# An automatic greedy summarization system at INEX 2013 Tweet Contextualization Track

Andréa Carneiro Linhares[1]

UFC - Universidade Federal do Ceará
Rua Estanislau Frota, S/N, Centro, Sobral-CE, Brasil
andrea.linhares@ufc.br

**Abstract.** According to the organizers, the aim of the 2013 INEX Tweet Contextualization Track is: "*...given a 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 (and short) summary, composed of passages from [...] Wikipedia.*" We present an automatic greedy summarizer named REG applied to the INEX 2013 task. REG summarizer uses a greedy optimization algorithm to weigh the sentences. The summary is obtained by concatenating the relevant sentences, weighed in the optimization step. The results show that the REG system (using original tweets with a manual processing) do not perform very well on INEX 2013 contextualization track.

**Keywords:** Automatic greedy summarization system, REG, Tweet contextualization.

## 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, 3]. 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 [4] has been used in information extraction, information retrieval, question-answering, and it may also be used in text summarization [5]. REG[1] is an automatic greedy summarization system [6] which uses graph methods to spot the most important sentences in the document.

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 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:

[1] *REsumeur Glouton* (Greedy summarizer).

1. Transform the questions into queries, then associate them to a set of documents;
2. Filter and sort these documents to calculate various degrees of similarity;
3. 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 [7]. 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 [8, 9]. In the systems described in [8], a learning approach is used. A document set is used to train a classifier that estimates the probability that a given sentence is included in the extract. In [9], 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 [10] 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. The REG system begins with the proper representation of the documents using a vector space model, then weigh the sentences by a greedy optimization algorithm[11]. The process to produces a summary is performed by concatenating the relevant sentences, weighed in the optimization step.

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 REG are presented in Section 4. Section 5 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) [12]. 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.**"[2]

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 this tweet about?" using a recent cleaned dump of the Wikipedia[3]. As organizers claim, the general process involves three steps:

1. Tweet analysis.

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

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

2. Passage and/or XML elements retrieval.
3. 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 Tweets set

598 tweets in English were collected by the organizers from Twitter[4] 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 REG summarization system

The REG system includes three modules. The first one is responsible for the text vectorial processing (Cortex system [13])[5]) with processes of filtering, stemming and standardization. The second applies to the greedy algorithm and performs the calculation of the adjacency matrix. We get the phrase weighing $\nu$ of the algorithm directly. Thus, the relevant sentences will be selected as having the greatest weigh. The third module generates summaries and displays concatenation of relevant sentences.

### 3.1 Preprocessing and vector space representation

Documents are pre-treated with conventional filtering algorithms of functional words, normalization and stemming [14] to reduce the dimensionality. A bag of words representation produces a matrix $S_{[P\times N]}$ of frequencies / absences made of $\mu = 1, \cdots, P$ phrases (lines); $\boldsymbol{\sigma}_\mu = \{s_{\mu,1}, \cdots, s_{\mu,i}, \cdots, s_{\mu,N}\}$ and a vocabulary of $i = 1, N$ terms (columns).

The presence of the word $i$ is represented by its frequency $TF_i$ (his absence by 0, respectively), and a sentence $\boldsymbol{\sigma}_\mu$ is an array of $N$ occurrences.

### 3.2 Greedy solution

A graph $G = (V, E)$ is created from the vector representation of documents, where $S$ vertices express sentences and $A$ is the set of edges. An edge between two nodes is established if the corresponding phrases have at least one word in common.An adjacency matrix $A_{[P\times P]}$ is constructed from the matrix $S_{[sentences \times words]}$. The calculation is as follows: scan the line $i$, and for each element $a_{i,j}$ equal to 1, down by $j$ column to identify other phrases that share the word.

The proposed algorithm works as follows:

[4] `www.tweeter.com`

[5] The system Cortex performs unsupervised relevant sentences using several metrics controlled by a decision algorithm extraction.

1. generate $A$, which has rows and columns of sentences considered;
2. calculate the weight of vertices (the sum of incoming edges of the vertex);
3. calculate the degree of each vertex (the number of sentences partitioned with other words);

The solution adopted is based on a calculation of greedy search paths.

## 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)[6], in particular on the INEX 2013 Tweet Contextualization Track. We have performed a simplification of tweets provided by using a simple manual processing and this new list $T$ of tweets was normalized before continue our experimentation protocol.

The strategy of REG 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 $T$. The REG summarizer system extracts of $D$ the most relevant sentences following $T$. Then, this subset of sentences is sorted by the date of documents $d_i$. The summarizers add sentences into the summary until the word limit is reached.

### 4.1 INEX tweets simplification

The strategy employed to generate 598 queries from tweets was very simple. The tweets not carrying information words were removed. Then, the summarizer used the query as a title of a big multi-document retrieved by Indri engine.

We show an example of our manual processing. Let's consider the tweet number 303260378618531840 that the summary should contextualize:

```
<topic id="16923118118174720 0">
<tweet>
Ibra booked for having a barney with the referee
The only thing he has done noticeable tonight Into stoppage time now
</tweet>
```

Then, query 169231181181747200 is simplified as show: q = "`Ibra booked for having a barney with the referee The only thing he has done noticeable tonight`"

### 4.2 Results

The REG system used three methods to normalize the words: lemmatization, stemming and 4-ultra stemming [15]. Each method corresponds to a different run, identified by 263, 264 and 265, respectively.

[6] `https://inex.mmci.uni-saarland.de/`

**Table 1.** Informativity results for REG system (runs 263-265)

| **Rank** | Participant | Run | Manual | All.skip |
|---|---|---|---|---|
| 1 | 199 | 256 | y | 0.8861 |
| 2 | 199 | 258 | n | 0.8943 |
| 19 | 138 | 265 | n | 0.9789 |
| 20 | 138 | 263 | n | 0.9793 |
| 21 | 138 | 264 | n | 0.9798 |
| 23 | 180 | 269 | n | 0.9999 |
| 24 | 180 | 269 | y | 0.9999 |

INEX had provided two evaluations: the informativity and readability of the candidates summaries (500 words). The tables 1 and 2 shows the official results of some participants of INEX 2013 contextualization task: runs 263 to 265, the two first places and the two last ones. Table 1 uses the **all** overlapping reference values to compare the performance of the different runs. In particular, the run 265 performs better than 263 and 264 and 4-ultra stemming outperforms stemming. Unfortunately, the divergence results provided by REG are not very good.

The same process to compare the readability results presented on table 2. In this case, stemming outperforms 4-ultra stemming.

**Table 2.** Readability results for REG system (runs 263-265)

| **Rank** | Run | Mean average (%) |
|---|---|---|
| 1 | 275 | 72.44 |
| 2 | 256 | 72.13 |
| 12 | 264 | 38.56 |
| 14 | 265 | 37.92 |
| 18 | 263 | 32.75 |
| 21 | 277 | 20.00 |
| 22 | 269 | 00.04 |

## 5 Conclusions

In this paper we have presented the REG (*REsumeur Glouton*) summarization system applied on INEX 2013 Tweet Contextualization Track. REG is an automatic greedy summarization system which uses graph methods to spot the most important sentences in the document.

REG summarizer used a normalized list issued from a manual processing on the original tweets as inputs. It did not provide good results in the informativity evaluation, but in the readability context, it could be more competitive with

some improvements on the queries sent to Indri engine. The manual process strategy was quite simple.

## 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. Gregory Salton. *The SMART Retrieval System - Experiments un Automatic Document Processing*. Englewood Cliffs, 1971.
5. 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.
6. A. LINHARES, J.-M. Torres-Moreno, and J. Ramirez. Résumé automatique 4-lingue avec un algorithme glouton. In *14° Congrès ROADEF 2013*, Université de Technologie de Troyes, France, 13-15 février, 2013. ROADEF.
7. 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.
8. 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.
9. 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.
10. 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.
11. Rada Mihalcea. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, ACLdemo '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
12. Shlomo Geva, Jaap Kamps, Ralf Schenkel, and Andrew Trotman, editors. *Comparative Evaluation of Focused Retrieval - 9th International Workshop of the Initiative 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.
13. 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.
14. I. Mani and M. T. Maybury. *Advances in Automatic Text Summarization*. The MIT Press, 1999.
15. J.M. Torres-Moreno. Beyond stemming and lemmatization: Ultra-stemming to improve automatic text summarization. *CoRR*, abs/1209.3126, 2012.