Microblog Contextualization: Advantages and Limitations of a Multi-Sentence Compression Approach

Elvys Linhares Pontes[1,2,3], Stéphane Huet[1], and Juan-Manuel Torres-Moreno[1,2,3]

[1] LIA, Université d'Avignon et des Pays de Vaucluse, Avignon, 84000 France
[2] Polytechnique Montréal, Montréal (Québec) Canada
[3] Université du Québec à Montréal, Montréal (Québec) Canada
elvys.linhares-pontes@alumni.univ-avignon.fr

Abstract. The content analysis task of the MC2 CLEF 2017 lab aims to generate small summaries in four languages to contextualize microblogs. This paper analyzes the challenges of this task and also details the advantages and limitations of our approach using a cross-lingual compressive text summarization. We split this task in several subtasks in order to discuss their setup. In addition, we suggest an evaluation protocol to reduce the bias of the current metrics toward the approaches by extraction.

Keywords: Microblog Contextualization, Multi-Sentence Compression, Word Embedding, Wikipedia

1 Introduction

The MC2 CLEF 2017 [3] lab analyzed the context and the social impact of microblogs. This lab was composed of three main tasks: 1/ Content Analysis, 2/ Microblog Search, and 3/ Time Line Illustration. The Content Analysis task involved itself several items: classification, filtering, language recognition, localization, entity extraction, linking open data, and summarization of Wikipedia pages and microblogs. Specifically, the summarization item, on which we focus here, aims to generate a textual summary using Wikipedia pages to contextualize a microblog in four languages (English, French, Portuguese, and Spanish).

This paper aims to present the complexity and challenges of the MC2 task to contextualize microblogs in four languages. We also carry out an analysis of our last year's participation (named CLCTS) [5] that proposed a cross-language compressive text summarization method to extract information from several language versions of Wikipedia in order to enhance informativeness. Our approach analyzes this task in several subtasks, each being prone to errors; this requires to measure how each subtask acts on the quality of summaries. Therefore, we propose an evaluation protocol to evaluate this task in two ways: end-to-end and by subtask.

This paper is organized as follows. Section 2 briefly describes a baseline approach and an overview of the CLCTS architecture to tackle the MC2 task.

Next, in Sections 3 and 4, we analyze the challenges of this task, the advantages and limitations of our approach. Then, we propose a protocol to evaluate this task in several ways in Section 5. Finally, we compare our approach with other state-of-the-art methods and we make final conclusions in Section 6 and 7, respectively.

2 System Architecture

A simple baseline for the MC2 task aims to retrieve information about a festival in a microblog from the Wikipedia databases in four languages (English, French, Portuguese, and Spanish). Then, this system selects the most relevant sentences that describe a festival to generate a short summary of 120 words independently for each language version. However, this approach does not cross-check the facts between languages and an extractive summarization may contain several irrelevant words that reduce the informativeness of summaries.

In order to improve the informativeness, we jointly take into account several language versions of Wikipedia and the sentences are compressed in order to retain only the relevant information. However, this analysis increases the complexity of the MC2 task. Considering these problems, we divided this task into subtasks. In this regard, we present their challenges, advantages, and limitations.

We first divided our system into two main parts. The first one (see Fig. 1, left side) aims to retrieve the Wikipedia pages that best describe the festival mentioned in a microblog (Section 3). Then, we scored these Wikipedia pages according to their relevance with respect to a microblog.

The second part of our system (see Fig. 1, right side) analyzes the best scored pages, then it extracts the relevant information from this subset in order to generate a short text summary. Our approach creates clusters of similar sentences, then we use a Cross-Language Compressive Text Summarization (CLCTS) system (Section 4) to compress the clusters and then generate summaries in four languages describing a festival.

3 Wikipedia Document Retrieval

The set of CLEF microblogs is composed of tweets in several languages related to festivals around the world. Wikipedia provides a description of a given festival in several languages (e.g. the Avignon Festival has a dedicated page in 17 languages). We independently analyze four language versions of Wikipedia (en, es, fr, and pt) for each microblog, by repeating the whole process first to retrieve the best Wikipedia pages and then to summarize the pages for the four versions of Wikipedia.

The following subsections describe the procedure to analyze and to retrieve the Wikipedia pages which are the most related to a festival in a microblog.

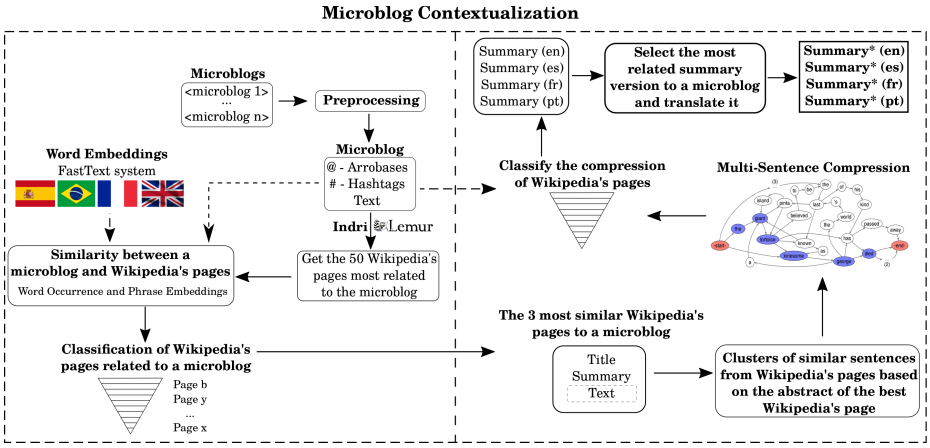


Fig. 1. Our system architecture to contextualize the microblogs.

3.1 Wikipedia Page Retrieval

The first challenge of the MC2 task is to retrieve the Wikipedia pages that best describe a festival in a microblog. A microblog is written in a specific language and contains usernames, hashtags, text, and punctuation marks. Based on this microblog, a system has to identify the most relevant Wikipedia pages in four languages with respect to a festival.

We assume that hashtags and usernames represent the keywords of a tweet, and they are independent of the language. In other words, the festival name, its geographic localization, or a show name normally have the same name in different languages (e.g. "Festival d'Avignon" in French and "Avignon Festival" in English share the same keywords). We remove all punctuation marks. From hashtags, usernames, and the plain text (i.e. the tweet without hashtags, usernames, and punctuation), we create Indri queries to retrieve 50 Wikipedia documents per each microblog[4]. These Indri queries have hashtags, usernames, and the word "festival" as keywords.

The procedure described above is simple but has several limitations. Some language versions of the Wikipedia database have very little information or no page at all about a festival. In this case, the Indri system may retrieve pages about other festivals (e.g. "Avignon Festival" is not available in Portuguese). Besides, some of these festivals have names that vary according to the language and our system does not translate these names to retrieve these pages in other languages. Another characteristic that we do not take into account is the date of a microblog. Normally, people write their microblogs during festivals, therefore timestamp could have helped us to identify the correct festival.

[4] `https://www.lemurproject.org/indri.php`

3.2 Selection of Wikipedia Pages

The Wikipedia pages retrieved by the Indri system may contain several subjects. Indri returns these pages sorted by relevance, where the first page is the most relevant, the second is less relevant and so on. However, the quality of these results depends on the Indri query and the amount of information available about a festival. Some microblogs only contain limited information about a festival, e.g. the location of a festival or the name of a show. In this case, a system has to identify the correct festival among several with similar characteristics, presentations in common, or in the same location.

To confirm the relevance of the Wikipedia pages retrieved by Indri, we select the pages most related to a microblog. Normally, the title of a Wikipedia document has few words and contains the core information, while the summary of the document, which is usually made of the first paragraphs of the article before the start of the first section, is larger and provide additional information[5]. Therefore, we consider Equation (4) to compute the relevance score of the Wikipedia document D with respect to the microblog T.

$$\text{score}_{title} = \alpha_1 \times \text{sim}(ht, title) + \alpha_2 \times \text{sim}(un, title) + \alpha_3 \times \text{sim}(nw, title) \quad (1)$$

$$\text{score}_{sum} = \beta_1 \times \text{sim}(ht, sum) + \beta_2 \times \text{sim}(un, sum) + \beta_3 \times \text{sim}(nw, sum) \quad (2)$$

$$\text{sim}(x, y) = \gamma_1 \times \text{cosine}(x, y) + \gamma_2 \times \text{occur}(x, y) \quad (3)$$

$$\text{score}_{doc} = \text{score}_{title} + \text{score}_{summary} \quad (4)$$

where ht are the hashtags of the tweet T, un the usernames of T, nw the normal words of T, and sum the summary of D. $\text{occur}(x, y)$ represents the number of occurrences of x in y, while $\text{cosine}(x, y)$ is the cosine similarity between x and y using Continuous Space Vectors[6] [2].

We empirically set up the parameters as follows: $\alpha_1 = \alpha_2 = 0.1, \alpha_3 = 0.01, \beta_1 = \beta_2 = 0.05, \beta_3 = 0.005, \gamma_1 = 1$ and $\gamma_2 = 0.5$. These coefficients give more weights to hashtags than usernames and the tweet text, and compensate the shorter length of the titles of Wikipedia articles with respect to their summary. These pages may contain several subjects and we only want to keep the pages that describe the festival of the microblog. Therefore, we finally keep in each language the three Wikipedia documents with the highest scores to be analyzed by the Text Compression (TC) system.

Our system prioritizes the information in hashtags and arrobases; however, a microblog has few information about a festival and, sometimes, this information is too general or too specific to easily identify a festival. Another problem is that

[5] We did not consider the whole text of Wikipedia pages because it is sometimes huge and we preferred to rely on the work of the contributors to build the summary of the article.

[6] We used the pre-trained word embeddings (en, es, fr, and pt) of FastText system [2] that is available in `https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md`.

the Wikipedia dataset has several kinds of pages, e.g. lists of festivals based on a show, cities, or types of festival. These pages contain irrelevant information about a particular festival and may reduce the informativeness of summaries.

4 Text Summarization

One of the biggest challenges of the Microblog Contextualization task is to summarize all the information available in a correct and informative summary about a festival. As we described before, the retrieved pages may contain wrong information because they may be in different languages and describe various festivals.

While famous festivals have several Wikipedia pages that describe in detail all previous editions, less prominent ones have only one page or no article at all in Wikipedia. For this reason, we use the best scored page as the reference for the contextualization of microblogs. This analysis helps to have access to the correct subject and avoid using information about other subjects. The abstract provided at the start of the Wikipedia pages is assumed to be good enough to be coherent and to provide a basic explanation about a festival. However, relying only on this part of the article may lead to miss relevant information about the festival that could be obtained from other sections or even other pages. For this reason, we preferred to use the summary of the top article as a basic abstract and to improve its quality with relevant information using Multi-Sentences Compression (MSC) (i.e. generate sentences that are shorter and more informative than the original sentences of the summary). Then, we translate the best summaries for the languages that have poor summaries.

In the case some Wikipedia pages do not have an abstract, the whole text is analyzed. Nevertheless, this text may have additional information that is not relevant to contextualize a festival in only 120 words. Therefore, our approach strongly depends on the best scored page abstract to generate a correct summary.

4.1 Clustering

Clustering enables the identification of subjects and relevant information inside a document. These clusters are composed of similar sentences. The objective of this process is to divide a document in topics where each cluster describes a specific topic.

As we consider the sentences of the summary of the best scored page as key sentences, we create clusters made of sentences from the three first retrieved pages, and similar to each key sentence. Two sentences are considered as similar if the cosine similarity between them is bigger than a threshold[7].

It can happen that some festivals have only a single relevant Wikipedia page. The cosine similarity normally helps in selecting only pertinent sentences; however, particularly in this case, sentences which are similar to key sentences may deal with different subjects and may still be included in clusters with irrelevant information.

[7] We empirically set up a threshold of 0.4 to consider two sentences as similar.

4.2 Multi-Sentence Compression

The problematics of text summarization is to produce summaries that are both grammatical and informative while meeting length restrictions, 120 words in the task considered here. Since most of sentences in Wikipedia are long, we attempt to compress them to preserve only the relevant information. We use a MSC method to generate a shorter and hopefully more informative compression for each cluster. Our MSC method adopts the approach proposed by Linhares Pontes et al. [8, 6] to model a document D as a Word Graph (WG), where vertices represent words and arcs represent the cohesion of the words. The weights of the arcs represent the level of cohesion between the words of two vertices based on the frequency and the position of these words in the sentences (Equation 5).

$$w(e_{i,j}) = \frac{\text{cohesion}(e_{i,j})}{freq(i) \times freq(j)}, \tag{5}$$

$$\text{cohesion}(e_{i,j}) = \frac{freq(i) + freq(j)}{\sum_{f \in D} dist(f,i,j)^{-1}}, \tag{6}$$

$$dist(f,i,j) = \begin{cases} pos(f,i) - pos(f,j), \text{ if } pos(f,i) < pos(f,j) \\ 0, \qquad\qquad\qquad \text{otherwise} \end{cases} \tag{7}$$

This approach relies on the analysis of keywords, in order to ensure to keep the core information of the cluster, and the 3-grams of the document, in order to preserve the grammaticality. Since each cluster to compress is composed of similar sentences, we consider that there is only one topic; the Latent Dirichlet Allocation (LDA) method is used to identify the keywords of this topic [1].

From the weights of 2-grams (Equation 5), the relevance of a 3-gram is based on the relevance of the two 2-grams, as described in Equation 8:

$$\text{3-gram}(i,j,k) = \frac{qt_3(i,j,k)}{\max_{a,b,c \in WG} qt_3(a,b,c)} \times \frac{w(e_{i,j}) + w(e_{j,k})}{2}, \tag{8}$$

In order to generate a better compression, the objective function expressed in Equation 9 is minimized in order to improve the informativeness and the grammaticality.

$$\text{Minimize } \Big(\alpha \sum_{(i,j) \in A} b_{i,j} \cdot x_{i,j} - \beta \sum_{k \in K} c_k \cdot w_k - \gamma \sum_{t \in T} d_k \cdot z_t\Big) \tag{9}$$

where x_{ij} indicates the existence of the arc (i,j) in the solution, $w(i,j)$ is the cohesion of the words i and j (Equation 5), z_t indicates the existence of the 3-gram t in the solution, d_t is the relevance of the 3-gram t (Equation 8), c_k indicates the existence of a word with label (keyword) k and β is the geometric average of the arc weights in the graph (more details in [8, 6]). Finally, the 50 best solutions are computed according to the objective (9) and we select the sentence with the lowest final score (Equation 10) as the best compression.

$$score_{norm}(f) = \frac{e^{score_{opt}(f)}}{||f||}, \tag{10}$$

where $score_{opt}(f)$ is the value of the path to generate the compression f from Equation 9. Like Linhares Pontes et al. [8], we set up the parameters to $\alpha = 1.0$, $\beta = 0.9$ and $\gamma = 0.1$.

Our approach assumes that clusters are composed of only correct sentences (subject+verb+object) to generate correct compressions. Another limitation is the similarity of sentences in a cluster. A cluster has to describe a single topic; otherwise, the MSC will merge information of several subjects and generate a compression with wrong information.

4.3 Summary Generation

The last step of summarization is the generation of summaries. While original sentences are likely to be more grammatically correct than compressions, the compressed sentences are by definition shorter and have in principle more relevant information. Therefore, we prefer to add a compression in the summary if this compression is considered more relevant than the original sentences.

We generate summaries by concatenating the most similar compression to a microblog without redundant sentences. The relevance of sentences/compressions is calculated based on the average TF-IDF. We add a sentence/compression to the summary only if the cosine similarity between this compression and the sentences already added in the summary is lower that a threshold of 0.4.

Let us note that our approach does not check the time of facts and consequently, it may generate summaries that do not preserve the sequence of facts.

4.4 Best Summary

The best possible scenario is the generation of a summary for each language version of Wikipedia. However, some language versions do not have a page or have a small text describing a specific festival. Therefore, we analyzed four summaries (one for each language version) for each microblog and we only retain the summary which contains the best description. We consider a summary as relevant if it is similar to the microblog. As the translation process generates some errors, we translate a language version summary only if the quality of the best summary is much better than other versions[8]. Therefore, we used the Yandex library[9] to translate the kept summary into other languages (en, es, fr, and pt).

[8] We translate a summary into a target language only if the summary in the target language has a similarity score (cosine similarity between the summary and the microblog) lower by 0.2 than the similarity score between the best summary and the microblog.

[9] `https://tech.yandex.com/translate/`

The pipeline made of the summarization and translation processes is prone to errors, which reduces the quality of summaries. However, we have to use information from other language versions of Wikipedia when the available information about a festival in a language is poor or does not exist.

5 Evaluation Protocol

The MC2 task contains several subtasks and the automatic evaluation of this task as an end-to-end problem generates incomplete results. In our opinion, the best way to evaluate this task is to split it in two subtasks (Wikipedia page retrieval and Text Summarization (TS)). In this case, we can estimate the impact of each subtask in the contextualization.

Our proposition for the evaluation protocol is composed of three steps: Wikipedia pages retrieval, TS and microblog contextualization (Figure 2). For the Wikipedia pages retrieval subtask, systems have to determine which Wikipedia pages describe a festival in a microblog. The TS subtask consists in generating a summary of a festival based on one or several Wikipedia pages. Finally, the microblog contextualization task is composed of both subtasks.

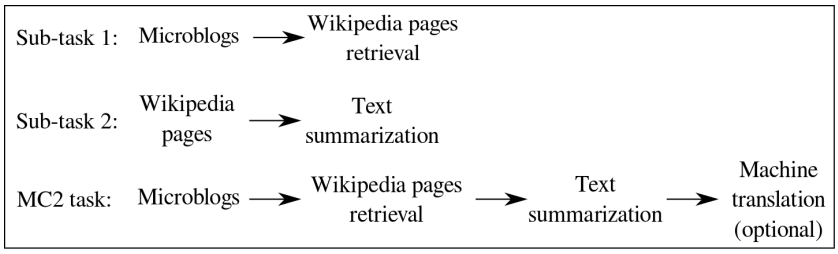


Fig. 2. Proposition of an evaluation protocol for MC2 task composed of two subtasks.

The Wikipedia pages retrieval subtask can be evaluated with a list of the Wikipedia pages related to a microblog. We can evaluate the performance of a system if it retrieves the correct Wikipedia pages for each microblog. The TS subtask and microblog contextualization task can be analyzed in several ways: automatic, semi-automatic and manual evaluations. Automatic (FRESA [9]) and semi-automatic (ROUGE [4]) evaluation systems analyze the overlap of n-grams between reference summaries and candidate summaries (or original text) to determine the quality of candidate summaries. However, compression and translation methods change the structure of sentences by generating paraphrases and new n-grams that may not exist in reference summaries (or source document), thereby reducing ROUGE (or FRESA) scores. In this case, a manual evaluation is required to evaluate the quality of these summaries.

6 Related Work and Propositions

Several studies have analyzed Text Summarization (TS) and Cross-Language Text Summarization (CLTS) [10, 11, 7]. TS aims to create a short, accurate, and fluid summary of a longer text document; CLTS also generates a summary but the language of the summary is different from the language of the source documents. As we described before, some language versions of Wikipedia have a limited content so the CLTS can produce more correct and informative summaries.

Wan [10] considered the information in the source and in the target language to estimate the relevance of sentences for cross-lingual summarization. He proposed a graph-based summarization method (CoRank) that considers a sentence as relevant if this sentence in both languages is heavily linked with other sentences in each language separately (source-source and target-target language similarities) and between languages (source-target language similarity). Zhang *et al.* [11] analyzed Predicate-Argument Structures (PAS) to obtain an abstractive English-to-Chinese CLTS. They split parallel sentences at the level of bilingual concepts and facts and use the CoRank method to fuse these structures and to generate cross-lingual summaries considering their saliency and their translation quality. Linhares Pontes *et al.* have published a recent work about cross-language text summarization of multiple texts written about the same topic [7]. Their method analyzes the information in both languages (source and target languages) to extract as much information as possible about documents. In addition, they use SC and MSC to compress and improve the informativeness of sentences and, consequently, the quality of the summary.

The methods described above need a group of documents that describe a same subject to generate a correct summary; however, the MC2 task does not necessarily provide correct documents about a festival and the use of these methods can generate bad summaries. A possible solution is to ensure the quality of the source documents about a same subject and to adapt these methods to analyze Wikipedia pages.

7 Conclusion

The Microblog Contextualization task is composed of several challenges that can modify the quality of results. Depending on the microblog, this task may require the generation of multi-lingual and cross-lingual summaries. We proposed a solution for each subtask in order to generate more informative summaries; however, this task involves several subtasks and the performance of our system depends on all these subtasks. This pipeline of subtasks complicates the identification of errors and the performance analysis of our approach. Another major problem is the lack of a training corpus to test and to adapt our system for this task.

We hope the organizers will make available a complete training/test dataset with all information about the main task and its subtasks in the next edition of Microblog Contextualization task. Our system is modular and can contextualize microblogs with several approaches. For example, we can remove the MSC

and/or the automatic translation methods in our approach. With this dataset, we could evaluate and improve our system.

Acknowledgement

This work was partially financed by the European Project CHISTERA-AMIS ANR-15-CHR2-0001.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. Journal Machine Learning Research 3, 993–1022 (Mar 2003), http://dl.acm.org/citation.cfm?id=944919.944937
2. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
3. Ermakova, L., Goeuriot, L., Mothe, J., Mulhem, P., Nie, J.Y., SanJuan, E.: Microblog cultural contextualization lab overview. In: CLEF 2017 Experimental IR Meets Multilinguality, Multimodality, and Interaction. Lecture Notes in Computer Science, vol. 10456, pp. 304–314. Springer (2017)
4. Lin, C.Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Workshop Text Summarization Branches Out (ACL). pp. 74–81 (2004)
5. Linhares Pontes, E., Huet, S., Torres-Moreno, J.M., Linhares, A.C.: Microblog contextualization using continuous space vectors: Multi-sentence compression of cultural documents. In: Working Notes of the CLEF Lab on Microblog Cultural Contextualization. vol. 1866. CEUR-WS.org (2017)
6. Linhares Pontes, E., Huet, S., Gouveia da Silva, T., Linhares, A.C., Torres-Moreno, J.M.: Multi-sentence compression with word vertex-labeled graphs and integer linear programming. In: TextGraphs-12: the Workshop on Graph-based Methods for Natural Language Processing. Association for Computational Linguistics (2018)
7. Linhares Pontes, E., Huet, S., Torres-Moreno, J.M., Linhares, A.C.: Cross-language text summarization using sentence and multi-sentence compression. In: 23rd International Conference on Natural Language & Information Systems (NLDB). Lecture Notes in Computer Science, vol. 10859, pp. 467–479. Springer (2018)
8. Linhares Pontes, E., Gouveia da Silva, T., Linhares, A.C., Torres-Moreno, J.M., Huet, S.: Métodos de otimização combinatória aplicados ao problema de compressão multifrases. In: Anais do XLVIII Simpósio Brasileiro de Pesquisa Operacional (SBPO). pp. 2278–2289 (2016)
9. Torres-Moreno, J.M.: Automatic Text Summarization. Wiley & Sons (2014)
10. Wan, X.: Using bilingual information for cross-language document summarization. In: ACL. pp. 1546–1555 (2011)
11. Zhang, J., Zhou, Y., Zong, C.: Abstractive cross-language summarization via translation model enhanced predicate argument structure fusing. IEEE/ACM Trans. Audio, Speech & Language Processing 24(10), 1842–1853 (2016)