Aspect Based Background Document Retrieval for News Articles

Kuang Lu and Hui Fang

Department of Electrical and Computer Engineering,
University of Delaware
{lukuang, hfang}@udel.edu

Abstract

Background information is essential for readers to understand news articles. Moreover, background information is often multi-faceted [4], which introduces extra complexity to the problem of background retrieval. In this year's News Track, we explored how to build **entity graphs** on news articles to identify aspects, and leverage the aspects to retrieve background information. More specifically, given a query news article, the entities in it and their relations are used to build the entity graph, and aspects are extracted using community analysis. Subsequently, the discovered aspects are individually used to retrieve background news articles, and the per-aspect results are merged to form the final background article list for the query article.

1 Introduction

News Track organizers and journalists designed the News Track in collaboration with the purpose of identifying news readers' search needs as well as providing the test bed to investigate techniques for the needs. In this year's News Track, we focused on the background linking task of the track, which is designed for the information need of background information.

According to Fox, a professional journalist and editor, background information can be basic information of the news story, or *connections* between the story and other related ones [4]. In other words, there can be multiple *aspects* for the background information. Based on this, our effort this year centered around how to identify the aspects, and how to leverage the aspects to perform background information retrieval. For aspect identification, we hypothesized that entities are strong indicators of aspects in that different aspects involve different sets of entities. Moreover, the set of entities of an aspect tend to co-occur more often than entities from different aspects. Based on these hypotheses, we built an **entity graph** for each query article where nodes are entities and edges represent co-occurrences of the entities. We then employed community analysis [2] to segment the graph into aspects (or communities in the context of community analysis). For background retrieval of a single aspect, two methods

were tested which are different in terms of the text representations of aspects used for retrieval. One method used only entities. However, in addition to entities, the other method used non-entity words surrounding them as well. Finally, we merged results of aspects by assigning weights to aspects based on how related they are to the main story of the article. The score of a background article is computed as the weighted sum of its scores of all aspects. This score is subsequently used to rank the background articles.

2 Data Processing

Our data pre-processing steps are the same as our participating runs of the last two years. We first adopted the de-duplication method from Bimantara et al. [1], which processed files in the lexical order of the file names. Documents were de-duplicated based on the document title, author name, and published date. In addition, articles belonging to "Opinion", "Letters to the Editor", and "The PostVíew" section were discarded in accordance with the guideline. An article that was not removed by the above steps was stored by its id, title, timestamp, and the aggregated text of all of its paragraphs.

For entity recognition, following the previous two years, DBpedia Spotlight [3] was used. DBpedia[1] is a knowledge base that contains structured information about Wikipedia pages. There is a unique DBpedia entity corresponding to a Wikipedia page and therefore we use the term Wiki entity and DBpedia entity interchangeably in the rest of the paper. DBpedia Spotlight can automatically annotate DBpedia entities from the text of the articles in the collection, which accomplishes our entity annotation goal. The identified entities were subsequently replaced by their canonical forms (e.g. the form appears in DBpedia), which are also provided by the toolkit. For example, "the Red Planet" and "Mars" are all mapped to the canonical form "Mars". We hope that this would help us to more accurately match the entities in the query articles and background articles. The parameter "Confidence" of the tool is set to 0.5. We treat the identified entities, either single-term or multi-term, as single words and use the Indri [5] toolkit to index the documents.

3 Methods

As mentioned before, our aspect based method consists of three steps: aspect identification, individual aspect result retrieval, and aspect result merging. In order to identify aspects, entity graphs were built using the Wiki entities. Edges were added between entities co-occurring in paragraphs. The weights of the edges were determined by the word distances between them in paragraphs. More specifically, the word distance between two entities e_1 and e_2 in a paragraph p was computed as:

$$W(e_1, e_2, p) = 1 - \frac{1 + \#\text{ of words between } e_1, e_2}{|p|}, \tag{1}$$

[1] https://wiki.dbpedia.org/

where $|p|$ is the length (i.e. the number of words) of the paragraph. The paragraph word distances of two entities in paragraphs that they co-occur were averaged to obtain the aggregated word distance, which was then used as the edge weight between e_1 and e_2. The Louvain method proposed by Blondel et al. [2] was applied to the entity graph of the article to segment the graph into aspects. This method is designed for community analysis, which attempts to separate a weighted network into densely connected communities/sub-graphs. It accomplishes this by approximately optimizing modularity, which measures how edges are concentrated within communities instead of between them. It is clear that the method can segment entities following our intuition of grouping the entities co-occur often into the same aspects.

For individual aspect result retrieval, as mentioned earlier, we applied two methods which either used only entities of the aspects, or both entities and non-entity words surround them. The surrounding words were obtained by setting a word window of size ten, applying the window to each occurrence of the entities of an aspect to obtain spans of text, and extracting all non-entity words from the union of the spans. The union of the spans was used so that an occurrence of a non-entity word could only be counted once. It is important to note that, in the individual aspect result retrieval step, not all documents in the collection were scored. Instead, we first obtained a result list for each query article by using a baseline method, and only scored the articles in the result of the baseline method for different aspects. The baseline method used all words in the article as the retrieval query. The query then was searched against the collection. A time filter was applied to remove results published after the query article. The top 100 articles from the remaining result were then scored by the aspects with the two methods mentioned above. Using a baseline and performing re-ranking on its results is not only more efficient but also can be more effective if the baseline method is reasonably effective and can filter out irrelevant results. In fact, the baseline was indeed shown to be effective in previous years' News Tracks as one of the top performing runs. The baseline method will be referred to as "all_words" in the remainder of the report, whereas the two aspect result retrieval methods are named as "aspect_entities" and "aspect_all_words", respectively.

The final step of our method, which is aspect result merging, requires assigning weights to aspects. In order to do that, we first obtained the language model of an aspect by using the text spans of the aspect that were produced by the "aspect_all_words" method to extract all words, both entity and non-entity. Maximum likelihood estimation of the aspect language model was computed on the extracted words. The language model of the whole article was obtained similarly by applying maximum likelihood estimation on all words of the article. We then computed the query likelihood of the aspect language model with respect to the article language model as the weight of the aspect. The rationale of it is that if it is very likely to observe the aspect from the whole article, it means that the aspect is important in discussing the mains story of the article, and therefore the aspect is more important and needs to be assigned with a higher weight. It is important to note that, after merging results of different aspects using such weights to obtain a score for each candidate document, we further combined this score with that from the "all_word" baseline. The intuition behind this is that it could ensure the retrieved articles discuss the aspects in the sense that is related to the query article. Linear interpolation was used to combine these two scores and the weights

Table 1: NDCG@5 for submitted runs

Method	**NDCG@5**
$udel\_fang\_AW$	0.5437
$udel\_fang\_CE$	0.5454
$udel\_fang\_CW$	0.5292

for the scores from the baseline method and proposed methods were tuned on the last two years' data and were set as 0.7 and 0.3, respectively.

4 Run Description and Results

We submitted three runs, all of which employed language modeling with Dirichlet smoothing as the retrieval method. The first run is a baseline run called udel_fang_AW, which was implemented using the "all_word" baseline. We also submitted two runs with the same aspect identification method and aspect result merging method, but different aspect background retrieval methods to test our idea of using aspects for background retrieval. The one using the "aspect_entities" method is called udel_fang_CE, and the other one using "aspect_all_words" is called udel_fang_CW. The effectiveness of the submitted runs are reported in Table 1 as NDCG@5.

As can be seen, no improvements can be observed of runs with the proposed methods compared to the baseline method. Moreover, the effectiveness decreases slightly when both entity and non-entity words are used, though the difference is not statistically significant at the level of 0.05 using paired student t-test.

5 Conclusion

In this year's News Track, we investigated identifying and using aspects in the query article to find background news articles. Although no benefits can be observed for the proposed methods, we believe this direction is still promising and plan to explore further in the future since there are potential improvements that can be applied to these methods. For instance, using *all* entities in an article to build the entity graph of the article for mining aspects might not be optimal since there are entities belonging to the main story other than the background aspects. Moreover, the weighting of aspects can be improved as well. Using the query likelihood of the aspect language model to the article language model might prioritize the aspects explained well in the article, which might not require additional information. However, readers might want to know more about the aspects that are mentioned but not discussed in detail in the original article.

References

[1] Bimantara, A., Blau, M., Engelhardt, K., Gerwert, J., Gottschalk, T., Lukosz, P., Piri, S., Shaft, N.S., Berberich, K.: htw saar trec 2018 news track. In: Proceedings of TREC 2018 (2018)

[2] Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment 2008(10), P10008 (2008)

[3] Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving efficiency and accuracy in multilingual entity extraction. In: Proceedings of the 9th International Conference on Semantic Systems (I-Semantics) (2013)

[4] Fox, W.: Writing the News. Iowa State University Press, 3 edn. (2001)

[5] Strohman, T., Metzler, D., Turtle, H., Croft, W.B.: Indri: A language model-based search engine for complex queries. In: Proceedings of the International Conference on Intelligent Analysis. vol. 2, pp. 2–6. Citeseer (2005)