Ultra-stemming and Statistical Summarization at INEX 2013 Tweet Contextualization Track

Juan-Manuel Torres-Moreno[1] and Patricia Velázquez-Morales[2]

[1] École Polytechnique de Montréal - Département de Génie Informatique
CP 6079 Succ. Centre Ville H3C 3A7 Montréal (Québec), Canada.
juan-manuel.torres@univ-avignon.fr
[2] VM Labs, 84000 Avignon, France.
patricia_velazquez@yahoo.com
http://www.polymtl.ca

Abstract. According to the organizers, the objective of the 2013 INEX Tweet Contextualization Task is: "*...The Tweet Contextualization aims at providing automatically information - a summary that explains the tweet. This requires combining multiple types of processing from information retrieval to multi-document summarization including entity linking.*" We present the Cortex summarizer applied to the INEX 2013 task. Cortex summarizer uses several sentence selection metrics and an optimal decision module to score sentences from a document source. The results show that Cortex system performs well on INEX task.

Keywords: INEX, Automatic Summarization System, Tweet contextualization, Cortex.

1 Introduction

Automatic text summarization is indispensable to cope with ever increasing volumes of valuable information. An abstract is by far the most concrete and most recognized kind of text condensation [1, 2]. We adopted a simpler method, usually called *extraction*, that allow to generate summaries by extraction of pertinence sentences [2–5]. Essentially, extracting aims at producing a shorter version of the text by selecting the most relevant sentences of the original text, which we juxtapose without any modification. The vector space model [6, 7] has been used in information extraction, information retrieval, question-answering, and it may also be used in text summarization [8]. CORTEX[3] is an automatic summarization system [9] which combines several statistical methods with an optimal decision algorithm, to choose the most relevant sentences.

An open domain Question-Answering system (QA) has to precisely answer a question expressed in natural language. QA systems are confronted with a fine and difficult task because they are expected to supply specific information and

[3] *CORTEX es Otro Resumidor de TEXtos* (CORTEX is anotheR TEXt summarizer).

not whole documents. At present there exists a strong demand for this kind of text processing systems on the Internet. A QA system comprises, *a priori*, the following stages [10]:

- Transform the questions into queries, then associate them to a set of documents;
- Filter and sort these documents to calculate various degrees of similarity;
- Identify the sentences which might contain the answers, then extract text fragments from them that constitute the answers. In this phase an analysis using Named Entities (NE) is essential to find the expected answers.

Most research efforts in summarization emphasize generic summarization [11–13]. User query terms are commonly used in information retrieval tasks. However, there are few papers in literature that propose to employ this approach in summarization systems [14–16]. In the systems described in [14], a learning approach is used (performed). A document set is used to train a classifier that estimates the probability that a given sentence is included in the extract. In [15], several features (document title, location of a sentence in the document, cluster of significant words and occurrence of terms present in the query) are applied to score the sentences. In [16] learning and feature approaches are combined in a two-step system: a training system and a generator system. Score features include short length sentence, sentence position in the document, sentence position in the paragraph, and tf.idf metrics. Our generic summarization system includes a set of eleven independent metrics combined by a Decision Algorithm. Query-based summaries can be generated by our system using a modification of the scoring method. In both cases, no training phase is necessary in our system.

This paper is organized as follows. In Section 2 we explain the INEX 2013 Tweet Contextualization Track. In Section 3 we explain the methodology of our work. Experimental settings and results obtained with Cortex summarizer are presented in Section 4. Section 6 exposes the conclusions of the paper and the future work.

2 INEX 2013 Tweet Contextualization Track

The Initiative for the Evaluation of XML Retrieval (INEX) is an established evaluation forum for XML information retrieval (IR) [17]. In 2013, tweet contextualization INEX task at CLEF 2013, aims "*given a new tweet, the system must provide some context about the subject of the tweet, in order to help the reader to understand it. This context should take the form of* **a readable summary, not exceeding 500 words, composed of passages from a provided Wikipedia corpus.**"[4]

Like in Question Answering track of INEX 2011 and 2012, the present task is about contextualizing tweets, i.e. answering questions of the form "What is

[4] `https://inex.mmci.uni-saarland.de/tracks/qa/`

this tweet about?" using a recent cleaned dump of the Wikipedia[5]. As organizers claim, the general process involves three steps:

- Tweet analysis.
- Passage and/or XML elements retrieval.
- Construction of the answer.

Then, a relevant passage segment contains:

- Relevant information but
- As few non-relevant information as possible (the result is specific to the question).

2.1 Document Collection

The corpus has been constructed from a dump of the English Wikipedia from November 2012. All notes and bibliographic references were removed to facilite the extraction of plain text answers. (Notes and bibliographic references are difficult to handle). Resulting documents contains a title, an abstract and section. Each section has a sub-title. Abstract end sections are made of paragraphs and each paragraph can have entities that refer to Wikipedia pages.

2.2 Tweets set

598 tweets in English were collected by the organizers from Twitter[6] for the Track 2013.. Tweets were selected and checked among informative accounts (for example, @CNN, @TennisTweets, @PeopleMag, @science...), in order to avoid purely personal tweets that could not be contextualized. Information such as the user name, tags or URLs will be provided.

3 Cortex Summarization System

Cortex [18, 19] is a single-document extract summarization system. It uses an optimal decision algorithm that combines several metrics. These metrics result from processing statistical and informational algorithms on the document vector space representation.

The INEX 2013 Tweet Contextualization Track evaluation is a real-world complex question (called long query) answering, in which the answer is a summary constructed from a set of relevant documents. The documents are parsed to create a corpus composed of the query and the the multi-document retrieved

[5] See the official INEX 2013 Tweet Contextualization Track Website: `https://inex.mmci.uni-saarland.de/tracks/qa/`.

[6] `www.tweeter.com`

by a Perl program supplied by INEX organizers[7]. This program is coupled to Indri system[8] to obtain for each query, 50 documents from the whole corpus.

The idea is to represent the text in an appropriate vectorial space and apply numeric treatments to it. In order to reduce complexity, a preprocessing is performed to the question and the document: words are filtered, lemmatized and stemmed. The Cortex system uses 11 metrics (see [20, 19] for a detailed description of these metrics) to evaluate the sentence's relevance.

By example, the topic-sentence overlap measure assigns a higher ranking for the sentences containing question words and makes selected sentences more relevant. The overlap is defined as the normalized cardinality of the intersection between the query word set T and the sentence word set S.

$$\text{Overlap}(T, S) = \frac{card(S \cap T)}{card(T)} \tag{1}$$

The system scores each sentence with a decision algorithm that relies on the normalized metrics. Before combining the votes of the metrics, these are partitioned into two sets: one set contains every metric $\lambda^i > 0.5$, while the other set contains every metric $\lambda^i < 0.5$ (values equal to 0.5 are ignored). We then calculate two values α and β, which give the sum of distances (positive for α and negative for β) to the threshold 0.5 (the number of metrics is Γ, which is 11 in our experiment):

$$\alpha = \sum_{i=1}^{\Gamma} (\lambda^i - 0.5); \quad \lambda^i > 0.5 \tag{2}$$

$$\beta = \sum_{i=1}^{\Gamma} (0.5 - \lambda^i); \quad \lambda^i < 0.5 \tag{3}$$

The value given to each sentence s given a query q is calculated with:

$$\begin{aligned} &\text{if}(\alpha > \beta) \\ &\qquad \text{then} \;\; \text{Score}(s, q) = 0.5 + \tfrac{\alpha}{\Gamma} \\ &\qquad \text{else} \;\; \text{Score}(s, q) = 0.5 - \tfrac{\beta}{\Gamma} \end{aligned} \tag{4}$$

The Cortex system is applied to each document of a topic and the summary is generated by concatenating higher score sentences.

[7] See: `http://qa.termwatch.es/data/getINEX2011corpus.pl.gz`

[8] Indri is a search engine from the Lemur project, a cooperative work between the University of Massachusetts and Carnegie Mellon University in order to build language modelling information retrieval tools. See: `http://www.lemurproject.org/indri/`

3.1 Ultrastemming + Cortex

In Automatic Text Summarization, preprocessing is an important phase to reduce the space of textual representation. Classically, stemming and lemmatization have been widely used for normalizing words. However, even using normalization on large texts, the curse of dimensionality can disturb the performance of summarizers. The main idea of ultra stemming[21] is to avoid analyzers, lemmatizers, stemmers and stoplists.

4 Experiments Settings and Results

In this study, we used the document sets made available during the Initiative for the Evaluation of XML retrieval (INEX)[9], in particular on the INEX 2012 Tweet Contextualization Track.

The strategy of Cortex system to deal multi-document summary problem is quite simple: first, a long single document D is formed by concatenation of all $i = 1, ..., n$ relevant documents provided by Indri engine: $d_1, d_2, ...d_n$. The first line of this multi-document D is the tweet T. Cortex summarizer extracts of D the most relevant sentences following T. Then, this subset of sentences is sorted by the date of documents d_i. The summarizer adds sentences into the summary until the word limit is reached. To evaluate the performance of Cortex system on INEX tweet contextualization track, we used the online package available from INEX website[10].

5 Results

Table 1 shows the official results of informativeness on INEX 2013 contextualization task. The performances (rank) of our summarizers are: Cortex lemmatization (Run 259)=15/24, Cortex stemming (Run 260)=16/24 and Cortex 4-ultra stemming (Run 261)=14/24.

Table 1. Informativeness results for Cortex system (runs 259-261)

Rank	Participant	Run	Manual	All.skip
1	199	256	y	0.8861
...				
14	129	261	n	**0.9670**
15	129	259	n	**0.9679**
16	129	260	n	**0.9680**
...				
24	180	269	y	0.9999

[9] `https://inex.mmci.uni-saarland.de/`
[10] `http://qa.termwatch.es/data/`

Table 2 shows the official results of Readability on INEX 2013 contextualization task. The performances (rank) of our summarizers are: Cortex lemmatization (Run 259)=15/22, Cortex stemming (Run 260)=13/22 and Cortex 4-ultra stemming (Run 261)=17/22.

Table 2. Readability results for Cortex system (runs 259-261)

Rank	Run	Mean average (%)
1	275	72.44
...		
13	260	**38.21**
15	259	**38.78**
17	261	**36.42**
...		
22	269	00.04

6 Conclusions

In this paper we have presented the Cortex summarization system applied on INEX 2013 Tweet Contextualization Track. The first one is based on the fusion process of several different sentence selection metrics. The decision algorithm obtains good scores on the INEX 2013 Tweet Contextualization Track (the decision process is a good strategy without training corpus). The second one is based on the divergence between summary and the source document.

Cortex summarizer using 4-ultra stemming as normalization has obtained very good results in informativeness evaluations. However, Cortex using stemming normalization outperforms Cortex using lemmatization and 4-ultra stemming. A module of compound words may improves the performance of our summarizer. We show that a simple statistical summarizer show good performances in this complex task.

References

1. ANSI. American National Standard for Writing Abstracts. Technical report, American National Standards Institute, Inc., New York, NY, 1979. (ANSI Z39.14.1979).
2. J.M. Torres-Moreno. *Resume automatique de documents : une approche statistique.* Hermes-Lavoisier, 2011.
3. H. P. Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2(2):159, 1958.
4. H. P. Edmundson. New Methods in Automatic Extracting. *Journal of the ACM (JACM)*, 16(2):264–285, 1969.
5. I. Mani and M. Mayburi. *Advances in automatic text summarization.* The MIT Press, U.S.A., 1999.

6. Gregory Salton. *The SMART Retrieval System - Experiments un Automatic Document Processing*. Englewood Cliffs, 1971.
7. Gregory Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.
8. I. Da Cunha, S. Fernandez, P. Velázquez Morales, J. Vivaldi, E. SanJuan, and J.M. Torres-Moreno. A new hybrid summarizer based on vector space model, statistical physics and linguistics. In *MICAI 2007: Advances in Artificial Intelligence*, pages 872–882. Springer Berlin/Heidelberg, 2007.
9. J.M. Torres-Moreno, P. Velazquez-Morales, and JG. Meunier. Condensés automatiques de textes. *Lexicometrica. L'analyse de données textuelles : De l'enquête aux corpus littéraires*, Special(www.cavi.univ-paris3.fr/lexicometrica), 2004.
10. C. Jacquemin and P. Zweigenbaum. Traitement automatique des langues pour l'accès au contenu des documents. *Le document en sciences du traitement de l'information*, 4:71–109, 2000.
11. Jose Abracos and Gabriel Pereira Lopes. Statistical Methods for Retrieving Most Significant Paragraphs in Newspaper Articles. In Inderjeet Mani and Mark T. Maybury, editors, *ACL/EACL97-WS*, Madrid, Spain, July 11 1997.
12. Simone Teufel and Marc Moens. Sentence Extraction as a Classification Task. In Inderjeet Mani and Mark T. Maybury, editors, *ACL/EACL97-WS*, Madrid, Spain, 1997.
13. Eduard Hovy and Chin Yew Lin. Automated Text Summarization in SUMMARIST. In Inderjeet Mani and Mark T. Maybury, editors, *Advances in Automatic Text Summarization*, pages 81–94. The MIT Press, 1999.
14. Julian Kupiec, Jan O. Pedersen, and Francine Chen. A Trainable Document Summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 68–73, 1995.
15. Anastasios Tombros, Mark Sanderson, and Phil Gray. Advantages of Query Biased Summaries in Information Retrieval. In Eduard Hovy and Dragomir R. Radev, editors, *AAAI98-S*, pages 34–43, Stanford, California, USA, March 23–25 1998. The AAAI Press.
16. Judith D. Schlesinger, Deborah J. Backer, and Robert L. Donway. Using Document Features and Statistical Modeling to Improve Query-Based Summarization. In *DUC'01*, New Orleans, LA, 2001.
17. Shlomo Geva, Jaap Kamps, Ralf Schenkel, and Andrew Trotman, editors. *Comparative Evaluation of Focused Retrieval - 9th International Workshop of the Inititative for the Evaluation of XML Retrieval, INEX 2010, Vugh, The Netherlands, December 13-15, 2010, Revised Selected Papers*, volume 6932 of *Lecture Notes in Computer Science*. Springer, 2011.
18. J.M. Torres-Moreno, P. Velazquez-Moralez, and J. Meunier. *CORTEX, un algorithme pour la condensation automatique de textes*. In *ARCo*, volume 2, page 365, 2005.
19. J.M. Torres-Moreno, P.L. St-Onge, M. Gagnon, M. El-Bèze, and P. Bellot. Automatic summarization system coupled with a question-answering system (qaas). *in CoRR*, abs/0905.2990, 2009.
20. J.M. Torres-Moreno, P. Velazquez-Morales, and J.G. Meunier. *Condensés de textes par des méthodes numériques*. *JADT*, 2:723–734, 2002.
21. J.M. Torres-Moreno. Beyond stemming and lemmatization: Ultra-stemming to improve automatic text summarization. *CoRR*, abs/1209.3126, 2012.