The CLaC System at the TREC 2019 News Track

Pavel Khloponin and Leila Kosseim

ClaC Lab
Department of Computer Science and Software Engineering
Gina Cody School of Engineering and Computer Science
Concordia University
Montreal QC, Canada
p_khlopo@encs.concordia.ca, leila.kosseim@concordia.ca

Abstract. This paper describes our approach to the TREC 2019 News Track. The goal of the News Track is to provide background links and entity linking to target news articles within a collection of articles. Our approach first represents all articles in the collection using Doc2Vec embeddings then computes the cosine similarity between the target article with all other articles in the collection and outputs the 100 closest ones. Although simple, this approach allows minimum pre-processing and is agnostic to the content of the documents. The overall results of the shared tasks are below the median with a nDCG@5 of 0.4298, however, for specific topics, the approach achieved the best scores among all participating teams.

Keywords: News TREC · Doc2Vec · Cosine similarity.

1 Introduction

Given the sheer number of electronic sources of news available today, it is important to develop approaches for the automatic recommendation of contextual information for users to better understand a news article.

In order to address this need, since 2018, the News Track at TREC has proposed two related shared tasks: background linking and entity ranking (Soboroff, Huang, and Harman 2018). The goal of the background linking task is to provide relevant background information to news articles through the identification of related articles. On the other hand, entity ranking focuses on providing a list of names, concepts, artifacts, etc. mentioned in news articles, which will help readers better understand the news. For our first participation to the News Track, we only participated to the first task: background linking.

For the 2019 background linking task, the data set provided by NIST consists of a collection of about 600,000 news articles from the Washington Post. Given a search topic, which is itself an article from the corpus selected by TREC organizers, participants need to select up to 100 related articles from the corpus and output them from the most related to the least related. For evaluation purposes, only the top 5 articles from each list are considered. A 5 point rank is manually assigned by NIST assessors for each of the top 5 articles during the evaluation step, where rank is between 0 (little or no useful information) and 4 (must appear in recommendations or critical context will be missed). The total score is based on the nDCG@5 (Järvelin and Kekäläinen 2002) metric with the gain metric 2^{rank}.

One important criteria for judging related articles is diversity. Because we did not have a clear understanding of the notion of article diversity in the news recommendation context we did not specifically address this aspect of the task.

For the official evaluation, 60 search topics were used. Prior to the evaluation, the organizers also provided 50 search topics and their corresponding manually evaluated results (that is, all articles evaluated manually with a rank from 0 to 4) from the TREC News 2018, which used the same article collection. We used these 50 topics and 8508 evaluated backlinks for validation purposes.

2 Data Preprocessing

The TREC News 2019 organisers provided a document collection of 595,037 news articles from The Washington Post published between 2012 and 2017. This document collection was the same provided at the 2018 edition of the task but with exact duplicate articles removed. Each news article is stored in "JSON-lines" format and represented as a single line of JSON. Each document contains 8 types of meta-information:

1. id
2. article URL
3. title
4. author
5. publication date
6. type (blog post or article)
7. news source
8. content field

Apart from the id (#1) field, which was used for identification purposes, only the article title (#3) and the text extracted from the content field (#8) were considered. The title was prepended to the content and processed as a single document. The content itself was stored in a form of content blocks, where each content block can be a text paragraph, an image, a video, a tweet, a citation, etc. Each content block itself may contain meta-information (up to 133 different fields), such as MIME-type, type, kicker (category), content, subtype, source, URLs, etc. Based on this meta-information we identified blocks with frequently appearing content types and checked if they have any useful text descriptions, and kept for further processing only blocks with paragraphs, image captions, headers, and quotes. Blocks were further cleaned from embeds, links, images, and other HTML tags, preserving only plain text.

original number of articles	595,037	
near-duplicates	36,359	preserved
sandbox content	873	preserved
“lorem ipsum” articles	81	preserved
wire and opinion articles	23,074	removed
articles used to build the models	571,963	
average size of article (characters)	11,714	
average size after pre-processing (characters)	4,458	

Table 1. Statistics of the TREC News document collection

NIST required participants to ignore wire articles, editorial content and opinion posts, which have “Opinion”, “Letters to the Editor”, or “The Post’s View” values in the content meta-information block with type ”kicker”. Due to this, the initial set of 595,037 articles was reduced by 23,074 to 571,963 items. This is shown in Table 1.

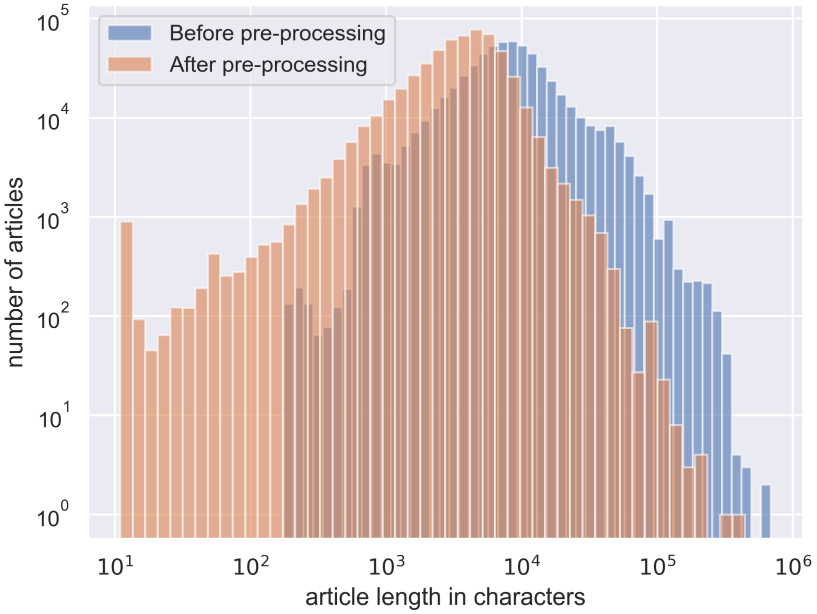

Fig. 1. Length distribution of the articles in the collection

As shown in Table 1, many draft articles (content preview or articles demonstrating website functionality) were discovered during data exploration: 873 articles with URL path starting with “/test/wp/”, presumably indicating content taken from the section of the website not intended to be public (content playground), and 81 document with “Lorem ipsum...” (common placeholder text) content in the article text. They were preserved in the dataset.

As shown in Table 1, the 571,963 documents considered have an average length of 11,417 characters, but only 4,458 after pre-processing. Figure 1 shows the distribution of the article lengths before and after pre-processing. Some articles composed almost entirely of meta-information and have very little text inside (short statements, video players, cited tweets, etc.), while others have very long texts (transcripts of debates, conferences, testimony, crime reports). As Figure 1 shows, the majority have between 1,000 and 10,000 characters.

3 Approach

3.1 Document Representation

After pre-processing, we used the Doc2Vec distributed representation (Le and Mikolov 2014) to represent each document. The rational for this choice was our expectation that related articles would be closer to each other in vector space than unrelated articles. Hence, document vector distance would be a good approximation of document content relatedness. Search topics and manually ranked documents from 2018 (see Section 1) were used as a validation set to select among different combinations of parameters for text processing and model parameters. For creating the document vectors: stopwords were filtered using the NLTK English stopwords list (Bird, Klein, and Loper 2009), numbers were replaced with the “NUM” token, and the text was case folded. Embeddings were built using Doc2Vec PV-DM embedding with 10 epochs. Using a different size of embeddings significantly changed the produced backlinks and their position in the list. Two different sizes for embeddings were selected: 100 and 300.

3.2 Proximity Measure

After obtaining the embedding weights for the articles, the proximity between two documents vectors was computed. Due to lack of time and resources only the cosine distance was used. Given two document vectors α and β, the cosine of the angle θ between α and β is computed as:

$$cos(\theta) = \frac{\alpha \cdot \beta}{||\alpha|| ||\beta||} = \frac{\sum_{i=1}^{n} \alpha_i \beta_i}{\sqrt{\sum_{i=1}^{n} \alpha_i^2} \sqrt{\sum_{i=1}^{n} \beta_i^2}}$$

where values close to -1 will, hopefully, correspond to documents with opposite meaning, close to 0 to documents on uncorrelated topics and close 1 to documents with similar topics. The cosine value is directly used as the relevancy score in the final output of the system.

3.3 Validation

In order to set the values of the various hyperparameters, the models were evaluated on with the 2018 data given. To evaluate our models, we generated the top 5 backlinks for each topic and computed nDCG@5 when compared with 2018 dataset. If a generated backlink was not in the 2018 validation set, we assigned it a rank of 0 (not relevant backlink) to calculate nDCG@5. As shown in Table 2 the model with embedding size of 100 dimensions achieved average an nDCG@5 of 0.3378, and with an embedding size of 300 dimensions achieved average an nDCG@5 of 0.3032. The fact that about 30% of the returned links did not appear in the validation set makes these numbers only lower bounds to the total scores they would have achieved. It also does not give us confidence about which of the models will perform better on the 2019 search topics.

The organizers reported that the top team reached the median nDCG@5 across all topics of about 0.45 (Soboroff, Huang, and Harman 2018). The exact value for the median nDCG@5 performance across all teams was not reported in the overview paper. However, two papers from TREC 2018 News Track proceedings mentioned it to be equal to 0.3448 (Bimantara et al. 2018) and 0.2792 (Lu and Fang 2018). It is not clear how this was calculated. Nevertheless, it gives a rough estimate for our model performance with this year data.

Run	**nDCG@5**
`clac_100_cos`	0.3378
`clac_300_cos`	0.3032

Table 2. Results with 2018 data

3.4 Submitted Runs

Two runs were submitted: `clac_100_cos` and `clac_300_cos`; based on the models described in Section 3.1 and the proximity measure from Section 3.2 with embeddings sizes 100 and 300 respectively.

3.5 Performance and Model Size

The models were built in python 3 with the Gensim library on a desktop computer with Intel® Core™ i7-7800X CPU @ 3.50GHz with 6 cores and 12 threads. One epoch of training the model takes about 14 min. Hence the total training time for 10 epochs is about 2.5 hours. The model size on the disk takes 1.2GB. Generating results for 60 topics took 9 sec including 6 sec for the reading model to memory. On documents with pre-computed embeddings, the model was able to generate predictions for 28 topics per second for documents with known embedding vector and 16 topics per second for new articles including the generation of dense vectors. These low resource requirements make the model applicable in a production environment.

4 Results and Analysis

4.1 Official Scores

As indicated in Section 3.4, we have submitted two runs: `clac_100_cos` and `clac_300_cos`. For each topic, NIST provided us with our official scores as well as the minimum, maximum and median scores across all submitted runs. Full judgment data was not available at the time of writing this paper, but we can compare our runs with hypothetical participants with maximum, minimum and median scores for each topic. As shown in Table 3, overall, both our runs performed below the collective median with scores of 0.4057 and 0.4298 compared to 0.5295. This was to be expected given the simplicity of the approach and our performance on the validation data from 2018 (see Section 3.3).

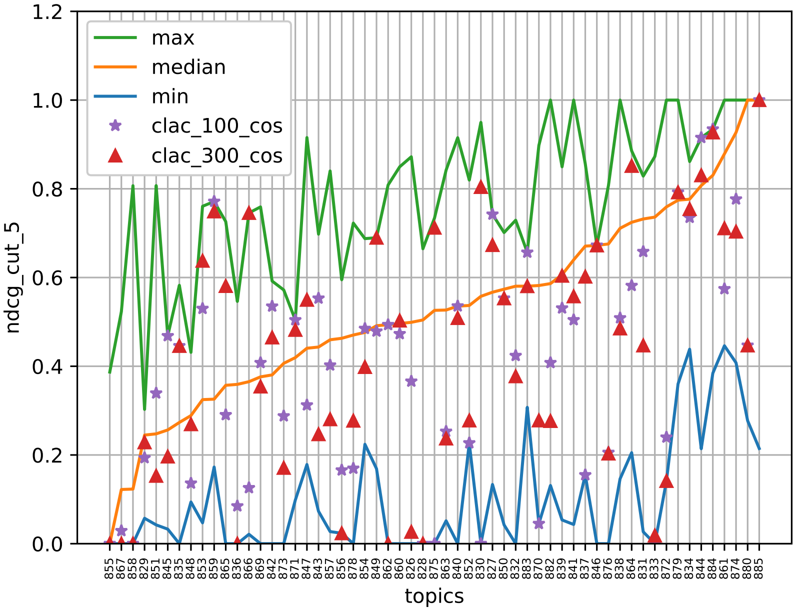

Fig. 2. Results per topic in increasing order of median nDCG@5

However, surprisingly `clac_300_cos` performed better than `clac_100_cos` where as the results with the 2018 dataset predicted the opposite (see Table 2)

Figure 2 shows the scores for each topic, in decreasing order of the median score. Difficult topics for all participants appear on the left hand side of Figure 2, while easier topics are on the right-hand side. For a few topics our approach had the lowest performance among all runs (8 topics for `clac_100_cos` and 9 topics

Run	nDCG@5
clac_100_cos	0.4057
clac_300_cos	0.4298
TREC max	0.7737
TREC median	0.5295
TREC min	0.1002

Table 3. 2019 overall results of our runs

for clac_300_cos), they are all situated on the minimum line. On the other hand, for a few other topics, all points are situated on the maximum line (8 topics for clac_100_cos and 6 topics for clac_300_cos) hence we achieved the best performance among all News TREC runs.

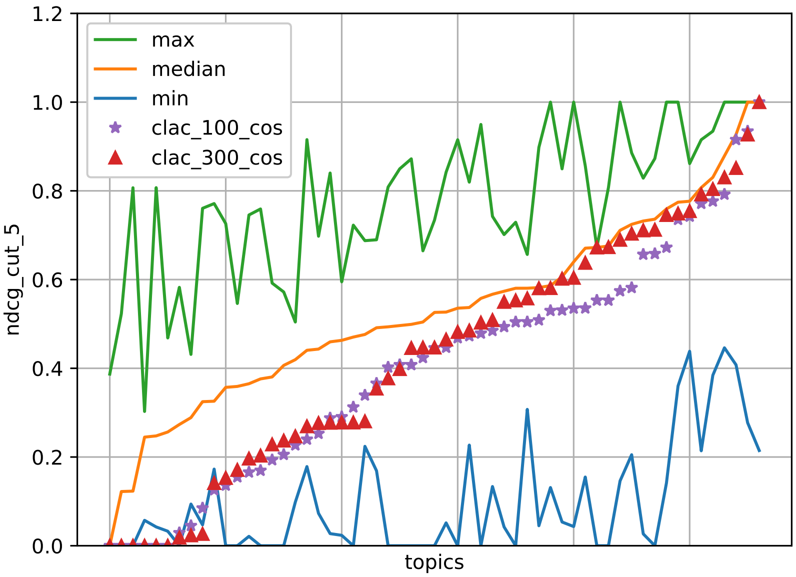

Fig. 3. Results sorted by nDCG@5 plotted on top of aggregated results (see Figure 2), topics not necessarily match.

Figure 3 shows the scores per topic in increasing order of nDCG@5. This figure helps to compare our runs with the collective median and seems to show that clac_300_cos, in general, performs better than clac_100_cos.

4.2 Analysis

As shown in Figure 2, one of our best performance is for topic 859 (663c2790): Lottery sales, casino revenue a billion-dollar boon for Maryland. For this topic we achieved an nDCG@5 of 0.7713 for `clac_100_cos` and 0.7490 for `clac_300_cos` whereas the median was only 0.3381. The model's outputs for topic 859 are listed in Figure 4. All retrieved articles discuss revenue and bushiness related activities of casinos in specific geographic location, share the same narrative and keyword.

ID	Score	Title
23a664d0	0.7484	Maryland casinos rake in nearly $87 million in October
ba6587d8	0.7476	Maryland casinos hit jackpot: States 4 venues collected $69 million in May
bf89d364	0.7331	Md. casinos see slight drop in overall revenue since last year
b67cd198	0.7199	Marylands casinos raked in $833 million in the past fiscal year
5f48c58a	0.7195	Md. casino revenue has jumped since MGM opened, but nearby venues took a hit

Fig. 4. Top 5 backlinks for topic 859 (663c2790) with calculated cosine similarity score `clac_100_cos`

On the other hand, one of our worst performance is for topic 833 (0e43fce6): Worried about MERS in South Korea? Visitors can (mostly) breathe easy. For this topic we achieved an nDCG@5 of only 0.0625 for `clac_100_cos` and 0.0716 for `clac_300_cos` with the median being 0.74075. The produced backlinks are listed in Figure 5. Even though the search topic and produced backlinks have a very similar narrative and are discussing some kind of danger (terrorist attack, Ebola or MERS[1]) localized geographically in a frame of traveling (planes, tickets, insurance) with medical personal (a nurse, a doctor or CDC[2]) involved in the context, our system failed to build an important relation between the articles, which is connected to MERS. Only the last of the top 5 backlinks mention

ID	Score	Title
9bd541d2	0.7270	Is it safe to travel to Paris and other European cities?
65612ad2	0.7017	The travel industry is taking precaution, but is it calming anxiety about disease?
46af8ae8	0.7006	Tips for traveling to Africa in the age of Ebola
4050a3da	0.6801	The most pressing summer travel questions answered
528fe4f7	0.6669	Why youre not going to get Ebola in the U.S.

Fig. 5. Top 5 backlinks for topic 833 (663c2790) with calculated cosine similarity score for `clac_100_cos`

[1] Middle East Respiratory Syndrome

[2] Centers for Disease Control and Prevention

MERS. The cosine considers all vector dimensions to have the same importance in the distance calculation but this example seems to show that all dimensions should not necessarily have the same weight.

We also analysed content from the playground described in Section 2. Only 4 draft articles got into the submission files well below the cutoff line. We additionally discovered that many articles with the same URLs differ only in the last segment which is usually a number. These articles are either identical or have minor modifications or more details are provided. We assumed these numbers represented different versions of the same article. Hence, 36,359 articles could be additionally eliminated as duplicates based on these criteria. Unfortunately, this fact was discovered after the submission.

5 Conclusions and Future Work

This paper presented our model developed for the TREC 2019 News Track. The model uses a simple Doc2Vec representation and cosine similarity measure which allows for a plug and play approach with minimal tuning required. The model exploits the assumption that related articles should have close document vectors. However, the results of our runs seem to indicate that this assumption may not always hold as the model missed significant connections for the sake of extrinsic factors. The model could be more appropriate to look for near-duplicates and similar articles than backlinks or as a part of a more advanced pipeline. On the other hand, our models are very light-weight and only require 1.2GB space on disk and can generate backlinks for the entire collection in just 8 hours making it usable in a production setting. During the analysis of the results, it became obvious that the model is missing clues that could be obtained by giving more priority to named entities.

Acknowledgements

This work was financially supported by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Soboroff, Ian, Shudong Huang, and Donna Harman (2018). “2018 News Track Overview”. In: URL: https://trec.nist.gov/pubs/trec27/papers/Overview-News.pdf.

Järvelin, Kalervo and Jaana Kekäläinen (2002). “Cumulated Gain-based Evaluation of IR Techniques”. In: *ACM Transactions on Information Systems (TOIS)* 20.4, pp. 422–446. ISSN: 1046-8188.

Le, Quoc and Tomas Mikolov (2014). “Distributed Representations of Sentences and Documents”. In: *Proceedings of the 31st International Conference on Machine Learning*. Ed. by Eric P. Xing and Tony Jebara. Vol. 32. Proceedings of Machine Learning Research 2. Bejing, China: PMLR, pp. 1188–1196. URL: http://proceedings.mlr.press/v32/le14.html.

Bird, Steven, Ewan Klein, and Edward Loper (2009). *Natural Language Processing with Python*. 1st. O’Reilly Media, Inc. ISBN: 0596516495, 9780596516499.

Bimantara, Agra et al. (2018). “htw saar @ TREC 2018 News Track”. In: URL: https://trec.nist.gov/pubs/trec27/papers/htwsaar-N.pdf.

Lu, Kuang and Hui Fang (2018). “Paragraph as Lead - Finding Background Documents for News Articles”. In: URL: https://trec.nist.gov/pubs/trec27/papers/udel_fang-N.pdf.