Radboud University at TREC 2019

Chris Kamphuis*
C.Kamphuis@cs.ru.nl

Tanja Crijns†
tanjacrijns@msn.com

Faegheh Hasibi*
F.Hasibi@cs.ru.nl

Arjen P. de Vries*
arjen@acm.org

February 2020

1 Introduction

The Radboud University Information Retrieval (RU/IR) research group has a research interest in graph based approaches to IR, where we aim to exploit the flexibility of a graph representation of documents and other types of information (such as entities) to achieve increased retrieval effectiveness, e.g. by integrating extra knowledge about a domain. The main focus of our participation in TREC 2019 has been the News Track, where we see a large potential to improve search using graph based representations. We have also participated in the new Conversational Assistance Track, where we have explored how to make use of the conversational context to improve ranking answer passages.

2 News Track

2.1 Background linking

For the background linking task, we first reviewed the submissions to the 2018 edition of the Track [12]. One of the best performing runs in 2018 uses BM25 to produce an initial ranking and subsequently re-ranks those results using a relevance model, while filtering out the articles that were published *later* than the topic article [7]. We re-implemented this approach

*Radboud University, Nijmegen, The Netherlands
†Alumna, Radboud University, Nijmegen, The Netherlands

as a baseline, for comparison to a variety of ways to include entity information in the search process.

ru_bm25_rm3_fil We started with a run similar to the best performing run of last year, using the Anserini (v0.6.0) [13] system. Firstly, documents were retrieved using BM25. From the source article a query was constructed, the 100 terms with the highest TF · IDF per topic document were used. Using 100 terms worked well when evaluating on the topics of last year. From the top 10 retrieved documents, an RM3 query was constructed; in the remainder, we refer to this method as $RM3_{BM25}$.[1] Using the newly formulated query, again the whole collection was ranked. Articles published later than the source article were filtered out of the resulting ranked list.

ru_bm25_rm3 A run without the date filter, to confirm or disprove its effectiveness.

ru_sdm_rm3_fil Because the Sequential Dependency Model (SDM) has shown good baseline performance in many entity retrieval benchmarks, we include a run where we replace the BM25 initial ranking by one produced by SDM, again interpolated with that run re-ranked using a relevance model; denoted by $RM3_{SDM}$. We use the settings suggested by the Anserini group in 2018.[2] Finally, we apply a date filter to the result list.

ru-ent-90-10-df The next method re-ranks the baseline using entity information. We tagged the articles in the collection using TagMe [4]. Using the linked entities, we can compute the ELR score introduced by Hasibi et al. [5] for each pair of articles (query document and candidate news article). Eq. (1) defines the ELR score:

$$f_E(e, D) = \log \sum_{f \in \mathcal{F}} w_f^E \left[(1-\alpha)\, tf_{\{0,1\}\left(e, \hat{D}_f\right)} + \alpha \frac{df_{e,f}}{df_f} \right] \tag{1}$$

As we only used article content to build the index, we do not sum over fields (there is only a single content field, $w_f^E = 1$). The value of α for Jelinek-Mercer smoothing equals 0.1. $tf_{\{0,1\}\left(e, \hat{D}_f\right)}$ takes a value of 1 if entity e is present in document $\hat{D}_f$, 0 otherwise. $df_{e,f}$ is the number of documents

[1] Technically, RM3 refers to a linear combination of the original query's results *using a language modelling approach to IR* with those of a query produced by Lavrenko's Relevance Model 1 [8], derived from the top documents [6]. In our case however, it is the BM25 retrieval score that is interpolated with the results of the RM1 relevance model, denoted as $RM3_{BM25}$.

[2] `https://github.com/castorini/anserini/blob/master/docs/runbook-trec2018-anserini.md`, Last Accessed: October 28th, 2019

where entity e is present, which is normalized by the total number of documents df_f (as there is only one field). The ELR score is then combined with the $RM3_{BM25}$ score, following Eq. (2).

$$rsv = \lambda_T \sum_{q_i \in Q} f_T(q_i, D) + \lambda_E \sum_{e \in E(Q)} s(e) f_E(e, D) \tag{2}$$

Here, $f_T(q_i, D)$ equals the $RM3_{BM25}$ score, and $s(e)$ corresponds to the linking confidence score (provided by TagMe) for entity e in the source article. Both partial scores are normalized using min-max normalization (prior to their combination), and weighted by parameters λ_T and λ_E. For this run λ_T and λ_E are set to respectively 0.9 and 0.1. Finally, articles published after the topic article date are filtered out.

ru-ent-95-05-df The approach as above with $\lambda_T = 0.95$ and $\lambda_E = 0.05$.

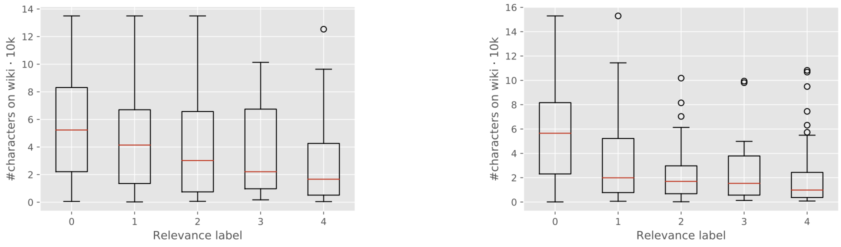
2.2 Entity Ranking

For the entity ranking task (that should perhaps have been called an *entity salience* task), we looked into finding simple yet effective heuristics. Only a limited number of entities has to be ranked in order of importance, and the entities and their mentions in the source articles were provided by the task. Based on pilot experiments using the 2018 data, we found that simple heuristics are highly effective. Our runs for this task use just these heuristics, with the aim to provide a hard-to-beat baseline for more advanced methods to compare to.

ru-t-order The first thing we tried (evaluated on last year's test collection) was to create a run where the entities are simply ranked in the order as they are presented in the topic file; making the implicit assumption that the topic author lists important entities first. This approach already yields a high NDCG score; perhaps just because the number of entities to rank is low and the ratio of relevant entities among those to be ranked is high.

ru-m-order Salient entities tend to appear at the start of a news article, a phenomenon that is well known and provides a strong baseline [3]. Our second run therefore ranks the entities in their order of mention in the article, another competitive baseline that yields a higher NDCG than **ru-t-order** on the 2018 test collection.

ru-tf-m-ord Another assumption we made was that entities that are important for a story, tend to be mentioned more often than entities that are less important. This run ranks entities according to their mention frequency in the article. If entities are mentioned an equal number of times, they were



(a) TREC News 2018.

(b) TREC News 2019.

Figure 1: Relation between relevance grade and Wikipedia page length for entities in the 2018 (left) and 2019 (right) test collections.

ranked according to their position, preferring the entity that is mentioned first.

ru-invwiki The track only considers entities with a Wikipedia page. We decided to use the number of tokens on the Wikipedia page of the entity as an indication of salience, based on the observation that entities with less associated information in the knowledge graph are more discriminative, and therefore *a priori* more likely to be relevant. Figure 1a shows the correlation between relevance and Wikipedia page length in the 2018 test collection, supporting the potential of this characteristic. The run ranks the entities based on the number of tokens in their Wikipedia page, where the top ranked entity has the lowest number of tokens.

ru-tf-invwiki Instead of only ranking entities on the number of tokens in their Wikipedia page, we first rank the entities on the number of mentions in the topic article. If the number of mentions is equal between two entities, the entity with the longer Wikipedia page is ranked higher.

2.3 Results

Tables 1 and 2 summarize the results for the background linking and entity ranking tasks, respectively. We discuss each of the tasks below.

2.3.1 Background linking

As shown by the difference in effectiveness scores between `ru_bm25_rm3` and `ru_bm25_rm3_fil`, the date filter did not improve results on the 2019 test collection, as opposed to last year (date was one of the strongest features for background reading in [7]). Initially, when this track was started,

Table 1: Results background linking

Method	NDCG@5
ru_bm25_rm3_fil	0.513
ru_bm25_rm3	**0.527**
ru_sdm_rm3_fil	0.495
ru-ent-90-10-df	0.503
ru-ent-95-05-df	0.502

it was determined that articles would not be relevant when they were published after the topic article. The idea behind this was that when an article is published, only articles that are published earlier can be recommend as background reading. However, the assumed user scenario was changed later on: when reading an article from the past, articles that were published later might actually be more suited as background reading, as they can provide a more information (e.g. by providing information on how the story developed). So, the context was swapped from reading the topic article at the moment it is published, to reading it some point after it was published; making newer articles potentially suitable for background reading. We attribute the difference in effectiveness of applying a date filter to this change in user scenario; maybe, a large proportion of the runs submitted in 2018 did apply a filter on publication date, explicitly or implicitly, and the pool simply did not include the relevant articles published after the topic article.

The run using $RM3_{SDM}$ did not perform as well as its corresponding $RM3_{BM25}$ run. We have not analyzed the root of this difference in detail, but a possible explanation lies in the simple approach to produce the query; the bi-grams and skip-grams considered are those that include any of the top 100 terms that were selected using TF·IDF weighting, an approach that might not identify the most informative phrases.

Adding entity information also did not help improve effectiveness. The ru-ent-90-10-df run achieves higher effectiveness than ru-ent-95-05-df, so we plan to explore further by increasing the weight on the entity-related part of the model; properly tuning the parameter might result in a positive effect. The method we used to re-rank the results using entity information was initially proposed for an entity retrieval task, and the TagMe entity linker [4] was created with the goal of linking short fragments of text to the knowledge base. As we tagged full news articles that are likely written differently from short pieces of text, the linking quality might have

Table 2: Results entity ranking

Method	NDCG@5
ru-t-order	0.397
ru-m-order	0.500
ru-tf-m-order	0.538
ru-invwiki	**0.622**
ru-tf-invwiki	0.585

suffered.

2.3.2 Entity ranking

The first interesting thing we observe in the entity ranking results is the effectiveness score of the trivial baseline that ranks the entities according to topic order ru-t-order, with an NDCG@5 score of approximately 0.4. We should take into account that the task is fundamentally different from most IR tasks, because the fraction of relevant "documents" (entities) is much higher for this problem than in the usual case of ad-hoc retrieval; even a random permutation will do well, seemingly. Let us keep in mind that high scores do not necessarily indicate that we understand the task in depth.

Ordering the entities by their appearance in the topic article gives an NDCG@5 of approximately 0.5, an increase of 10% absolute. This result confirms the intuition that news articles mention important entities in the first few sentences. Taking the mention occurrence frequency into account by ranking on the within-document frequency and breaking ties by the mention's location increases the NDCG@5 to 0.538. Frequently mentioned entities automatically have a higher likelihood to appear earlier in the article, so the two approaches are clearly dependent. The observed performance increase matches our intuition that entities mentioned multiple times in the topic article are more relevant than those mentioned only once.

The best performing run is ru-invwiki, which achieves a NDCG@5 of 0.622, almost another 10% absolute more effective than the previously best approach. This run ranks the entities according to the length of their Wikipedia pages, where entities with shorter Wikipedia pages have been ranked higher. The idea behind this heuristic is that it serves a similar purpose as IDF in regular retrieval tasks. Solely using Wikipedia length as a metric gives a better effectiveness than doing this only to break ties for the entities ranked on within-article frequency (`ru-tf-invwiki`). Interestingly, when evaluating these approaches on last year's topics set, we found the opposite to be true. Finding a way to combine these metrics into one would be a nice follow up study, for which a variety of TF · IDF weighting schemes may be considered.

Given that these very simple heuristics already achieve high effectiveness scores, we agree with the track organizers to increase the difficulty of the problem investigated. The entities to be ranked should probably not be pre-selected and provided in the topic files, shifting the focus of the problem to the complete entity linking pipeline.

3 Conversational Assistance Track

The Radboud University IR (RU/IR) research group also participated in the Conversational Assistance Track (CAsT), with the goal to explore the usefulness of conversation context for ranking the passages.

We consider a two-stage retrieval system that uses Google's BERT language model [2] to re-rank candidate answer passages retrieved by a standard BM25 retrieval system, inspired by two recent papers that show significant improvements for ranking answer passages over a BM25 baseline [10, 11].

The basic approach is to calculate BERT activations for query and answer pairs, and re-rank the candidate answer passages identified by BM25 accordingly. Preliminary experiments using the MS-MARCO collection [9] confirmed the reported effectiveness of such an approach, where for many queries their known relevant answer passage is indeed ranked (sometimes much) higher with BERT re-ranking.

A challenge for re-ranking with BERT however is that the query length is inherently limited by the size of the model that we can use. Google shares pre-trained models with a context of 512 tokens, which limits the amount of conversation context that can be taken into account. In our experiments, we consider two approaches to deal with this limitation: simply clipping the conversation context that is considered to that which fits the input size of the model, or, alternatively, applying a fusion method to combine the rankings for multiple turns in the conversation.

Initial work to explore this direction has been described in the Master's thesis by one of the authors of this paper, and we hope to validate the results of her experiments using the CAsT test collection [1]. The results of [1] indicated that using conversational context was only beneficial in the

BM25 retrieval step. Adding conversational context when re-ranking with BERT decreased performance in comparison to re-ranking with BERT using only the relevant query. However, the *max fusion* approach showed the most promising results, and this approach has been included in the runs.

We have two runs for evaluation:

bm25_bert_fc, in the BM25 candidate retrieval step, we use all conversational context and the last query as input. For the BERT re-ranking step, we only use the last query.

bm25_bert_rankf, in the BM25 candidate retrieval step, we use all conversational context and the last query as input. For the BERT re-ranking step, we consider the three last turns in the conversation as context. We re-rank each turn with all candidate passages and apply MAX score fusion to the three intermediate results.

	NDCG@5	MAP
bm25_bert_fc	0.347	0.159
bm25_bert_rankf	0.350	0.159
Median	0.296	0.174

We cannot immediately confirm that BERT re-ranking is effective, because we did not submit a plain BM25 run. We see that the mean average precision of our complete result list is lower than that of the median system, but the early precision measured by NDCG@5 is higher. We cannot conclude however that taking more context into account leads to improved effectiveness, as the difference in NDCSG@5 is negligible. More detailed analysis of the results will be included in the final TREC proceedings paper.

References

[1] Crijns, T.: Have a chat with BERT; passage re-ranking using conversational context. Master's thesis, Radboud University (Aug 2019), https://www.ru.nl/publish/pages/769526/tanja_crijns_msc_thesis_ds_22_8_2019.pdf

[2] Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019,

Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers). pp. 4171–4186. Association for Computational Linguistics (2019), https://www.aclweb.org/anthology/N19-1423/

[3] Dunietz, J., Gillick, D.: A new entity salience task with millions of training examples. In: Bouma, G., Parmentier, Y. (eds.) Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden. pp. 205–209. The Association for Computer Linguistics (2014), https://www.aclweb.org/anthology/E14-4040/

[4] Ferragina, P., Scaiella, U.: Tagme: On-the-fly annotation of short text fragments (by wikipedia entities). In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management. pp. 1625–1628. CIKM '10, ACM, New York, NY, USA (2010). https://doi.org/10.1145/1871437.1871689, http://doi.acm.org/10.1145/1871437.1871689

[5] Hasibi, F., Balog, K., Bratsberg, S.E.: Exploiting entity linking in queries for entity retrieval. In: Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval. pp. 209–218. ACM (2016)

[6] Jaleel, N.A., Allan, J., Croft, W.B., Diaz, F., Larkey, L.S., Li, X., Smucker, M.D., Wade, C.: Umass at TREC 2004: Novelty and HARD. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004. vol. Special Publication 500-261. National Institute of Standards and Technology (NIST) (2004), http://trec.nist.gov/pubs/trec13/papers/umass.novelty.hard.pdf

[7] John Foley: Trec News Background-Linking 2018: Filter By Time! . https://jjfoley.me/2019/07/24/trec-news-bm25.html (2019)

[8] Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Croft, W.B., Harper, D.J., Kraft, D.H., Zobel, J. (eds.) SIGIR 2001: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, September 9-13, 2001, New Orleans, Louisiana, USA. pp. 120–127. ACM (2001). https://doi.org/10.1145/383952.383972, https://doi.org/10.1145/383952.383972

[9] Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A human generated machine reading comprehension dataset. In: Besold, T.R., Bordes, A., d'Avila Garcez, A.S., Wayne, G. (eds.) Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016. CEUR Workshop Proceedings, vol. 1773. CEUR-WS.org (2016), http://ceur-ws.org/Vol-1773/CoCoNIPS_2016_paper9.pdf

[10] Nogueira, R., Cho, K.: Passage re-ranking with BERT. CoRR **abs/1901.04085** (2019), http://arxiv.org/abs/1901.04085

[11] Padigela, H., Zamani, H., Croft, W.B.: Investigating the successes and failures of BERT for passage re-ranking. CoRR **abs/1905.01758** (2019), http://arxiv.org/abs/1905.01758

[12] Soboroff, I., Huang, S., Harman, D.: Trec 2018 news track overview. In: The Twenty-Seventh Text REtrieval Conference (TREC 2018) Proceedings. Gaithersburg, Maryland, USA (2019)

[13] Yang, P., Fang, H., Lin, J.: Anserini: Reproducible ranking baselines using Lucene. Journal of Data and Information Quality **10**(4), 16 (2018)