’, ‘at’, ‘any’, ‘before’, ‘them’, ‘same’, ‘and’, ‘been’, ‘have’, ‘in’, ‘will’, ‘on’, ‘does’, ‘yourselves’, ‘then’, ‘that’, ‘because’, ‘what’, ‘over’, ‘why’, ‘so’, ‘can’, ‘did’, ‘not’, ‘now’, ‘under’, ‘he’, ‘you’, ‘herself’, ‘has’, ‘just’, ‘where’, ‘too’, ‘only’, ‘myself’, ‘which’, ‘those’, ‘i’, ‘after’, ‘few’, ‘whom’, ‘t’, ‘being’, ‘if’, ‘theirs’, ‘my’, ‘against’, ‘a’, ‘by’, ‘doing’, ‘it’, ‘how’, ‘further’, ‘was’, ‘here’, ‘than’)

REFERENCES

[1] Shearer and K. E. Matsa. News use across social media platforms 2018, [Online]. Available: https://www.journalism.org/2018/09/10/news-use-across-social-media-platforms-2018/.

[2] Ian Soboroff, Shudong Huang, and Donna Harman, *TREC 2018 News Track Overview*, 2018.

[3] Pepijn Boers, *Graph representations of news articles for background linking*, 2020, pp. 40.

[4] Essam, Marwa, and Tamer Elsayed. BigIR at TREC, *Graph-Based Analysis for News Background Linking* 2019, pp. 1–4.

[5] P. Yang and J. Lin, “Anserini at TREC 2018: Centre, common core, and news tracks”, in Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018, Gaithersburg, Maryland, USA, E. M. Voorhees and A. Ellis, Eds., ser. NIST Special Publication, vol. 500-331, National Institute of Standards and Technology (NIST), 2018.

[6] https://trec.nist.gov/data/wapost