IRIT at INEX 2012: Tweet Contextualization

Liana Ermakova, Josiane Mothe

Institut de Recherche en Informatique de Toulouse
118 Route de Narbonne, 31062 Toulouse Cedex 9, France
liana.ermakova.87@gmail.com, josiane.mothe@irit.fr

Abstract. In this paper, we describe an approach for tweet contextualization developed in the context of the INEX 2012. The task was to provide a context up to 500 words to a tweet from the Wikipedia. As a baseline system, we used TF-IDF cosine similarity measure enriched by smoothing from local context, named entity recognition and part-of-speech weighting presented at INEX 2011. We modified this method by adding bigram similarity, anaphora resolution, hashtag processing and sentence reordering. Sentence ordering task was modeled as a sequential ordering problem, where vertices corresponded to sentences and sequential constraints were represented by sentence time stamps.

Keywords: Information retrieval, tweet contextualization, summarization, sentence extraction, sequential ordering problem, hashtag, anaphora resolution.

1 Introduction

In 2012 at the tweet contextualization INEX task, systems should provide a context about the subject of the tweet. The context should be a readable summary up to 500 words composed of passages from the English Wikipedia corpus from November 2011 [1]. INEX organizers selected about 1000 non-personal tweets in English.

Twitter is "a microblogging service that enables users to post messages ("tweets") of up to 140 characters - supports a variety of communicative practices; participants use Twitter to converse with individuals, groups, and the public at large, so when conversations emerge, they are often experienced by broader audiences than just the interlocutors" [2]. Twitter's data flow is examined in order to measure public sentiment, follow political activity and news [3]. However, tweets may contain information that is not understandable to user without some context. User may be not familiar with mentioned named entities like persons, organizations or places. Searching for them on a mobile device is time consuming and expensive. Therefore providing concise coherent context seems to be helpful. Contextualization as a summary on a specific topic may be used at libraries, editorial boards, publishers, Universities and Schools, cellular providers. The last ones can include it in a package of services for their clients, e.g. to clarify information about news tweet on a mobile device without web searching. In the summary, a customer will find relevant context for names, people, places and events from the news tweet.

Though the idea to contextualize tweets is quite recent [4], there are several works on summarization [5] as well as on sentence retrieval [6]. Saggion and Lapalme (2002) provide the following definition of a summary:

A summary is "condensed version of a source document having a recognizable genre and a very specific purpose: to give the reader an exact and concise idea of the contents of the source" [7].

Summaries may be either "extracts" (the most important sentences extracted from the original text), or "abstracts" (if these sentences are paraphrased) [8]. Anyway abstract generation is based on extracting components [9] and that is why sentence retrieval module seems to be the most valuable with regard to summary informativeness.

This year we modified the extraction component developed for INEX 2011 [10] which showed the best results according to relevance evaluation [11]. However, there were several drawbacks in readability: unresolved anaphora and sentence ordering. Apparently, anaphora resolution should result on not only readability, but also informativeness of a text. Thus, we added anaphora resolution and we reordered the extracted sentences with regard to a graph model. So, the task was reduced to travelling salesman problem which was solved by greedy nearest neighbor algorithm. Sentences were modeled as graph vertex and the similarity measure between them corresponded to edges. Moreover, we improved our approach by using linear combination of bigram and unigram similarity measure instead of unigram cosine. Last year two sentences from a New York Times article were considered as a query. This year approximately 1000 real tweets were collected by the organizers [1]. The tweets contained hashtags and @replies. Hashtags seems to provide very important information and therefore we assigned to them additional weight.

The paper is organized as follows. Firstly, we describe the modifications we made relative to previous year. Then we discuss evaluation results. Future development description concludes the paper.

2 Method Description

2.1 Searching for Relevant Sentences

The baseline system is based on TF-IDF cosine similarity measure enriched by smoothing from local context, named entity recognition and part-of-speech weighting [10].

First changes we made concern bigrams. Bigrams provide more specific information than unigrams and they are frequent enough in comparison with trigrams. Bigrams treating does not imply any syntactic analysis. The number of shared bigrams are often used to evaluate summaries [11][12]. Therefore, for each query and each sentence we computed the linear combination of the unigram and bigram cosine. We assigned the weight 0.3 and 0.7 to unigram and bigram similarity measure respectively.

In order to resolve pronoun anaphora we added the mention from the previous context. A mention is added in a summary only if other mentions excluding pronouns do

not occur in the same sentence as the pronoun anaphora. Anaphora was also resolved at the stage of sentence extraction. Since all mentions of the same notion may be considered as contextual synonyms, we included them into vector representation of a sentence, i.e. we expanded the original sentence by the contextual synonyms of all concepts occurring within this sentence. Anaphora resolution was performed by Stanford CoreNLP[1].

One of the features frequently used in the Twitter is the hashtag symbol #, which "is used to mark keywords or topics in a Tweet. It was created organically by Twitter users as a way to categorize messages" and facilitate the search [13]. Hashtags are inserted before relevant keywords or phrases anywhere in tweets – at the beginning, middle, or end. Popular hashtags often represents trending topics. Bearing it in mind, we put higher weight to words occurring in hashtags. Usually key phrases are marked as a single hashtag. Thus, we split hashtags by capitalized letters.

Moreover, important information may be found in @replies, e.g. when a user reply to the post of a politician. "An @reply is any update posted by clicking the "Reply" button on a Tweet" [14]. Sometimes people use their names as Twitter usernames. Therefore, we split these usernames in the way we did it with hashtags.

2.2 Sentence reordering

As Barzilay et al. showed in 2002 sentence ordering is crucial for readability [15]. In single document summarization the sentence order may be the same as the initial relative order in the original text. However, this technique is not applicable to multi-document summarization. Therefore, we propose an approach to increase global coherence of text on the basis of its graph model, where vertices represents sentences and the same TF-IDF cosine similarity measure as in searching for relevant sentences. If two relevant sentences are neighbors in the original text, they are considered as a single vertex. The hypothesis is that neighboring sentences should be somehow similar to each other and the total distance between them should be minimal. Firstly, we computed the similarity between sentences and reduced sentence ordering task to travelling salesman problem.

The travelling salesman problem (TSP) is an NP-hard problem in combinatorial optimization. Given a list of cities and their pairwise distances, the task is to find the shortest possible route that visits each city exactly once and returns to the origin city. In the symmetric case, TSP may be formulated as searching for the minimal Hamiltonian cycle in an undirected graph. Asymmetric TSP implies a directed graph [16]. The obvious solution is to use brute force search, i.e. find the best solution among all possible permutations. The complexity of this approach is $O(n!)$ while other exact algorithms are exponential. Therefore, we chose the greedy nearest neighbor algorithm with minor changes.

Since sentence ordering does not request to return to the start vertex and the start vertex is arbitrary, we tried every vertex as the start one and chose the best result.

[1] http://nlp.stanford.edu/software/corenlp.shtml

However, this method does not consider chronological constraints. So, we modified the task and it gave us the sequential ordering problem (SOP).

SOP "is a version of the asymmetric traveling salesman problem (ATSP) where precedence constraints on the vertices must also be observed" [17]. SOP is stated as follows. Given a directed graph, find a Hamiltonian path of the minimal length from the start vertex to the terminal vertex observing precedence constraints.

Usually SOP is solved by the means of integer programming. Integer programming is NP-hard and these methods achieved only limited success [17]. Therefore, we solved the problem as follows. Firstly, we ordered sentences with time stamps $s_1 - s_2 - \cdots - s_n$. Sentences without time stamp were added to the set $P = \{p_j\}_{j=\overline{1,m}}$. For each pair $s_i - s_{i+1}$ we searched for the shortest path passing through vertices from P. These vertices were removed from P and $i = i + 1$. If $i = n$, we searched for the shortest path passing through all vertices in P and the edge with the maximal weight was removed.

The major disadvantage of this approach is that a text with the same repeated sentence would be falsely overscored. The naïve approach to avoid it is to use a threshold value. However, it cannot deal with sentences of almost the same sense but different length (e.g. with more adjectives). We adopted the idea of H. G. Silber and K. F. Mccoy that nouns provide the most valuable information [18] and that is why we propose to introduce coefficients to distinguish the impact of nouns, other significant words and stop-words. A sentence was mapped into a noun set. These sets were compared pairwise and if the normalized intersection was greater than a predefined threshold the sentences were rejected.

3 Evaluation

Summaries were evaluated according to their informativeness and readability [1].

Informativeness was estimated as the overlap of a summary with the pool of relevant passages (number of relevant passages, vocabulary overlap and the number of bigrams included or missing). For each tweet, all passages were merged and sorted in alphabetical order. Only 15 passages with the highest score from each run were added in the pool. Assessors had to provide a binary judgment on whether the passage is relevant to a tweet or not.

We submitted three runs. The first run A considered only the unigram cosine between a query and a sentence. The second run C took into account the linear combination of the unigram and bigram similarity measures but did not imply anaphora resolution. The third one B differed from C by resolved anaphora.

Informativeness results for the submitted runs are presented in Table 1. Column *Run* corresponds to the run id, *Unigrams, Bigrams and Skip bigrams* represents the proportion of shared unigrams, bigrams and bigrams with gaps of two tokens respectively. According to informativeness evaluation, the impact of the linear combination of the unigram and bigram similarity measures is smaller than the impact of anaphora resolution.

Table 1. Informativeness evaluation

Run	Unigrams	Bigrams	Skip bigrams	Average
B	0.8484	0.9294	0.9324	0.9034
C	0.8513	0.9305	0.9332	0.9050
A	0.8502	0.9316	0.9345	0.9054

Readability was measured as an average score of proportion of text that makes sense in context (relevance), proportion of text without syntactical errors (syntax) and proportion of text without unresolved anaphora and redundant information (structure). Readability evaluation also provides evidence that anaphora resolution has a stronger influence on average score than the use of bigram cosine. It increases dramatically the structure score.

Table 2. Readability evaluation

Run	Relevance	Syntax	Structure	Average
B	**0.4964**	**0.4705**	**0.4204**	**0.4624**
153	0.4984	0.4576	0.3784	0.4448
164	0.4759	0.4317	0.3772	0.4283
162	0.4582	0.4335	0.3726	0.4214
197	0.5487	0.4264	0.3477	0.4409
C	**0.449**	**0.4203**	**0.3441**	**0.4045**
A	**0.4911**	**0.3813**	**0.3134**	**0.3953**

4 Conclusion

In this article, we describe a method to tweet contextualization based on the local Wikipedia dump. As a baseline system, we used TF-IDF cosine similarity measure enriched by smoothing from local context, named entity recognition and part-of-speech weighting presented at INEX 2011. We modified this method by adding bi-gram similarity, anaphora resolution, hashtag processing and sentence reordering. Sentence ordering task was modeled as a sequential ordering problem, where vertices corresponded to sentences and sentence time stamps represented sequential constraints. We proposed the greedy algorithm to solve the sequential ordering problem based on chronological constraints. However, the organizers did not evaluate sentence order. In order to deal with redundant information we mapped each sentence into a noun set. These sets were compared pairwise and if the normalized intersection was greater than a predefined threshold, the sentences were rejected.

According to informativeness evaluation, the impact of the linear combination of the unigram and bigram similarity measures is smaller than the impact of anaphora resolution. Readability evaluation also provides evidence that anaphora resolution has a stronger influence on average score than the use of bigram cosine.

In future, we plan to work further with anaphora resolution and sentence ordering. It seems to be useful to find additional features special for the Twitter and to expand queries by synonyms and relations from WordNet. This should increase relevance as well as readability.

5 References

[1] "INEX 2012 Tweet Contextualization Track." [Online]. Available: https://inex.mmci.uni-saarland.de/tracks/qa/. [Accessed: 02-Aug-2012].

[2] D. Boyd, S. Golder, and G. Lotan, "Tweet, Tweet, Retweet: Conversational Aspects of Retweeting on Twitter," in *Proceedings of the 2010 43rd Hawaii International Conference on System Sciences*, 2010, pp. 1–10.

[3] N. Savage, "Twitter as medium and message," *Commun. ACM*, vol. 54, pp. 18–20, 2011.

[4] E. Meij, W. Weerkamp, and M. de Rijke, "Adding Semantics to Microblog Posts," *Pro-ceed-ings of the fifth ACM inter-na-tional con-fer-ence on Web search and data min-ing*, 2012.

[5] S. Gholamrezazadeh, M. A. Salehi, and B. Gholamzadeh, "A Comprehensive Survey on Text Summarization Systems," *Computer Science and its Applications*, pp. 1–6, 2009.

[6] V. G. Murdock, "Aspects of Sentence Retrieval," *Dissertation*, 2006.

[7] H. Saggion and G. Lapalme, "Generating Indicative-Informative Summaries with SumUM," *Association for Computational Linguistics*, vol. 28, no. 4, pp. 497–526, 2002.

[8] J. Vivaldi, I. da Cunha, and J. Ramırez, "The REG summarization system at QA@INEX track 2010," *INEX 2010. Workshop Preproceedings*, pp. 238–242, 2010.

[9] G. Erkan and D. R. Radev, "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization," *Journal Of Artificial Intelligence Research*, vol. 22, pp. 457–479, 2004.

[10] L. Ermakova and J. Mothe, "IRIT at INEX: Question Answering Task," *Focused Retrieval of Content and Structure, 10th International Workshop of the Initiative for the Evaluation of XML Retrieval (INEX 2011), Geva, S., Kamps, J., Schenkel, R. (Eds.). Lecture Notes in Computer Science, Springer*, pp. 219–227, 2012.

[11] E. SanJuan, V. Moriceau, X. Tannier, P. Bellot, and J. Mothe, "Overview of the INEX 2011 Question Answering Track (QA@INEX)," *Focused Retrieval of Content and Structure, 10th International Workshop of the Initiative for the*

Evaluation of XML Retrieval (INEX 2011), Geva, S., Kamps, J., Schenkel, R. (Eds.), 2012.

[12] C.-Y. Lin, “ROUGE: A Package for Automatic Evaluation of Summaries,” *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pp. 74–81, 2004.

[13] “Twitter Help Center | What Are Hashtags ("#" Symbols)?” [Online]. Available: https://support.twitter.com/articles/49309-what-are-hashtags-symbols. [Accessed: 02-Aug-2012].

[14] “Twitter Help Center | What are @Replies and Mentions?” [Online]. Available: https://support.twitter.com/groups/31-twitter-basics/topics/109-tweets-messages/articles/14023-what-are-replies-and-mentions. [Accessed: 02-Aug-2012].

[15] R. Barzilay, N. Elhadad, and K. R. McKeown, “Inferring Strategies for Sentence Ordering in Multidocument News Summarization,” *Journal of Artificial Intelligence Research*, no. 17, pp. 35–55, 2002.

[16] В. В. Морозенко, *Дискретная математика: учеб. пособие*. Пермь: ПГУ, 2008.

[17] I. T. Hernádvölgyi, “Solving the sequential ordering problem with automatically generated lower bounds,” *Proceedings of Operations Research 2003*, pp. 355–362, 2003.

[18] H. G. Silber and K. F. Mccoy, “Efficiently computed lexical chains as an intermediate representation for automatic text summarization,” *Computational Linguistics - Summarization*, vol. 28, no. 4, pp. 1–11, 2002.