July 4-6, 2018
Athens, Greece
2018
41st International Conference on
Telecommunications and Signal Processing
TSP

Automatic text summarization by mean-absolute constrained convex optimization

Claudiu POPESCU, **Lăcrimioara GRAMA** , Corneliu RUSU

Technical University of Cluj-Napoca
Faculty of Electronics, Telecommunications and Information Technology
Basis of Electronics Department
Signal Processing Group

The work of the second was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI-UEFISCDI, project number PNIII-P2-2.1-PED-2016-1608, 222PED/2017, within PNCDI III.
The work of the third was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI-UEFISCDI, project number PNIII-P2- 2.1-BG-2016-0378, 54BG/2016, within PNCDI III.

Outline

Research Aim

Proposed Algorithm

Implementation

Dataset

Experimental Results

Conclusion

Research Aim

The aim of this research was to propose a new algorithm for extractive text summarization

- **Text summarization** = the process of generating a short text, starting from a larger text document/ group of documents, with the property that it contains as much relevant information from the original text/ texts as possible

When a human writes a summary for some text, he/ she first understands the message encoded in the sentences and then, usually using other words, produces a shorter text with the same meaning

- **Abstractive summarization**
- It is considered, in general, an unsolvable problem with the current technology

Research Aim

With additional restrictions (for instance the type of data included in the text is known *a priori*), the task becomes more manageable and some solutions are known

Some new techniques, in special deep learning, are currently under consideration in this field

Most effort has been made in the more simple challenge of generating a summary by extracting parts from the original text/ texts

- **Extractive summarization**
- Our paper is a continuation of this effort

Research Aim

In general, a system for text summarization is build from

- A method of representing the information contained in the text units,
- Some relevance criteria,
- And a procedure for extraction based on this criteria

The concrete instantiations of these elements vary widely, generating a large number of algorithms, each one with strong and weak points

An interesting line of research is that of formulating the summarization as a submodular function maximization problem

- This approach proved to be quite useful for both practical and theoretical reasons and it is similar to ours, in the way that it attacks the summarization from an optimization perspective

Research Aim

Our work tries to solve the standard extractive summarization problem using an approach based on formulating a convex program with constraints

The proposed method is based on convex minimization and the properties of the L_1 norm

- Properties that have found a lot of applications in fields like signal processing and statistics/ machine learning

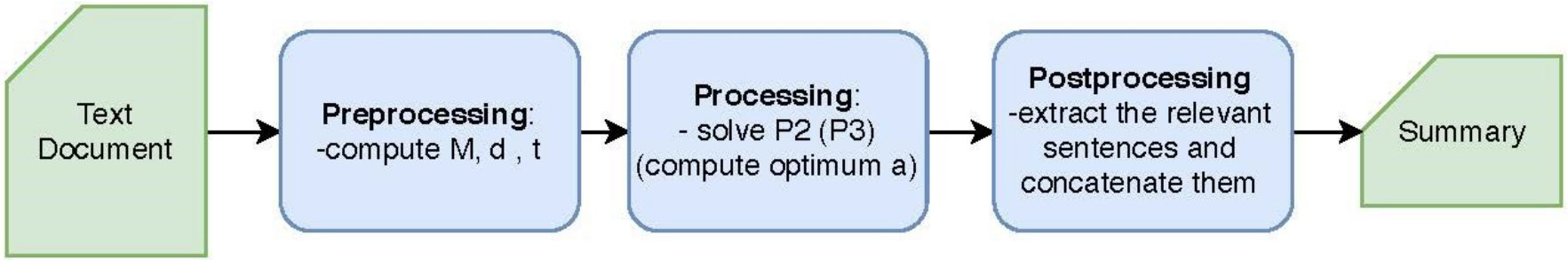
For comparison purpose we have also implemented a widely used summarization algorithm called TextRank

- This is a variation of the PageRank algorithm used in web search engines

Proposed Algorithm

Follows the general pattern

1. Preprocessing
2. Processing
3. Postprocessing



Algorithm – Preprocessing

This part is limited to only some basic operations:

- Tokenization;
- Conversion to lowercase;
- Stemming/lemmatization (with the Porter stemmer) – reduce the words to their base form;
- Stop words removal (eg., “and”, “for”, etc.)

We are also concerned with the numerical representation of the text (Term Frequency Inverse Document Frequency)

- Text $\rightarrow$ as a real-valued matrix $\mathbf{M} \in \mathbf{R}^{nxm}$
  - n – number of sentences, m – total number of distinct words (each row associated to a sentence, each column associated to a word)
  - Raw’s element m_{ij} – relevance of word j to the sentence i

Algorithm – Preprocessing

- Overall text can be represented as a vector **d** with each element d_j being the TF-IDF value for word *j*

We select a maximum number of *k* sentences that best approximate the text in this representation

- Some additional properties of the sentences
  - Like the length or the position in the text have an influence on the decision to introduce them into the summary
- We encode this additional information into a vector with positive real values **b**

Algorithm – Preprocessing

Within this framework we can formalize our intuitions quite easy in the following 0-1 integer program

- Vector **a**: binary vector
  - Tells what sentences we are keeping in the summary
- λ (positive real value) and vector **b**
  - Are user provided parameters
  - Encode the influence of the ”side information”, namely the *a priori* knowledge about the importance of different sentences
  - Set by trial and error method using the previous experience
- E.g.: the sentences from the first paragraph of a news article or those in the conclusion section of a scientific paper are usually more relevant than the others

$$\begin{cases} \min_{\mathbf{a}\in\mathbb{R}^n} \{ \| \mathbf{M}^T\mathbf{a} - \mathbf{d} \|_2^2 - \lambda \mathbf{b}^T\mathbf{a} \} \\ a_i \in \{0,1\}, \forall i = 1, n \\ \| \mathbf{a} \|_0 \leq k. \end{cases}$$

Algorithm – Preprocessing

$$\begin{cases} \min_{\mathbf{a}\in\mathbb{R}^n}\{\| \mathbf{M}^T\mathbf{a}-\mathbf{d} \|_2^2 - \lambda \mathbf{b}^T\mathbf{a}\} \\ a_i \in \{0,1\}, \forall i = 1,n \\ \| \mathbf{a} \|_0 \leq k. \end{cases}$$

- Matrix **M** and vector **d** are generated from data

The condition for the vector **a** to have only 0s and 1s may appear a bit unnatural

- This hard constraint will be relaxed anyway in the final program
- Nevertheless, it can become more acceptable if we use some form of normalization such that the elements in the matrix **M** will be less than those in the vector **d**

Unfortunately no efficient algorithm for this program is known to exists

Algorithm – Processing

An idea from compressive sampling came to our rescue

- L_0 pseudonorm (represents the number of non-zero elements of a vector; is not a true norm because it does not satisfy the triangle inequality) can be "approximated" by L_1
- Convex program
  - Can be efficiently solved

$$\begin{cases} \min_{\mathbf{a}\in\mathbb{R}^n}\{\| \mathbf{M}^T\mathbf{a}-\mathbf{d}\|_2^2 - \lambda\mathbf{b}^T\mathbf{a}\} \\ 0 \le a_i \le 1, \forall i = 1, n \\ \| \mathbf{a} \|_1 \le k. \end{cases}$$

- Basis pursuit can be seen as a mechanism for sparse signal reconstruction from incomplete measurements
- LASSO can be seen as an automatic sparse feature selection mechanism

Our algorithm can be interpreted as a sparse relevant parts extraction mechanism

Algorithm – Processing

$$\begin{cases} \min_{\mathbf{a}\in\mathbb{R}^n} \{\| \mathbf{M}^T\mathbf{a} - \mathbf{d} \|_2^2 - \lambda \mathbf{b}^T\mathbf{a}\} \\ 0 \leq a_i \leq 1, \forall i = 1, n \\ \| \mathbf{a} \|_1 \leq k. \end{cases}$$

The summary is finally generated (in the postprocessing step) by concatenating, in the original text order, the sentences associated with the greatest *k* non-zero elements in the vector **a**

Algorithm – Processing

An improved version

The title/headline can offer useful information

- This can be integrated in the objective function
  - By representing the title as a usual sentence with a (TF-IDF) vector **t** and
  - By trying to reflect its content in the selected sentences

$$\begin{cases} \min_{\mathbf{a}\in\mathbb{R}^n}\{\| \mathbf{M}^T\mathbf{a}-\mathbf{d}\|_2^2 + \lambda' \| \mathbf{M}^T\mathbf{a}-t\|_2^2 - \lambda \mathbf{b}^T\mathbf{a}\} \\ 0 \le a_i \le 1, \forall i = 1,n \\ \| \mathbf{a} \|_1 \le k. \end{cases}$$

  - λ' is also a tradeoff parameter

Algorithm – Postprocessing

The relevant sentences extracted in the previous step are concatenated

Additional steps that can be implemented

- Eliminate the grammatically wrong sentences
- Eliminate the ambiguities (“I”, “we”, etc.)

Implementation

The proposed solution was implemented in Python

- It offers good features for both language and numerical computation
- For preprocessing: the popular natural language processing library NLTK
- For the numerical computation part: Numpy and Scipy libraries
- The convex program was solved using a general purpose solver

Dataset

About 4500 press articles, with their human generated summaries

- For each article
  - The headline is available
  - The summaries tend to have an approximately constant number of sentences
- Regarding the difficulty of the task, it can be placed between
  - That of summarizing a scientific article, which has a lot of structure and key words
  - And that of summarizing a literary text with free and unexpected structure and usually with lots of common words with very context specific meaning

K. Vonteru, “News summary. Generating short length descriptions of news articles.” Available: https://www.kaggle.com/sunnysai12345/news-summary/data

Experimental Results

The evaluation was based on the ROUGE tool

- We selected ROUGE-1 (with unigrams) and we kept the default settings, except that we allowed stemming and stop words removal
- Basically ROUGE-1 evaluates the overlap between the generated and the human produced reference summary at words level (syntax and words order are ignored)

The computed metrics (at word level – the syntax and context are ignored) are

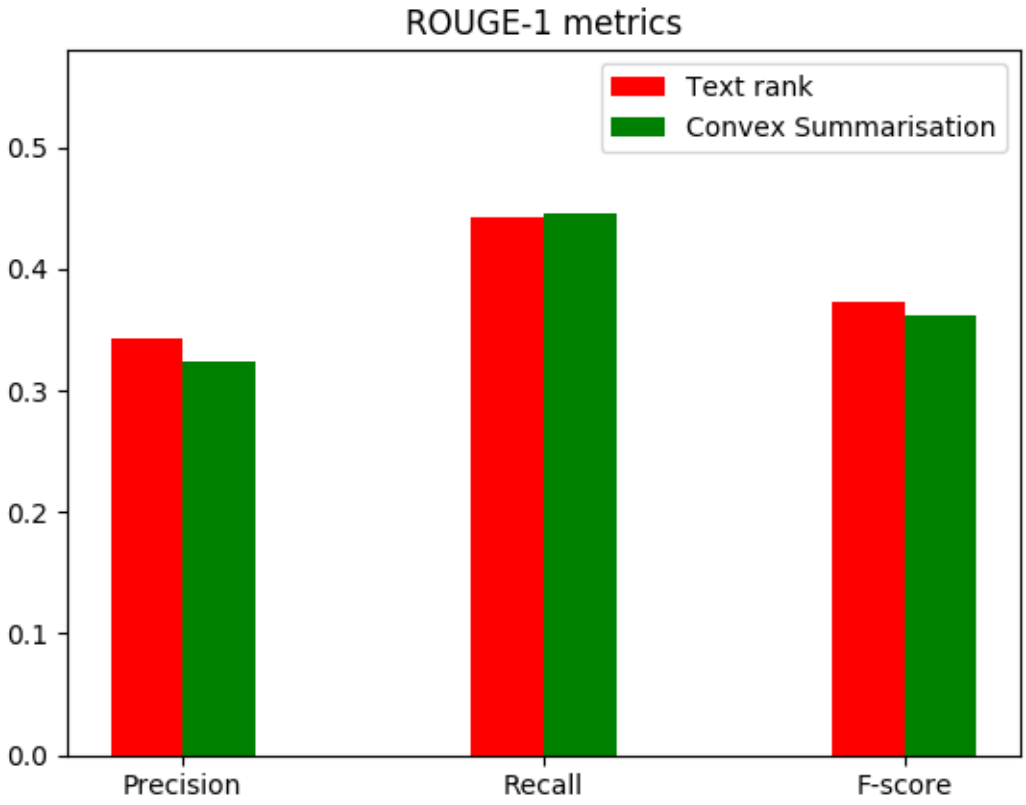
- **Precision** – ”what percent of the words in generated summary are also in the reference summary”
- **Recall** – ”what percent of the words in the reference summary are present in the generated summary”
- **F-score** – the harmonic mean of precision and recall

Experimental Results

Basic algorithm compared by TextRank

TextRank, in its standard form, can not take advantage of the side information

- We also ignore it by setting $\lambda = 0$



Experimental Results

Basic algorithm compared by TextRank

We do not need a very high precision solution of the optimization program, we want

- Just the right order
- And to distinguish between zero and nonzero values

For this reason we set a relatively big tolerance of 0.1

- We only lose less than 0.1% for all metrics

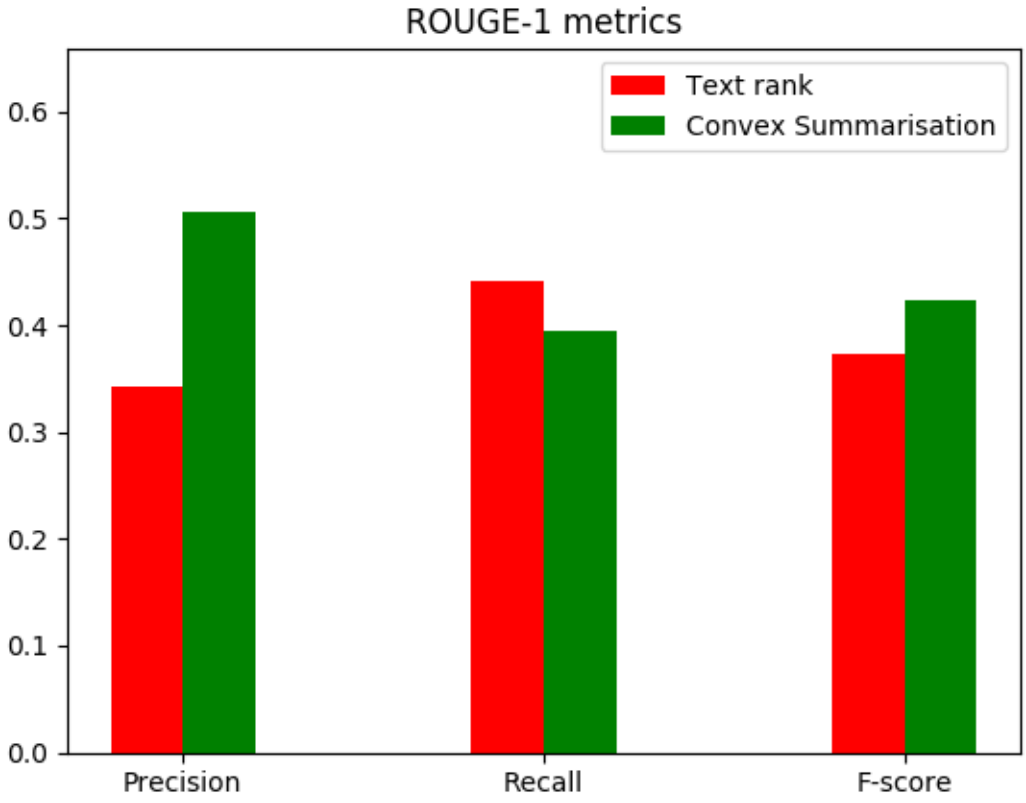
We have studied the impact of using side information

- $\lambda = 0.5$ and in vector **b** we set the first value to 1 and the last to 0.5
  - This choice is based on the empirical fact that these parts of the text tend to have higher importance
  - A small improvement is visible

Experimental Results

Improved algorithm compared by TextRank

- $\lambda = \lambda' = 0.5$, tol = 0.1
- This change can have a significant impact on the results without an important increase in the computation time



Experimental Results – Overview

Algorithm	Precision	Recall	F-score
TextRank	0.342	0.442	0.372
Convex Summarization (λ=0)	0.324	**0.446**	0.361
Convex Summarization (λ=0, tol=0.1)	0.326	0.434	0.359
Convex Summarization (λ=0.5, tol=0.1)	0.343	0.436	0.368
Improved Convex Summarization (λ=λ'=0.5, tol=0.1)	**0.506**	0.394	**0.423**

Conclusion

We have introduced a new algorithm for **extractive text summarization**, based on some simple and intuitive ideas, and tried to establish its properties

We have introduced to the field of text summarization/ information retrieval some ideas from the compressive sensing literature

Our method performs very well when compared with other similar algorithms (like TextRank)

Main **advantage**: possibility to naturally use side information

Drawback: execution time (is a few times higher than that of other methods)

Future Developments

Several improvements are possible

1. Use a specific algorithm for the convex program instead of a general solver, to increase execution speed
2. Add more postprocessing steps:
   - Eliminate the grammatically wrong sentences
   - Eliminate the ambiguities (“I”, “we”, etc.)

tsp 2018
July 4-6, 2018
Athens, Greece
2018
41st International Conference on
Telecommunications and Signal Processing
TSP

Automatic text summarization by mean-absolute constrained convex optimization

Claudiu POPESCU, Corneliu RUSU, **Lăcrimioara GRAMA**

Technical University of Cluj-Napoca
Faculty of Electronics, Telecommunications and Information Technology
Basis of Electronics Department
Signal Processing Group