Elastic Embedded Background Linking for News Articles with Keywords, Entities and Events

Luis Adrián Cabrera-Diego*
University of La Rochelle, L3i
La Rochelle, France
a.cabrera@jusmundi.com

Emanuela Boros
University of La Rochelle, L3i
La Rochelle, France
emanuela.boros@univ-lr.fr

Antoine Doucet
University of La Rochelle, L3i
La Rochelle, France
antoine.doucet@univ-lr.fr

ABSTRACT

In this paper, we present a collection of five flexible background linking models created for the News Track in TREC 2021 that generate ranked lists of articles to provide contextual information. The collection is based on the use of sentence embeddings indexes, created with Sentence BERT and Open Distro for ElasticSearch. For each model, we explore additional tools, from keywords extraction using YAKE, to entity and event detection, while passing through a linear combination. The associated code is available online as open-source software.

CCS CONCEPTS

• **Information systems** → **Retrieval models and ranking**; **Language models**; **Rank aggregation**.

1 INTRODUCTION

With the massification of the internet and mobile devices, such as smartphones, people have started to access news more frequently from digital sources than printed ones [11, 13]. This has meant that newspaper publishers have had to focus more on the digital experience and perform users' behavioral analysis for providing tools such as news recommendation [33]. Furthermore, as Pranjić et al. [27] indicate, linking news to other relevant articles can improve businesses' websites metrics such as user engagement and average time on page. Subsequently, this can improve revenues from ads or sponsored articles.

Therefore, in 2018 the *Text REtrieval Conference (TREC)* along with *The Washington Post*[1], decided to propose the News Track [30], a track where the goal is to enhance users' experience while reading news articles.

Since TREC 2020, the News Track is organized into two subtasks, *Background Linking* and *Wikification*. The former has been defined as the task where "given a news article, a system should retrieve other news articles that provide important context and/or background information that helps the reader better understand the query article" [29]. The latter exploits, as a means of contextualization, the linking of textual elements, such as concepts and artifacts, to an external knowledge-base, in this case to Wikipedia [31].

In this paper, we present the participation of the *L3i Laboratory* of the University of La Rochelle at the 2021 TREC News Track *Background Linking* task. Our participation consisted of five different approaches that used, for instance, keyword extraction, entities, and events detection, but also sentence embeddings and linear combination.

[1] https://www.washingtonpost.com/

2 RELATED WORK

Before TREC 2018 News Track, there is a reduced number of works that explore the use of news articles as a way to contextualize elements such as comments [1], tweets [14], or events [25].

Since the proposal of the News Track in TREC 2018, we have seen an increment of publications related to the contextualization of news articles using other news articles. Most of them are works explaining TREC participant systems [17, 20, 24]. However, we can find as well some other related outputs and analysis [12, 18].

More recently and besides TREC-related outputs, we can name the work of Pranjić et al. [27], where the authors explore different models to link background and related news articles in a Croatian corpus. Furthermore, Koloski et al. [19] explore the linking of cross-border news articles in Latvian and Estonian. Also, we can name the MIND dataset [33], a collection of news articles from *Microsoft News* that are associated with human behaviors, in order to explore news recommendation tasks.

3 DATA

For 2021, the TREC News Track organizers provided a corpus composed of 728,626 news articles and blog posts published by *The Washington Post* from January 2012 through December 2020. Each document, either news article or blog post, includes elements such as title, kicker (section header), body, author, images captions, and publication date. Also, TREC organizers delivered a list of 51 different topics, i.e. news articles, for which TREC News Track's participants had to propose background articles. For the 2021 edition of TREC News Track, the organizers also added a subtopic task, in which specific information, such as the background, is expected for each topic. In Figure 1, we present the topic structure used in the 2021 TREC News Track.

We first performed a pre-processing that consisted of parsing each document element, such as titles and captions, in order to get sentences. This pre-processing was done using Turku Neural Parser [16].

Once the documents were pre-processed, we decided to create embeddings for every document element using Sentence BERT [28], a fine-tuned BERT [10] which produces embeddings that can be compared using cosine similarity. Specifically, we made use of the

```
<top>
<num>Number: xxx </num>
<docid>f30b7db4−cc51−11e6−a747−d03044780a02</docid>
<url>https://www.washingtonpost.com/local/public−safety/
homicides−remain−steady−in−the−washington−region/
2016/12/31/
f30b7db4−cc51−11e6−a747−d03044780a02_story.html</url>
<title>Topic title</title>
<desc>I would like to learn more about this topic</desc>
<narr>Traditional TREC narrative paragraph on the topic</narr>
<subtopics>
<sub num="1">This is the first subtopic.</sub>
<sub num="2">And this is the second one.</sub>
</subtopics>
</top>
```

Figure 1: Structure of TREC News Track 2021 topics, where the description and subtopics fields were added.

pre-trained model *stsb-mpnet-base-v2*[2] which at the time of the experiments was the most performing model available.

Due to limitations on how many tokens can be processed by Sentence BERT, i.e. 128 byte-pair encoding tokens, and to avoid losing vital information, we calculated the embeddings sentence by sentence. To be precise, we requested from Sentence BERT model the dense representation of each token in every sentence. The final dense representation of a text portion was obtained by averaging the dense vector of every token.

It should be indicated as well that we created composite vectors, in which we calculated the average embeddings based on multiple document elements: Title-Lead, Title-Body, and Title-Body-Captions. We also processed, in the same way, each topic provided by the TREC organizers, which notably included the creation of dense vectors for the narration or for the subtopics.

For retrieving documents from the corpus, we indexed the pre-processed data using *Open Distro for ElasticSearch*[3] (ODFE), an *ElasticSearch*[4] branch which implements a performing k-NN algorithm that can be used to retrieve documents using dense vectors, such as embeddings.[5]

In total, we indexed 728,500 articles from The Washington Post, which corresponded to 99.98% of the articles provided by TREC organizers. The code for pre-processing and indexing the data is publicly available in GitHub[6]. It should be noted, in the code, that the indexes contained more information than the one detailed in this work. However, not all the information was used for the creation of the submitted approaches.

[2]https://huggingface.co/sentence-transformers/stsb-mpnet-base-v2
[3]https://opendistro.github.io
[4]https://www.elastic.co/
[5]Although we use ODFE instead of ElasticSearch, the documentation of the latter is valid except for the dense vectors queries. Thus, we will point to ElasticSearch 7.12's documentation in specific cases.
[6]https://github.com/EMBEDDIA/news_background_linking

4 EXPLORED APPROACHES

In this section, we describe in detail the five approaches we explored to provide background links for each topic:

(1) **KWVec**: keywords and dense vectors to retrieve the related background articles;
(2) **Lambda**: linear combination of multiple queries;
(3) **300K_ENT_PH**: the articles retrieved by KWVec are re-ranked with the utilization of entities and event mentions;
(4) **300K_ENT_PH_DN**: the articles retrieved and re-ranked by 300K_ENT_PH are again sorted depending on the description and the narrative field;
(5) **Lambda_narr**: the outcome produced by the Lambda approach is followed by re-ranking the recommended articles using the narrative field.

Each of the following sections detail the five approaches used to provide subtopic background linking. These five approaches consist of re-rankings of the former approaches.

4.1 Run 1: KWVec

This approach consists of using keywords and dense vectors to retrieve the related background articles for a determined topic.

Specifically, we start by extracting unigram keywords from the text produced by the concatenation of the title, body, and captions.[7] This is done using YAKE [9], an unsupervised keyword extractor. Once we have the unigram keywords, we obtain those related to the title by matching the title's unigrams and the obtained keywords.

The second step of KWVec consists of using a *boosting query*[8], where a collection of queries are used to retrieve the documents, and another set is used to decrease their relevance.

To retrieve the documents, we submit three different queries to ODFE. Two of them ask ODFE to retrieve the documents that are relevant to the keywords found by YAKE. To be precise, we search title keywords in titles and body keywords in bodies. These queries are done through a *query string query*[9].

Furthermore, as YAKE assigns a weight w to each keyword, we make use of these weights to increase or decrease the *query string query* relevance through the *boost* parameter. Nonetheless, as YAKE's weights interval is between $(0, \infty)$, where the lower the score the better, we modify it with Equation 1 to an interval of $(inf, 0]$, where the higher the score the better.

$$KW_{weight} = \begin{cases} -ln(w) & \text{if } w < 1 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The third query retrieves the most relevant documents using ODFE's *exact k-NN* and cosine similarity.[10] Specifically, the cosine similarity is calculated between the title-body dense vectors of the topic article and those found in the index.

It should be indicated that we modified ODFE's cosine similarity (s) score using Equation 2. The first reason is that ODFE's cosine

[7]We concatenate these text fields in order to get more relevant keywords. Focusing separately on smaller text portions, such as the title, produced less relevant keywords.
[8]https://www.elastic.co/guide/en/elasticsearch/reference/7.12/query-dsl-boosting-query.html
[9]https://www.elastic.co/guide/en/elasticsearch/reference/7.12/query-dsl-query-string-query.html
[10]https://opendistro.github.io/for-elasticsearch-docs/docs/knn/knn-score-script/

similarity is vertically translated, within the interval $[0, 2]$, to provide only positive scores. The second reason is to boost the cosine similarity by a scalar defined experimentally to 250 and prevent its fading with respect to the keywords scores.

$$Sim = \begin{cases} 250 \times (s-1) & if s \geq 1 \\ 0 & otherwise \end{cases} \quad (2)$$

We requested ODFE to reduce by 20% the relevance of documents that are associated with an unwanted kicker[11] and/or whose title was similar to the topic's. The former aspect was to reduce the relevance of articles that are frequently not used by The Washington Post's journalists. The latter aspect was calculated using *exact k-NN* and cosine similarity between titles dense vectors. We do this to avoid articles that might be considered relevant because they are either a duplicate of the topic article[12] or whose title is too similar.

4.2 Run 2: Lambda

Besides the previously described approach, we decided to explore a linear combination (see Equation 3) optimized through a Bayesian optimization algorithm [23][13]. Through this optimization, our goal was to determine the weights (λ) that different queries scores (x), such as title similarity, should be given in order to achieve the highest nDCG evaluation. This approach is similar to the one used by Cabrera-Diego et al. [8] for merging different systems outputs.

$$\lambda_1 x_1 + \lambda_2 x_2 + \ldots + \lambda_n x_n \quad (3)$$

For the Lambda approach, we explored four different independent queries[14], *title* to *title*, *body* to *body*, *lead* to *title* and *lead* to *body*, using two methods, keywords and dense vectors. This gave a total of eight different independent queries used for the optimization. The queries based on keywords use the method presented in Section 4.1, while queries based on dense vectors used an unmodified version of ODFE's *exact k-NN* and cosine similarity. Furthermore, for the Lambda approach, we removed from the recommended articles those with an unwanted kicker (see Footnote 11).

To calculate the value of the different λ, we used as training data the sets provided by the organizers from previous years plus some additional articles that we annotated ourselves.[15] Specifically, we requested ODFE to calculate[16] the relevance score of the eight queries for each document for which we had a gold standard score. Then, the Bayesian optimization proposed different λ weights, in the interval of $[-10, 10]$, that optimized an objective function.

The objective function to be maximized by the Bayesian optimization is presented in Equation 4, where G is a weighted harmonic average, Q_1 and Q_3 are respectively the first and third quartile, and Q_2 is the median. These values are calculated based on the nDCG@10 scores obtained by each topic for all the years (2018-2020).[17]

$$G = \frac{5Q_1Q_2Q_3}{(Q_2Q_3) + (2.5Q_1Q_3) + (1.5Q_1Q_2)} \quad (4)$$

The weighted harmonic average presented in Equation 4 was defined to boost the median (Q_2) nDCG@10 score, but also to create a negatively skewed distribution of the nDCG@10 scores, by boosting the third quantile (Q_3). This would mean that we expect most of the nDCG@10 scores to have higher values rather than lower ones.

4.3 Run 3: 300K_ENT_PH

This approach extends the KWVec method with a re-ranking step applied after the relevant documents were retrieved by the ODFE query. Thus, since named entity recognition (NER) has been playing an important role in information seeking and retrieval, we propose to exploit knowledge about entities and their relationships (events) for re-evaluating the relevance of the query results. For this and for taking advantage of the annotation efforts from previous campaigns, we leverage the fine-grained entities defined by the organizers of the TAC KBP *Recognizing Ultra Fine-grained Entities* (RUFES) 2020[18] and the events defined by the ACE 2005 evaluation campaign[19].

4.3.1 Fine-grained Entities. The KBP 2020 RUFES dataset provided by the organizers consisted of the development source documents and evaluation source documents drawn from a collection of The Washington Post news articles. The development source corpus and the evaluation source corpus had approximately 100, 000 articles each, from which 50 documents were annotated for the development set with entity types from an ontology that contains approximately 200 fine-grained entity types and that followed the same three-level x.y.z hierarchy as in the TAC-KBP 2019 EDL track [15][20]. For example, such an entity organized in a hierarchy is: *Photographer* is from an *Artist* that, in turn, is a subtype of *PER*[21].

In order to benefit from the extraction of these entity types, we made use of our recently proposed model for coarse-grained and fine-grained named entity recognition [3–5, 7] that consists in a hierarchical, multitask learning approach, with a fine-tuned encoder based on BERT [10]. This model includes the use of a stack of Transformer [32] blocks on top of the BERT encoder. The multitask prediction layer consists of separate conditional random field (CRF) layers.

In Table 1, we explore two pre-trained and fine-tuned BERT *cased* models, BERT-base and BERT-large. We further consider the BERT-large-cased +2×Transformer, and we extract the fine-grained entities from the query and the retrieved articles.

4.3.2 Events. The annotated data of the ACE 2005 corpus provided by the ACE evaluation is restricted to a range of types, each with a set of subtypes. Thus, only the events of an appropriate type

[11] Opinions, Opinion, Letters to the Editor, The Post's View, Global Opinions, All Opinions Are Local, Local Opinions

[12] Although the organizers removed most of the duplicate articles, the process was not without faults.

[13] https://github.com/fmfn/BayesianOptimization

[14] This means that each query was done one by one.

[15] We annotated five recommended articles per topic, about which we did not know anything. The recommended articles came from the title to title dense vector queries.

[16] https://www.elastic.co/guide/en/elasticsearch/reference/7.12/search-explain.html

[17] We explored different nDCG cuts, such as 50, 20 and 5. However, we found that, empirically, optimizing at 10 provided the best global results.

[18] https://tac.nist.gov/2020/KBP/RUFES/index.html

[19] http://catalog.ldc.upenn.edu/LDC2006T06

[20] RUFES annotation guidelines: https://tac.nist.gov/2020/KBP/RUFES/guidelines/RUFES2020AnnotationGuidelines.v1.1_draft.pdf

[21] *PER* refers to the entity type *Person*.

Table 1: Performance of different systems for RUFES, micro-strict.

Approaches	Precision	Recall	F1
BERT-base-cased	75.4	69.4	72.3
BERT-large-cased	79.1	72.5	75.6
+ 2 × **Transformer**			
BERT-base-cased	75.9	69.2	72.4
BERT-large-cased	**79.9**	**73.2**	**76.4**

Table 2: Performance of different systems for ACE 2005 on the blind test data, micro-strict.

Models	Precision	Recall	F1
BERT-base-cased	71.3	72.0	71.6
BERT-large-cased	69.3	**77.1**	73.0
+EntityMarkers			
BERT-base-cased	79.1	72.5	75.6
BERT-large-cased	**82.4**	75.7	**78.9**

are annotated in a document. The eight event types (with 33 subtypes in parentheses) are: *Life (Be-Born, Marry, Divorce, Injure, Die)*, *Movement (Transport)*, *Conflict (Attack, Demonstrate)*, etc.

Events are distinguished from their mentions in text. An event mention or a trigger is a span of text (an extent, usually a sentence) with a distinguished trigger word and zero or more arguments, which are entity mentions, timestamps, or values in the extent. Since there is nothing inherent in the task that requires the two levels of type and subtype, we will refer to the combination of event type and subtype (e.g., *Life.Die*) as the event type. If we consider the example sentence "*There was the free press in Qatar, Al Jazeera but its' offices in Kabul and Baghdad were bombed by Americans.*", an event extractor should detect a *Conflict.Attack* event mention, with the trigger word *bombed*.

For detecting events, we focus on the event mention detection, and we use a BERT-based model with entity markers [2, 6, 21, 22]. This method is adapted from the BERT-based model with *EntityMarkers* [2] applied for relation classification, to perform event detection.

The *EntityMarkers* model consists in augmenting the input data with a series of special tokens, e.g., if we consider a sentence $x = [x_0, x_1, \ldots, x_n]$ with n tokens, we augment x with two reserved word pieces to mark the beginning and the end of each entity in the sentence. Thus, the previous sentence becomes: *There was the free press in* $[GPE.Country_{start}]$ *Qatar* $[GPE.Country_{end}]$, $[ORG.CommercialOrganization_{start}]$ *Al Jazeera* $[ORG.CommercialOrganization_{end}]$ but its' offices in $[GPE.City_{start}]$ *Kabul* $[GPE.City_{end}]$ and *Baghdad* were bombed by $[ORG.Government.Agency_{start}]$*Americans* $[ORG.Government.Agency_{end}]$, where the different hierarchical entity types were detected by the previously presented model for fine-grained entity recognition.

In Table 2, we explore again the two pre-trained and fine-tuned BERT *cased* models, the BERT-base and BERT-large, with and without the entities previously predicted. We further consider the BERT-large-cased + 2 × Transformer, and we extract the event triggers from the query and the retrieved articles.

4.3.3 Re-ranking. For each sentence of the article, the entities and the event triggers are extracted and concatenated separated by a space, forming two separate text lines. Each line of entities or event triggers is encoded with Sentence BERT and then, the final representation is the sum of all the obtained vectors $v = (v_i)_{i=1}^{n}$ where each element $v_{i,j} = \sum_{j=1}^{n} xi, j$ We use the cosine similarity for comparing the entity representations, which is defined as follows:

$$\cos(Q, R) = \frac{QR}{\|Q\|\|R\|} = \frac{\sum_{i=1}^{n} Q_i R_i}{\sqrt{\sum_{i=1}^{n} (Q_i)^2}\sqrt{\sum_{i=1}^{n} (R_i)^2}} \quad (5)$$

where Q is the vector representation of the Query and Retrieved is the vector representation of the retrieved article.

$$score(R) = \Big(\cos(Q_{Entities}, R_{Entities}) + \cos(Q_{Events}, R_{Events})\Big)/2 \quad (6)$$

4.4 Run 4: 300K_ENT_PH_DN

This run is a re-ranking of the Run 3 (300K_ENT_PH) (Section 4.3) in which we include the cosine distances between the article text and the description and the narrative.

$$score(R) = \Big(\cos(Q_{Entities}, R_{Entities}) + \cos(Q_{Events}, R_{Events}) + \cos(Q_{BodyText}, R_{Narrative}) + \cos(Q_{BodyText}, R_{Description})\Big)/4 \quad (7)$$

4.5 Run 5: Lambda_narr

This run consisted in starting from the outcome produced by the Lambda approach (Section 4.2) and re-ranking the recommended articles using the narrative field. The narrative field is an element provided by TREC organizers, as shown in Figure 1. It offers a summary of what background is expected.

First, we calculated the cosine similarity between the narrative field dense vector and the recommended article's body dense vector. Then, we used a weighted harmonic mean to merge the rankings produced by the cosine similarity (R_{Narr}) and those produced by the Lambda approach (R_{Lambda}):

$$Lambda\_narr = \frac{3.25R_{Lambda}^{-1}R_{Narr}^{-1}}{(2.25R_{Lambda}^{-1}) + R_{Narr}^{-1}} \quad (8)$$

We used the reciprocal of all the rankings R, to indicate that the lower the rankings, i.e. 1st, the better. In Equation 8 we give priority to the ranking produced by R_{Narr} over R_{Lambda}.

To produce the final ranking, we sort $Lambda\_narr$ scores in descending order.

4.6 Subtopics Approaches

Regarding the background of articles following the subtopics, we submitted five different approaches, that are an extension of the previously described ones.

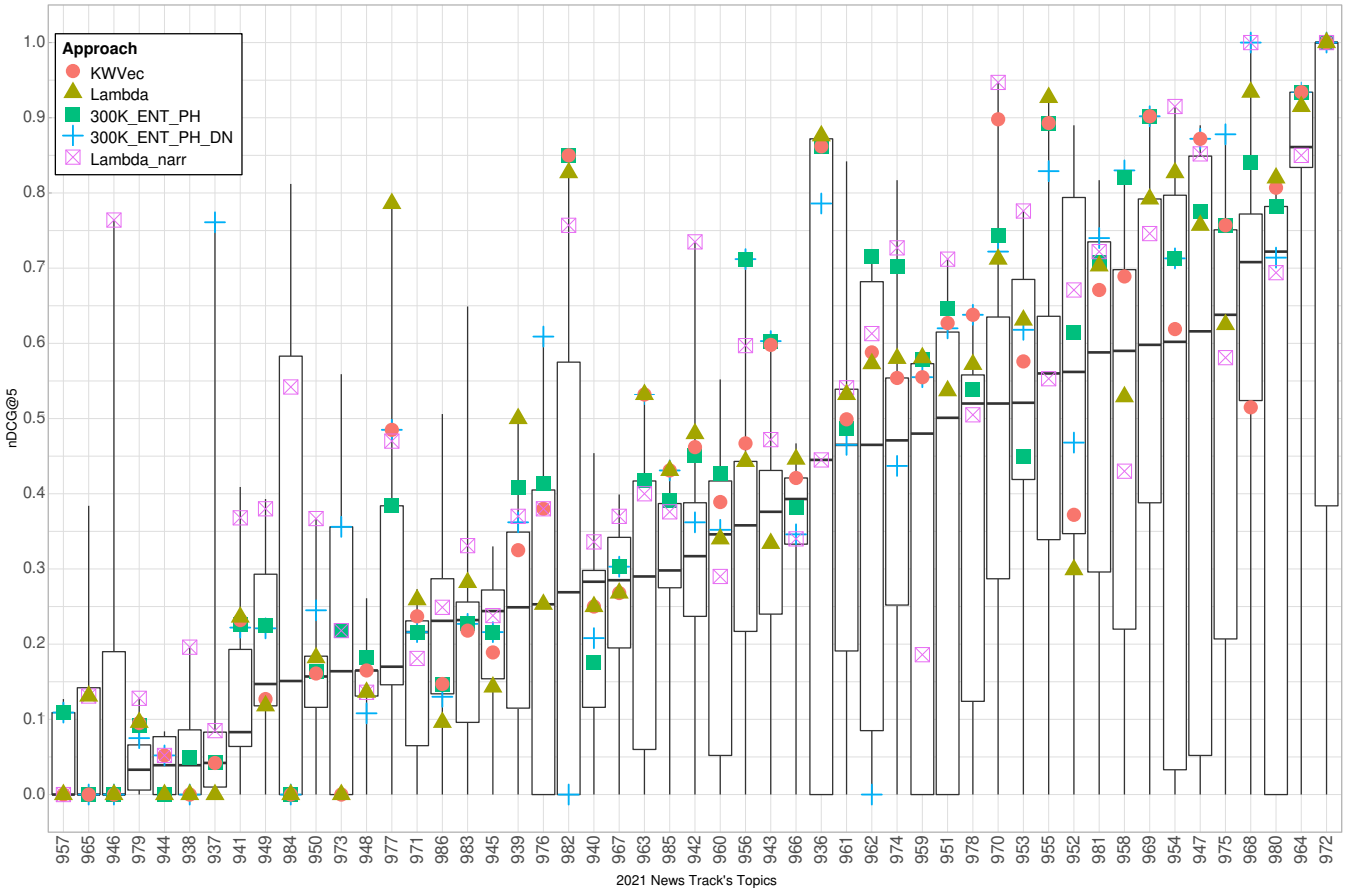

Figure 2: Boxplots of nDCG@5 score distribution for each topic based on all News Track submissions. The topics are sorted by their median nDCG@5. We present as well the nDCG@5 scores gotten by each of our approaches.

Run 1 (KWVec_sub): For this approach, we made use of the ranking produced by KWVec (Section 4.1) and re-ranked the recommended articles according to their cosine similarity with the subtopic. The re-ranking was done using the same ideas used in Section 4.5. Similarly, we applied a modified version of Equation 8:

$$KWVec\_sub = \frac{3.25R^{-1}_{KWVec}R^{-1}_{subtopic}}{(2.25R^{-1}_{KWVec}) + R^{-1}_{subtopic}} \quad (9)$$

Run 2 (Lambda_sub): This run is similar to KWVec_sub. However, instead of using the outcomes produced by KWVec, we make use of the outcomes produced by Lambda (Section 4.2). We also use Equation 9 with the respective changes to use R_{Lambda} instead of R_{KWVec}.

Runs 3, 4, & 5: Run 3 is a re-ranking of the initial runs to which the cosine similarity between the text body of the query article and the text of the subtopic are added. Runs 4 and 5 have the entities and the events removed, respectively.

$$score(R) = \Big(\cos(Q_{Entities}, R_{Entities}) + \cos(Q_{Events}, R_{Events}) + \cos(Q_{BodyText}, R_{Narrative}) + \cos(Q_{BodyText}, R_{Description}) + + \cos(Q_{BodyText}, R_{SubtopicText}) \Big)/5 \quad (10)$$

5 RESULTS

In Figure 2, we present, for each 2021 topic, the distribution of nDCG@5 scores calculated from all the submissions along with the scores obtained by each of our approaches. We can notice that for some topics, e.g. *957* or *979*, it was very hard to predict a good background article for all the participants. In these cases, the median is not only very low, but the full distribution is quite compact and close to zero. This contrasts with other topics, like *946*, *937* and *977*, where despite having a low median, at least one of our approaches managed to reach values similar or equal to the maximum nDCG@5 score. Finally, we can observe that for some topics it was easy to predict background articles for most participants, such as topic *964* and *972*.

In Table 3 we present a summary of Figure 2, where we indicate the number of nDCG@5 scores, produced by our runs for each topic, found within each nDCG@5 quartiles. It should be noted, that in Table 3, if the value associated with a quartile was equal to another one, e.g. $Q_0 = Q_1$, like in topic *946*, the score was assigned to the quartile closest to the median one (Q_2).

Based on the results present in Table 3, we can determine that the recommendations produced by our approaches generated an nDCG@5 greater than the participants' median in at least 60% of the topics. Specifically, KWVec 66.6%, Lambda 60.7%, 300K_ENT_PH 74.5%, 300K_ENT_PH_DN 64.7% and Lambda_narr 70.5%. Moreover, all our approaches achieved the maximum score nDCG@5 score in at least 9.8% of the topics, topped by 300K_ENT_PH_DN with a 21.5%.

In Figure 3, we present the distribution of nDCG@5 scores generated by each of our explored approaches. We can notice in Figure 3, that the best system has been KWVec with an nDCG@5 median of 0.462. We can further observe that for KWVec and Lambda the distribution of the scores tends to be negatively skewed, while 300K_ENT_PH, 300K_ENT_PH_DN, and Lambda_narr are positively skewed.

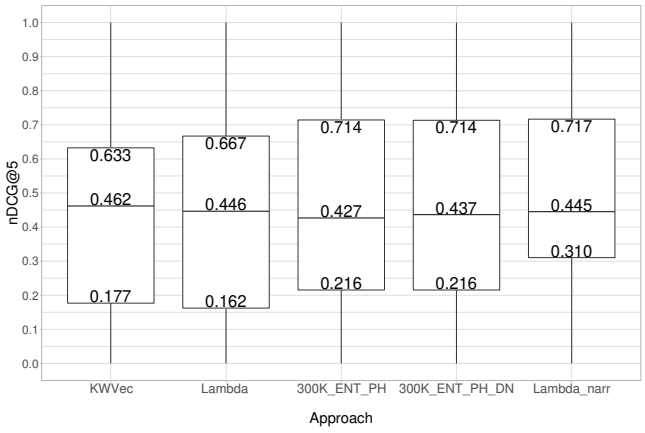

Figure 3: Boxplots representing the distribution of nDCG@5 scores obtained by each explored approach. We include the numerical values for the first, second (median), and third quartiles.

6 DISCUSSION

One aspect that we noticed from KWVec during the experimentation with the 2020 topics is that the scores obtained by the cosine similarity were, in multiple cases, diminished by the scores obtained by keywords. In other words, the final score given by ODFE to a document came mostly from the keywords, and not from the cosine similarity calculations. This is why we added weight (250) to Equation 2. However, this number was chosen experimentally based on 2020 topics.

Due to this, we decided to explore the Lambda approach, where we expected that the Bayesian optimization could automatically determine the weights (λ) that should be used to merge the scores to get the best nDCG scores. Nonetheless, the performance of Lambda did not surpass that of KWVec, even if similar queries were used along with more specific ones.

There are multiple possible reasons why the Lambda approach did not surpass KWVec's performance. In the first place, for training, we relied on data from previous years which were produced using different methods. This means that for training we used documents that on occasions would not be retrieved by our queries as highly relevant, and therefore we introduce a bias in the weights of certain queries. Sometimes the top retrieved documents by our queries had to be removed from the training as we did not know their gold standard relevance. In spite of the fact that we manually annotated some top retrieved documents, for which we did not have a gold standard relevance score, the additional scored documents seemed to be insufficient for the training. This last point can be because of the annotation quality and variety, as it focused on one type of query, the title-title similarity, and the process was done by just one person, who could naturally be biased.

With respect to Lambda_narr, although it did not surpass KWVec performance, we can determine from Figure 3, that re-ranking the documents according to the narrative produced interesting results. We managed to set 50% of the nDCG@5 scores within a smaller and better range of values $[0.310, 0.717]$ with respect to the other approaches. Nonetheless, most of the Lambda_narr scores were closer to 0.310 rather than to 0.717, creating a positively skewed distribution that affected its median. Despite this, Lambda_narr's median, 0.445, is similar to the one set by its parent, the Lambda approach, with an nDCG@5 of 0.446.

In regard to the re-rankings enhanced with entities and events or narratives, both runs, 300K_ENT_PH and 300K_PH_DN are rather homogeneous, with the same range of values $[0.126, 0.714]$, and slightly similar median values. However, both Q_1 and Q_3 nDCG@5 scores surpass those of KWVec and Lambda.

It is interesting to observe that despite the fact that the model 300K_PH_DN achieved the largest number of topics with a maximum score, 11 as seen in Table 3, its median did not surpass KWVec. It is possible that the 300K_PH_DN median was severely affected by the nDCG@5 scores of topics 982 and 962, which were zero, as seen in Figure 2.

In all the cases, the results obtained by 300K_ENT_PH and 300K_PH_DN, and especially the latter, could indicate that background linking could benefit from augmenting the articles with additional extracted information, such as named entities and events.

7 CONCLUSION

In this work, we presented the participation of the *Laboratory L3i, University of La Rochelle*, at TREC 2021 News Track Background Linking. From our participation, we noticed that, despite the existence of embeddings from fine-tuned language models such as Sentence BERT [28], keywords are still one of the most powerful sources of knowledge to rank news articles. Also, we observed that extracting additional textual elements, such as named entities and events, can be useful and, in some cases, they can provide unique information that will bring out the most relevant articles. Furthermore, re-ranking news articles based on simple inputs from journalists, like a summary of what it is expected to retrieve, can improve the performance of a news background linking system. Regarding training a

Table 3: Number of topics' nDCG@5 score found in each topic's quartile (Q) calculated by TREC organizers. The value in brackets represents the percentage of topics. Q_0 is the minimum score, Q_2 is the median and Q_4 is the maximum score.

Run	$x = Q_0$	$Q_0 < x < Q_1$	$Q_1 \leq x < Q_2$	$x = Q_2$	$Q_2 < x \leq Q_3$	$Q_3 < Q_4$	$x = Q_4$
KWVec	0 (0.0)	1 (1.9)	10 (19.6)	6 (11.7)	16 (31.3)	13 (25.4)	5 (9.8)
Lambda	1 (1.9)	3 (5.8)	12 (23.5)	4 (7.8)	11 (21.5)	13 (25.4)	7 (13.7)
300K_ENT_PH	0 (0.0)	0 (0.0)	8 (15.6)	5 (9.8)	17 (33.3)	16 (31.3)	5 (9.8)
300K_ENT_PH_DN	1 (1.9)	2 (3.9)	10 (19.6)	5 (9.8)	12 (23.5)	10 (19.6)	11 (21.5)
Lambda_narr	0 (0.0)	0 (0.0)	12 (23.5)	3 (5.8)	12 (23.5)	14 (27.4)	10 (19.6)

model which optimizes weights of different queries is still difficult. Nonetheless, based on our results, it could be feasible, but more annotated data would be necessary to reduce bias.

Finally, as future work, we would like to apply the previously explored background linking approaches in less-represented languages, such as Croatian and Finnish, through the Embeddia project [26].

ACKNOWLEDGMENTS

This work has been supported by the European Union's Horizon 2020 research and innovation program under grants 770299 (NewsEye) and 825153 (Embeddia), and by the ANNA and Termitrad projects funded by the Nouvelle-Aquitaine Region.

REFERENCES

[1] Ahmet Aker, Emina Kurtic, Mark Hepple, Rob Gaizauskas, and Giuseppe Di Fabbrizio. 2015. Comment-to-Article Linking in the Online News Domain. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Association for Computational Linguistics, Prague, Czech Republic, 245–249. https://doi.org/10.18653/v1/W15-4635

[2] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 2895–2905. https://doi.org/10.18653/v1/P19-1279

[3] Emanuela Boros and Antoine Doucet. 2021. Transformer-based Methods for Recognizing Ultra Fine-grained Entities (RUFES). *arXiv preprint arXiv:2104.06048* (2021).

[4] Emanuela Boros, Ahmed Hamdi, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Jose G. Moreno, Nicolas Sidere, and Antoine Doucet. 2020. Alleviating Digitization Errors in Named Entity Recognition for Historical Documents. In *Proceedings of the 24th Conference on Computational Natural Language Learning (CoNLL)*, Raquel Fernández and Tal Linzen (Eds.). Association for Computational Linguistics, Online, 431–441. https://doi.org/10.18653/v1/2020.conll-1.35

[5] Emanuela Boros, Ahmed Hamdi, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Jose Moreno, Nicolas Sidere, and Antoine Doucet. 2021. Atténuer les erreurs de numérisation dans la reconnaissance d'entités nommées pour les documents historiques. In *COnférence en Recherche d'Informations et Applications-CORIA 2021, French Information Retrieval Conference,*. CORIA, Online.

[6] Emanuela Boros, Jose G. Moreno, and Antoine Doucet. 2021. Event Detection with Entity Markers. In *Advances in Information Retrieval*, Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin Potthast, and Fabrizio Sebastiani (Eds.). Springer International Publishing, Cham, 233–240.

[7] Emanuela Boros, Elvys Linhares Pontes, Luis Adrián Cabrera-Diego, Ahmed Hamdi, José Moreno, Nicolas Sidère, and Antoine Doucet. 2020. Robust named entity recognition and linking on historical multilingual documents. In *Working Notes of CLEF 2020 - Conference and Labs of the Evaluation Forum*, Linda Cappellato, Carsten Eickhoff, Nicola Ferro, and Aurélie Névéol (Eds.), Vol. 2696. CEUR-WS, Thessaloniki, Greece, 1–17.

[8] Luis Adrián Cabrera-Diego, Stéphane Huet, Bassam Jabaian, Alejandro Molina, Juan-Manuel Torres-Moreno, Marc El-Bèze, and Barthélémy Durette. 2014. Algorithmes de classification et d'optimisation : participation du LIA/ADOC à DEFT'14. In *Actes du dixième DÉfi Fouille de Textes*. Association pour le Traitement Automatique des Langues, Marseille, France, 53–60.

[9] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. YAKE! Keyword extraction from single documents using multiple local features. *Information Sciences* 509 (2020), 257–289. https://doi.org/10.1016/j.ins.2019.09.013

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *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)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, Minneapolis, Minnesota, USA, 4171–4186. https://doi.org/10.18653/v1/n19-1423

[11] Elisa Shearer. 2021. More than eight-in-ten Americans get news from digital devices. *Pew Research Center* (Dec. 2021). https://www.pewresearch.org/fact-tank/2021/01/12/more-than-eight-in-ten-americans-get-news-from-digital-devices/

[12] Marwa Essam and Tamer Elsayed. 2020. Why is That a Background Article: A Qualitative Analysis of Relevance for News Background Linking. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland) *(CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 2009–2012. https://doi.org/10.1145/3340531.3412120

[13] Galen Stocking and Maya Khuzam. 2021. Digital News Fact Sheet. *Pew Research Center* (June 2021). https://www.pewresearch.org/journalism/fact-sheet/digital-news/

[14] Weiwei Guo, Hao Li, Heng Ji, and Mona Diab. 2013. Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Sofia, Bulgaria, 239–249. https://aclanthology.org/P13-1024

[15] Heng Ji, Avirup Sil, Hoa Trang Dang, Ian Soboroff, Joel Nothman, and Sydney Informatics Hub. 2019. Overview of TAC-KBP2019 Fine-grained Entity Extraction. In *2019 Text Analysis Conference Proceedings*. NIST, Gaithersburg, Maryland, USA.

[16] Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, Brussels, Belgium, 133–142. https://doi.org/10.18653/v1/K18-2013

[17] Pavel Khloponin and Leila Kosseim. 2020. The CLaC System at the TREC 2020 News Track. In *Proceedings of the 29th Text REtrieval Conference (TREC 2020)*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. SP 1266. NIST, Online. https://trec.nist.gov/pubs/trec29/papers/CLaC.N.pdf

[18] Pavel Khloponin and Leila Kosseim. 2021. Using Document Embeddings for Background Linking of News Articles. In *Natural Language Processing and Information Systems*, Elisabeth Métais, Farid Meziane, Helmut Horacek, and Epaminondas Kapetanios (Eds.). Springer International Publishing, Cham, 317–329.

[19] Boshko Koloski, Elaine Zosa, Timen Stepišnik-Perdih, Blaž Škrlj, Tarmo Paju, and Senja Pollak. 2021. Interesting cross-border news discovery using cross-lingual article linking and document similarity. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*. Association for Computational Linguistics, Online, 116–120. https://aclanthology.org/2021.hackashop-1.16

[20] Kuang Lu and Hui Fang. 2019. Leveraging Entities in Background Document Retrieval for News Articles. In *Proceedings of the 28th Text REtrieval Conference (TREC 2019)*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. SP 1250. NIST, Online. https://trec.nist.gov/pubs/trec28/trec2019.html

[21] Jose G Moreno, Emanuela Boros, and Antoine Doucet. 2020. TLR at the NTCIR-15 FinNum-2 Task: Improving Text Classifiers for Numeral Attachment in Financial Social Data. In *Proceedings of the 15th NTCIR Conference on Evaluation of Information Access Technologies*. National Institute of Informatics, Tokyo, Japan, 8–11.

[22] José G. Moreno, Antoine Doucet, and Brigitte Grau. 2021. Relation Classification via Relation Validation. In *Proceedings of the 6th Workshop on Semantic Deep*

Learning (SemDeep-6). Association for Computational Linguistics, Online, 20–27. https://aclanthology.org/2021.semdeep-1.4

[23] Jonas Močkus, Vytautas Tiešis, and Antanas Žilinskas. 1978. The application of Bayesian methods for seeking the extremum. In *Towards Global Optimisation*, George Philip Szegö and Laurence Charles Ward Dixon (Eds.). Vol. 2. North-Holland, Amsterdam, The Netherlands, 117–128.

[24] Shahrzad Naseri, John Foley, and James Allan. 2018. UMass at TREC 2018: CAR, Common Core and News Tracks.. In *Proceedings of the 27th Text REtrieval Conference (TREC 2018)*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. SP 500-331. NIST, Gaithersburg, Maryland, USA. https://trec.nist.gov/pubs/trec27/papers/Overview-News.pdf

[25] Joel Nothman, Matthew Honnibal, Ben Hachey, and James R. Curran. 2012. Event Linking: Grounding Event Reference in a News Archive. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, Jeju Island, Korea, 228–232. https://aclanthology.org/P12-2045

[26] Senja Pollak, Marko Robnik-Šikonja, Matthew Purver, Michele Boggia, Ravi Shekhar, Marko Pranjić, Salla Salmela, Ivar Krustok, Tarmo Paju, Carl-Gustav Linden, Leo Leppänen, Elaine Zosa, Matej Ulčar, Linda Freienthal, Silver Traat, Luis Adrián Cabrera-Diego, Matej Martinc, Nada Lavrač, Blaž Škrlj, Martin Žnidaršič, Andraž Pelicon, Boshko Koloski, Vid Podpečan, Janez Kranjc, Shane Sheehan, Emanuela Boros, Jose G. Moreno, Antoine Doucet, and Hannu Toivonen. 2021. EMBEDDIA Tools, Datasets and Challenges: Resources and Hackathon Contributions. In *Proceedings of the EACL Hackashop on News Media Content Analysis and Automated Report Generation*. Association for Computational Linguistics, Online, 99–109. https://www.aclweb.org/anthology/2021.hackashop-1.14

[27] Marko Pranjić, Vid Podpečan, Marko Robnik-Šikonja, and Senja Pollak. 2020. Evaluation of related news recommendations using document similarity methods. In *Proceedings of the Conference on Language Technologies and Digital Humanities (JDTH2020)*, Darja Fišer and Tomaž Erjavec (Eds.). Inštitut za novejšo zgodovino, Ljubljana, Slovenia, 81–86. https://doi.org/10.5281/zenodo.4059710

[28] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 3982–3992. https://doi.org/10.18653/v1/D19-1410

[29] Shudong Huang, Ian Soboroff, and Donna Harman. 2018. TREC 2018 News Track. In *Proceedings of the Second International Workshop on Recent Trends in News Information Retrieval (NewsIR'18)*, Dyaa Albakour, David Corney, Julio Gonzalo, Miguel Martinez, Barbara Poblete, and Andreas Valochas (Eds.), Vol. 2079. CEUR Workshop Proceedings, Grenoble, France. http://ceur-ws.org/Vol-2079/paper12.pdf

[30] Ian Soboroff, Shudong Huang, and Donna Harman. 2018. TREC 2018 News Track Overview. In *Proceedings of the 27th Text REtrieval Conference (TREC 2018)*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. SP 500-331. NIST, Gaithersburg, Maryland, USA. https://trec.nist.gov/pubs/trec27/papers/Overview-News.pdf

[31] Ian Soboroff, Shudong Huang, and Donna Harman. 2020. TREC 2020 News Track Overview. In *Proceedings of the 29th Text REtrieval Conference (TREC 2020)*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. SP 1266. NIST, Online. https://trec.nist.gov/pubs/trec29/papers/OVERVIEW.N.pdf

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc., Long Beach, CA, USA, 5998–6008.

[33] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, and Ming Zhou. 2020. MIND: A Large-scale Dataset for News Recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 3597–3606. https://doi.org/10.18653/v1/2020.acl-main.331