Radboud University at TREC 2020

Notebook Paper

Pepijn Boers
p.boers@cs.ru.nl

Chris Kamphuis
chris@cs.ru.nl

Arjen de Vries
arjen@acm.org

1 Introduction

The IR research group from the Radboud University (RUIR) has an interest in creating graph-based solutions for domain specific challenges. We aim to increase effectiveness by incorporating domain knowledge into graph development. This work was developed as part of a Master's thesis project about graph representations for news articles of TREC's background linking task [1]. The introduced work explores the use of named entities, novelty scores and diversification of background documents.

2 News Track

During the revision of background linking methods in prior editions of the News track, we noticed that many participants approached the problem similarly. They extract keywords from the focus article and issue them as search request in a classic ad-hoc information retrieval system. Even though such an approach has yielded good results, we saw some caveats and were interested in exploring an alternative background linking model.

The first apparent weakness we observed was the lack of connectivity between search terms; a bag-of-words query does not give much insight into coherence of a news story and might form a too simplistic representation of a news article for the task of background linking. A second potential weakness might lie in the undue focus on content overlap. Most models use retrieval algorithms like BM25 that rank most overlapping documents highest. Considering the objective to expand a reader's understanding, the presence of new information inside news articles should be somehow rewarded.

ru_graph For our first run we attempted to create a richer representation of news articles by connecting individual query terms in a weighted graph. Article terms were extracted based on their $tf \cdot idf$ score; the 100 highest scoring terms were selected as these have proven to be successful in earlier work [4]. Nodes were weighted based on an alternative form of the $tf \cdot idf$ score as found in the

work by Zhang et al. [9], see equations 1-3.

$$tf_{td} = \begin{cases} \dfrac{1 + log(f_{td} - 1)}{\sum_{i=1}^{n} f_{id}} & \text{if } f_{td} > 1 \\ \dfrac{1}{\sum_{i=1}^{n} f_{id}} & \text{if } f_{td} = 1 \end{cases} \tag{1}$$

$$idf_{tc} = log(\frac{|c| - f_{tc} + 0.5}{f_{tc} + 0.5} + 1) \tag{2}$$

$$W_{td} = tf_{td} \cdot idf_{tc} \tag{3}$$

Term frequency scores (tf_{td}) was obtained by taking the logarithm of the term's original frequency in the article and subsequently dividing that by the sum of all term frequencies in the document. The frequency of term t in document d is denoted by f_{td} and the sum of other terms is denoted as $\sum_{i=1}^{v} f_{id}$. The inverse document frequency (idf) was based on the number of articles in corpus c ($|c|$) and the frequency of a term in the corpus f_{tc}.
Furthermore, work on the entity ranking task from last year[4] showed that the importance of an entity in an article correlated strongly with their position in a text; the earlier an entity is mentioned the more relevant it tends to be. Evaluation on 2019 topics showed that our graph method effectiveness increased when appending a text position score to our node weights. The position score was based on the index of a paragraph a node occurred in, see (4) for the full equation.

$$W_{td} = tf_{td} \cdot idf_{tc} + \frac{1}{index(t, d)} \tag{4}$$

Nodes were connected using semantic features found in word embeddings. The connections were based on the distance between their vector representations as obtained from a large corpus. We used the Wikipedia word embeddings from Gerritse et al. [3]. The cosine similarity between two vectors determined the connection strength between nodes (Eq. 5).

$$W_{t_1,t_2} = \cos(\vec{t_1}, \vec{t_2}) \tag{5}$$

This run kept the widely used overlap criterion as a measure for article relevance, here translated into a variation of the Greatest Maximum Common Sub-graph [2]. A similarity score S gave an indication of the overlap between topic articles Q and candidate articles C by means of a common sub-graph G_{CS}. The similarity score was calculated by summing the node and edge weights in the common sub-graph and dividing those with the maximum sum of node and edge

weights respectively, see equation 6. Subsequently, the scores for node and edge similarity were scaled using hyper parameter λ ($\lambda = 0.5$ in our work).

$$sim(G_Q, G_C) = \lambda \frac{\sum_{n_i \in G_{CS}} w_{n_i}}{max\left(\sum_{n_i \in G_Q} w_{n_i}, \sum_{n_i \in G_C} w_{n_i}\right)} + (1-\lambda) \frac{\sum_{e_i \in G_{CS}} w_{e_i}}{max\left(\sum_{e_i \in G_Q} w_{e_i}, \sum_{e_i \in G_C} w_{e_i}\right)} \tag{6}$$

ru_g_ne In our second run we attempted to create an even richer representation of a news article by including named entities as additional graph nodes. These entities were retrieved using the Radboud Entity Linker (REL) [6] and were integrated in the existing graph.

ru_g_novelty Instead of recommending news articles that have the highest overlap with the focus article, we were interested to see how the rewarding of new information affected the background linking effectiveness. This run tested a method for the retrieval of articles with novel content, where we define novel content as the aggregation of nodes that possess an above average connection to the main story (common sub-graph). We presume that novel nodes contain concepts that are closely related to the main story, yet were not included in the focus article. We computed a novelty score per article by dividing the sum of novel weight by the sum of total weight. Since it remains important that articles cover the same subject matter, we used the harmonic mean of the similarity and novelty score to determine the rank of a candidate article.

ru_g_textrank Our fourth run experimented with a modified version of the TextRank algorithm[5] that recalculated the graph's weights. We presumed that by obtaining new node weights based on the internal connections, we could obtain a more accurate representation of a news story. Node weights were updated following an iterative process in which the sum of all incoming nodes contributed to a node's new weight. The weights of the incoming nodes were normalized using the number of their outgoing connections. The process was stopped whenever the difference between old a new weight became too small (10^{-5}) or when the maximum number of iterations (1000) was reached. We used a damping coefficient of 0.85.

ru_g_diversity Our last submission focused on the diversification of recommended background articles. Since there are no clear guidelines on what is meant with diversity in the context of background linking, we came up with a basic definition using entity types. Many entity extraction models allow for the collection of entity types; these types show the category to which you can assign a specific entity. Examples are: Lionel Messi (Person), F.C. Barcelona (Organization), The Nou Camp (Location). We extracted the following entity types: Person, Organization, Location and Miscellaneous. Articles with a focus on Locations generally contain more location-type entities and could bring a different perspective to a story than person-focused articles. Therefore, we restructured our background recommendations by creating a top 4 with unique

entity type majorities.

ans_bm25 In order to compare our graph methods with the earlier described bag-of-words approach, we collaborated with the Anserini team and used the Anserini software[7] to replicate three traditional runs. These runs were inspired by their work in 2018 and were similar to our work in 2019 [8, 4]. These runs used the BM25 framework to create an initial ranking using an article's top 100 $tf \cdot idf$ terms.

ans_bm25_rm3 The previous run with RM3 reranking.

ans_bm25_rm3_df Same as the previous run but documents that were published later than the focus article are filtered out. The assumption here is that articles that are published later than the focus article can not contain background knowledge. This does not necessarily have to be true following this track's guidelines.

fuse_ru_g_ne_ans_bm25_rm3 We noticed that although *ru_graph* and *ru_ans_bm25* scored similarly on 2019 data, the top 5 documents often consisted of different documents. We found that when we interleaved the rankings of these 2 methods that the resulting ranked was significantly better on the 2019 data. This run was not submitted to TREC.

Note *As a means to lower the high computation time that comes with the creation and comparison of graph methods, we assumed that the ans_bm25_rm3 model (very effective in 2019) would retrieve all possibly relevant documents in its top 100. Therefore, our graph models only performed a re-ranking of those documents to create their top 5.*

3 Results

Table 1: 2020 Results Background Linking Task

Model	NDCG@5	NDCG@10
ru_graph	0.5194	0.5304
ru_g_ne	0.5380	0.5338
ru_g_novelty	0.5106	0.5301
ru_g_textrank	0.4874	0.5010
ru_g_diversity	0.4599	0.4391
ans_bm25	0.5231	0.5105
ans_bm25_rm3	0.5673	0.5596
ans_bm25_rm3_df	0.5279	0.5053
fuse_ru_g_ne_ans_bm25_rm3	0.5913	0.5776

The introduced graph models show similar effectiveness as the bag-of-words models, scoring slightly lower on the mean metric, but slightly better on the median. The only exception is the model with the appended named entities,

which scored higher than two bag-of-words models. An overview is shown in table 1. Again the fusion method seems to help, but the results are not significantly better on this year's data alone.

References

[1] Boers, P. Graph Representations of News Articles for Background Linking. Master's thesis, Radboud University, the Netherlands, 2020.

[2] Bunke, H., and Shearer, K. A graph distance metric based on the maximal common subgraph. *Pattern Recognition Letters 19*, 3 (1998), 255 – 259.

[3] Gerritse, E., Hasibi, F., and De Vries, A. Graph-embedding empowered entity retrieval. In *European Conference on Information Retrieval* (2020), ECIR '20, Springer.

[4] Kamphuis, C., Hasibi, F., de Vries, A. P., and Crijns, T. Radboud University at TREC 2019. In *Proceedings of The Twenty-Eigth Text REtrieval Conference, TREC 2019, Gaithersburg, Maryland, USA* (2019), E. M. Voorhees and A. Ellis, Eds., vol. 1250 of *NIST Special Publication*, National Institute of Standards and Technology (NIST).

[5] Mihalcea, R., and Tarau, P. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing* (Barcelona, Spain, July 2004), Association for Computational Linguistics, pp. 404–411.

[6] van Hulst, J. M., Hasibi, F., Dercksen, K., Balog, K., and de Vries, A. P. REL: An entity linker standing on the shoulders of giants. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (2020), SIGIR '20, ACM.

[7] Yang, P., Fang, H., and Lin, J. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (New York, NY, USA, 2017), SIGIR '17, Association for Computing Machinery, p. 1253–1256.

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

[9] Zhang, Z., Wang, L., Xie, X., and Pan, H. A Graph Based Document Retrieval Method. *Proceedings of the 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, CSCWD 2018* (2018), 660–665.