Extractive Automatic Summarization: Does more linguistic knowledge make a difference?

Daniel S. Leite[1], Lucia H. M. Rino[1], Thiago A. S. Pardo[2], Maria das Graças V. Nunes[2]
Núcleo Interinstitucional de Lingüística Computacional (NILC)
http://www.nilc.icmc.usp.br
[1]Departamento de Computação, UFSCar
CP 676, 13565-905 São Carlos - SP, Brazil
[2]Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo
CP 668, 13560-970 São Carlos - SP, Brazil
{daniel_leite; lucia}@dc.ufscar.br , {taspardo,gracan}@icmc.usp.br

Abstract

In this article we address the usefulness of linguistic-independent methods in extractive Automatic Summarization, arguing that linguistic knowledge is not only useful, but may be necessary to improve the informativeness of automatic extracts. An assessment of four diverse AS methods on Brazilian Portuguese texts is presented to support our claim. One of them is Mihalcea's TextRank; other two are modified versions of the former through the inclusion of varied linguistic features. Finally, the fourth method employs machine learning techniques, tackling more profound and language-dependent knowledge.

1 Introduction

Usually, automatic summarization involves producing a condensed version of a source text through selecting or generalizing its relevant content. As a result, either an extract or an abstract will be produced. An extract is produced by copying text segments and pasting them into the final text preserving the original order. An abstract instead is produced by selecting and restructuring information from the source text. The resulting structure is thus linguistically realized independently of the surface choices of the source text. This comprises, thus, a rewriting task.

This article focuses solely on extracts of source texts written in Brazilian Portuguese. For extractive Automatic Summarization (AS), several methods have been suggested that are based upon statistics or data readily available in the source text. Word frequency (Luhn, 1958) and sentence position (Edmundson, 1969) methods are classic examples of that. Usually, extractive AS does not take into account linguistic and semantic knowledge in order to be portable to distinct domains or languages (Mihalcea, 2005). Graph-based methods aim at the same and have been gaining a lot of interest because they usually do not rely on any linguistic resource and run pretty fast. Exemplars of those are LexRank (Erkan and Radev, 2004) and TextRank (Mihalcea and Tarau, 2004). In spite of their potentialities, we claim that there is a compromise in pursuing a language-free setting: however portable a system may be, it may also produce extracts that lack the degree of informativeness needed for use. Informativeness, in the current context, refers to the ability of an automatic summarizer to produce summaries that convey most information of reference, or ideal, summaries. Our assessment thus aimed at verifying if parsimonious use of linguistic knowledge could improve extractive AS.

We argue that the lack of linguistic knowledge in extractive AS can be the reason for weak performance regarding informativeness. This argument follows from acknowledging that improvements on the scores usually obtained in that field have not been expressive lately. The most common metrics used to date, precision and recall, signal average results, suggesting that it is not enough to pursue completely language-free systems, no matter the current demands for portability in the global communication scenario. We focus here on TextRank, which can be used for summa-

rizing Brazilian Portuguese texts due to its language independence. To show that linguistic knowledge does make a difference in extractive AS, we compared four automatic summarizers: TextRank itself, two other modified versions of that, and SuPor-2 (Leite and Rino, 2006). TextRank works in a completely unsupervised way. Our two variations, although still unsupervised, include diverse linguistic knowledge in the preprocessing phase. SuPor-2 is the only machine learning-based system amongst the four ones, and it was built to summarize texts in Brazilian Portuguese, although it may be customized to other languages. Unlike the others, it embeds more sophisticated decision features that rely on varied linguistic resources. Some of them correspond to full summarization methods by themselves: *Lexical Chaining* (Barzilay and Elhadad, 1997), *Relationship Mapping* (Salton et al., 1997), and *Importance of Topics* (Larocca Neto et al., 2000). This is its unique and distinguishing characteristic.

In what follows we first review the different levels of processing in extractive AS (Section 2), then we describe TextRank and its implementation to summarize Brazilian Portuguese texts (Section 3). Our suggested modifications of TextRank are presented in Section 4, whilst SuPor-2 is described in Section 5. Finally, we compare the results of the four automatic summarizers when running on Brazilian Portuguese texts (Section 6), and make some remarks on linguistic independence for extractive AS in Section 7.

2 A Review of Automatic Summarization

Mani (2001) classifies AS methods based upon three levels of linguistic processing to summarize a text, namely:

- **Shallow level**. At this level only features at the surface of the text are explored. For example, location (Edmunson, 1969), sentence length and presence of signaling phrases (e.g., Kupiec et al., 1995). Combined, such features may yield a salience function that drives selection of sentences of the source text to include in a summary.
- **Entity level**. The aim here is to build an internal representation of the source text that conveys its entities and corresponding relationships. These amount to the information that allows identifying important text segments. Examples of such relations are word cooccurrence (e.g., Salton et al., 1997), synonyms and antonyms (e.g., Barzilay and Elhadad, 1997), logical relations, such as concordance or contradiction, and syntactic relations.
- **Discourse level**. At this level the whole structure of the source text is modeled, provided that its communicative goals can be grasped from the source text. The discourse structure is intended to help retrieving, e.g., the main topics of the document (e.g, Barzilay and Elhadad, 1997; Larocca Neto et al., 2000) or its rhetorical structure (e.g., Marcu, 1999), in order to provide the means for AS.

In this work we mainly focus on the entity level. Special entities and their relations thus provide the means to identify important sentences for building an extract. In turn, there is a loss of independence from linguistic knowledge, when compared to shallower approaches. Actually, apart from TextRank, the other systems described in this paper target entity level methods, as we shall see shortly.

3 The TextRank Method

The unsupervised TextRank method (Mihalcea and Tarau, 2004) takes after Google's PageRank (Brin and Page, 1998), a graph-based system that helps judge the relevance of a webpage through incoming and outgoing links. PageRank directed graphs represent webpages as nodes and their linking to other webpages as edges. A random walk model is thus applied to build a path between the nodes, in order to grade the importance of a webpage in the graph.

Similarly to grading webpages through traversing a graph, TextRank attempts to weight sentences of a text by building an undirected graph. Nodes are now sentences, and edges express their similarity degrees to other sentences in the text. Actually, the degree of similarity is based upon content overlap. As such, similarity degrees help assess the overall cohesive structure of a text. The more content overlap a sentence has with other sentences, the more important it is and more likely it is to be included in the extract.. Similarity is calculated through equation [1] (Mihalcea and Tarau, 2004), where S_i and S_j are sentences and w_k is a common token between them. The numerator is the sum of common words

between S_i and S_j. To reduce bias, normalization of the involved sentences length takes place, as shows the denominator.

$$Sim(S_i, S_j) = \frac{|\{w_k \mid w_k \in S_i \wedge w_k \in S_j\}|}{\log(|S_i|) + \log(|S_j|)} \quad [1]$$

Once the graph and all similarity degrees are produced, sentence importance is calculated by the random walk algorithm shown in equation [2]. $TR(V_i)$ signals sentence importance, d is an arbitrary parameter in the interval [0,1], and N is the number of sentences in the text. Parameter d integrates the probability of jumping from one vertex to another randomly chosen. Thus, it is responsible for random walking. This parameter is normally set to 0.85 (this value is also used in TextRank).

$$TR(V_i) = (1-d) + d \times \sum_{j=0}^{N-1} \left(TR(V_j) \times \frac{Sim(S_i, S_j)}{\sum_{k=0}^{N-1} Sim(S_j, S_k)} \right) \quad [2]$$

Initial *TR* similarity values are randomly set in the [0,1] interval. After successive calculations, those values converge to the targeted importance value. After calculating the importance of the vertices, the sentences are sorted in reverse order and the top ones are selected to compose the extract. As usual, the number of sentences of the extract is dependent upon a given compression rate.

Clearly, TextRank is not language dependent. For this reason Mihalcea (2005) could use it to evaluate AS on texts in Brazilian Portuguese, besides reporting results on texts in English. She also explored distinct means of representing a text without considering linguistic knowledge, emphasizing TextRank language and domain independence. She varies, e.g., the ways the graphs could be traversed using both directed and undirected graphs. Once a sentence is chosen to compose an extract, having undirected graphs makes possible, to look forward – from the sentence to its outgoing edges (i.e., focusing on the set of its following sentences in the text) – or to look backward, considering that sentence incoming edges and, thus, the set of its preceding sentences in the text.

Another variation proposed by Mihalcea is to replace the PageRank algorithm (Equation [2]) by HITS (Kleinberg, 1999). This works quite similarly to PageRank. However, instead of aggregating the scores for both incoming and outgoing links of a node in just one final score, it produces two independent scores. These are correspondingly named "authority" and "hub" scores.

4 Improving TextRank through variations on linguistic information

To improve the similarity scores between sentences in TextRank we fed it with more linguistic knowledge, yielding its two modified versions. The first variation focused just upon basic preprocessing; the second one, on the use of a thesaurus to calculate semantic similarity to promote AS decisions. However, we did not modify the main extractive algorithm of TextRank: we kept the graph undirected and used PageRank as the score determiner. Actually, we modified only the method of computing the edges weights.

4.1 Using Basic Preprocessing Methods

In applying Equation 1 for similarity scores, only exact matches between two words are allowed. Since in Brazilian Portuguese there are many morphological and inflexional endings for most words, this process becomes troublesome: important matches may be ignored. To overcome that, we used a stemmer for Brazilian Portuguese (Caldas Jr. et al., 2001) based upon Porter's algorithm (1980). We also removed stopwords from the source text, because they are not useful in determining similarity. The resulting version of TextRank is named hereafter 'TextRank+Stem+StopwordsRem'.

4.2 Using a Thesaurus

Our second TextRank variation involved plugging into the system a Brazilian Portuguese thesaurus (Dias-da-Silva et al., 2003). Our hypothesis here is that semantic similarity of the involved words is also important to improve the informativeness of the extracts under production. Thus, an extractive summarizer should consider not only word repetition in the source text, but also synonymy and antonymy.

Although plugging the thesaurus into the automatic summarizer did not imply changing its main method of calculating similarity, there were some obstacles to overcome concerning the following:

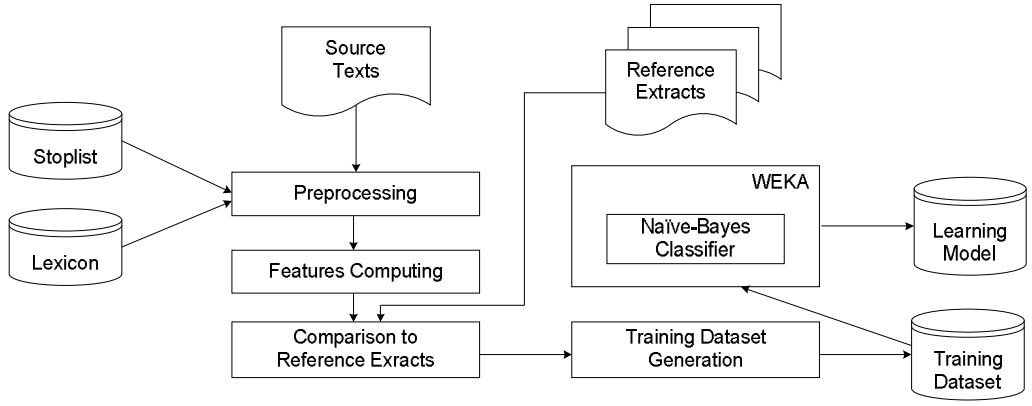

Figure 1. SuPor-2 training phase

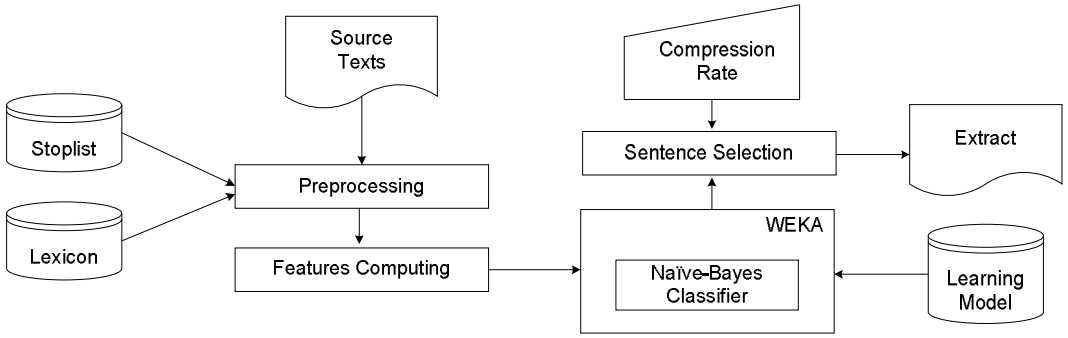

Figure 2. SuPor-2 extraction phase

a) Should we consider only synonyms or both synonyms and antonyms in addition to term repetition (reiteration)?

b) How to acknowledge, and disentangle, semantic similarity, when polissemy, for example, is present?

c) Once the proper relations have been determined, how should they be weighted? Just considering all thesaural relations to be equally important might not be the best approach.

Concerning (a), synonyms, antonyms, and term repetition were all considered, as suggested by others (e.g., Barzilay and Elhadad, 1997). We did not tackle (b) to choose the right sense of a word because of the lack of an effective disambiguation procedure for Brazilian Portuguese. Finally, in tackling (c) and, thus, grading the importance of the relations for sentence similarity, we adopted the same weights proposed by Barzilay and Elhadad (1997) in their lexical chaining method, which is discussed in more detail below. For both reiteration and synonymy, they assume a score of 10 for the considered lexical chain; for antonymy, they suggest a score of 7. The resulting version of TextRank is named here 'TextRank+Thesaurus'.

5 The SuPor-2 System

SuPor-2 is an extractive summarizer built from scratch for Brazilian Portuguese. It embeds different features in order to identify and extract relevant sentences of a source text. To configure SuPor-2 for an adequate combination of such features we employ a machine learning approach. Figures 1 and 2 depict the training and extraction phases, respectively.

For training, machine learning is carried out by a Naïve-Bayes classifier that employs Kernel methods for numeric feature handling, known as *Flexible Bayes* (John and Langley, 1995). This environment is provided by WEKA[1] (Witten and Frank, 2005), which is used within SuPor-2 itself. The training corpus comprises both source texts and corresponding reference extracts. Every sentence from a source text is represented in the training dataset as a tuple of the considered features.

[1] Waikato Environment for Knowledge Analysis. Available at http://www.cs.waikato.ac.nz/ml/weka/ (December, 2006)

Each tuple is labeled with its class, which signals if the sentence appears in a reference extract. The class label will be true if the sentence under focus matches a sentence of the reference extract and false otherwise.

Once produced, the training dataset is used by the Bayesian classifier to depict the sentences that are candidates to compose the extract (Figure 2). In other words, the probability for the "true" class is computed and the top-ranked sentences are selected, until reaching the intended compression rate.

When computing features, three full methods (M) and four corpus-based parameters (P) are considered. Both methods and parameters are mapped onto the feature space and are defined as follows:

(M) Lexical Chaining (Barzilay and Elhadad, 1997). This method computes the connectedness between words aiming at determining lexical chains in the source text. The stronger a lexical chain, the more important it is considered for extraction. Both an ontological resource and WordNet (Miller et al., 1990) are used to identify different relations, such as synonymy or antonym, hypernymy or hyponymy, that intervene to compute connectedness. The lexical chains are then used to produce three sets of sentences. To identify and extract sentences from those sets, three heuristics are made available, namely: (H1) selecting every sentence *s* of the source text based on each member *m* of every strong lexical chain of the text. In this case, *s* is the sentence that contains the first occurrence of *m*; (H2) this heuristics is similar to the former one, but instead of considering all the members of a strong lexical chain, it uses only the representative ones. A representative member is one whose frequency is greater than the average frequency of all words in the chain; (H3) a sentence *s* is chosen by focusing only on representative lexical chains of every topic of the source text. In SuPor-2, the mapping of this method onto a nominal feature is accomplished by signaling which heuristics have recommended the sentence. Thus, features in the domain may range over the values {'None', 'H1', 'H2', 'H3', 'H1H2', 'H1H3', 'H2H3', 'H1H2H3'}.

(M) Relationship Mapping (Salton et al., 1997). This method performs similarly to the previous one and also to TextRank in that it builds up a graph interconnecting text segments. However, it considers paragraphs instead of sentences as vertices. Hence, graph edges signal the connectiveness of the paragraphs of the source text. Similarity scores between two paragraphs are thus related to the degree of connectivity of the nodes. Similarly to Lexical Chaining, Salton et al. also suggest three different ways of producing extracts. However, they now depend on the way the graph is traversed. The so-called dense or bushy path (P1), deep path (P2), and segmented path (P3) aim at tackling distinct textual problems that may damage the quality of the resulting extracts. The dense path considers that paragraphs are totally independent from each other, focusing on the top-ranked ones (i.e., the ones that are denser). As a result, it does not guarantee that an extract will be cohesive. The deep path is intended to overcome the former problem by choosing paragraphs that may be semantically inter-related. Its drawback is that only one topic, even one that is irrelevant, may be conveyed in the extract. Thus, it may lack proper coverage of the source text. Finally, the segmented path aims at overcoming the limitations of the former ones, addressing all the topics at once. Similarly to Lexical Chaining, features in the Relationship method range over the set {'None','P1','P2','P3', 'P1P2', 'P1P3', 'P2P3', 'P1P2P3'}.

(M) Importance of Topics (Larocca Neto et al., 2000). This method also aims at identifying the main topics of the source text, however through the TextTiling algorithm (Hearst, 1993). Once the topics of the source text have been determined, the first step is to select sentences that better express the importance of each topic. The amount of sentences, in this case, is proportional to the topic importance. The second step is to determine the sentences that will actually be included in the extract. This is carried out by measuring their similarity to their respective topic centroids (Larocca Neto et al., 2000). The method thus signals how relevant a sentence is to a given topic. In SuPor-2 this method yields a numeric feature whose value conveys the harmonic mean between the sentence similarity to the centroid of the topic in which it appears and the importance of that topic.

(P) Sentence Length (Kupiec et al., 1995). This parameter just signals the normalized count of words of a sentence.

(P) Sentence Location (Edmundson, 1969). This parameter takes into account the position of a sentence in the text. It is valued, thus, in {'II','IM','IF','MI','MM','MF','FI','FM','FF'}. In this set the first letter of each label signals the position of the sentence within a paragraph (Initial, Medium, or Final). Similarly, the second letter signals the position of the paragraph within the text.

(P) Occurrence of proper nouns (e.g., Kupiec et al., 1995). This parameter accounts for the number of proper nouns in a sentence.

(P) Word Frequency (Luhn, 1958). This parameter mirrors the normalized sum of the word frequency in a sentence.

SuPor-2 provides a flexible way of combining linguistic and non-linguistic features for extraction. There are profound differences from TextRank. First, it is clearly language-dependent. Also, its graph-based methods do not assign weights to their vertices in order to select sentences for extraction. Instead, they traverse a graph in very specific and varied ways that mirror both linguistic interdependencies and important connections between the nodes.

6 Assessing the Four Systems

To assess the degree of informativeness of the systems previously described, we adopt ROUGE[2] (Lin and Hovy, 2003), whose recall rate mirrors the informativeness degree of automatically generated extracts by correlating automatic summaries with ideal ones.

The two modified versions of TextRank require linguistic knowledge but at a low cost. This is certainly due to varying only preprocessing, while the main decision procedure is kept unchanged and language-independent. Those three systems do not need training, one of the main arguments in favor of TextRank (Mihalcea and Tarau, 2004). In contrast, SuPor-2 relies on training and this is certainly one of its main bottlenecks. It also employs linguistic knowledge for both preprocessing and extraction, which TextRank purposefully avoids. However, using WEKA has made its adjustments less demanding and more consistent, indicating that scaling up the system is feasible.

[2] Recall-Oriented Understudy for Gisting Evaluation. Available at http://haydn.isi.edu/ROUGE/ (January, 2007).

In our assessment, the same single-document summarization scenario posed by Mihalcea (2005) was adopted, namely: (a) we considered the Brazilian Portuguese TeMário corpus (Pardo and Rino, 2003); (b) we used the same baseline, which selects top-first sentences to include in the extract; (c) we adopted a 70-75% compression rate, making it compatible with the compression rate of the reference summaries; and (d) ROUGE was used for evaluation in its Ngram(1,1) 95% confidence rate setting, without stopwords removal. TeMário comprises 100 newspaper articles from online Brazilian newswire. A set of corresponding manual summaries produced by an expert in Brazilian Portuguese is also included in TeMário. These are our reference summaries.

For training and testing SuPor-2, we avoided building an additional training corpus by using a 10-fold cross-validation procedure. Finally, we produced three sets of extracts using 'TextRank + Stem + StopwordsRem', 'TextRank + Thesaurus', and SuPor-2 on the TeMário source texts. Results for informativeness are shown in Table 1. Since Mihalcea's setting was kept unchanged, we just included in that table the same results presented in (Mihalcea, 2005), i.e., we did not run her systems all over again. We also reproduced for comparison the TextRank variations reported by Mihalcea, especially regarding graph-based walks by PageRank and HITS. Shaded lines correspond to our suggested methods presented in Sections 4 and 5, which involve differing degrees of dependence on linguistic knowledge.

It can be seen that 'TextRank+Thesaurus' and 'TextRank+Stem+StopwordsRem' considerably outperformed all other versions of TextRank. Compared with Mihalcea's best version, i.e., with 'TextRank (PageRank - backward)', those two methods represented a 6% and 9% improvement, respectively. We can conclude that neither the way the graph is built nor the choice of the graph-based ranking algorithm affects the results as significantly as do the linguistic-based methods. Clearly, both variations proposed in this paper signal that linguistic knowledge, even if only used at the preprocessing stage, provides more informative extracts than those produced when no linguistic knowledge at all is considered. Moreover, at that stage little modeling and computational effort is demanded, since lexicons, stoplists, and thesauri

are quite widely available nowadays for several Romance languages.

Even the baseline outperformed most versions of TextRank, showing that linguistic independence in a random walk model for extractive AS should be reconsidered. Actually, this shows that linguistic knowledge *does make a difference*, at least for summarizing newswire texts in Brazilian Portuguese.

In addition, SuPor-2 performance exceeds the best version of TextRank that uses no linguistic knowledge – 'TextRank (PageRank - backward)' – by about 14%.

System	ROUGE NGram(1,1)
SuPor-2	0,5839
TextRank+Thesaurus	0,5603
TextRank+Stem+StopwordsRem	0,5426
TextRank (PageRank - backward)	0,5121
TextRank (HIT hub - forward)	0,5002
TextRank (HITS authority - backward)	0,5002
Baseline	0,4963
TextRank (PageRank - undirected)	0,4939
TextRank (HITS authority - forward)	0,4834
TextRank (HIT hub - backward)	0,4834
TextRank (HITS authority -undirected)	0,4814
TextRank (HIT hub - undirected)	0,4814
TextRank (PageRank - forward)	0,4574

Table 1. Informativeness comparison between extractive summarizers

7 Final Remarks

A critical issue in the comparison presented above is the contrast between having an unsupervised or supervised summarizer, which is related to the issue on having linguistic-independent extractive summarizers. Perhaps the question that we should pose here is how interesting and useful an extractive automatic summarizer that is totally independent from linguistic knowledge can actually be. To our view, the more non-informative an extract, the less useful it may be. So, summarizers that do not reach a minimum threshold concerning informativeness are deemed to failure nowadays. Clearly, SuPor-2 requires language-dependent resources, but its main extraction procedure is still general enough to make it portable and adaptable to new domains and languages. Hence, SuPor-2 assessment suggests that it may be interesting to scale up SuPor-2.

Considering that SuPor-2 is one of the best extractive summarizers for Brazilian Portuguese texts (Leite and Rino, 2006) and 'TextRank+Thesaurus' performed only 4% below it, we can also argue in favor of providing even simple linguistic procedures for extractive AS. The latter system shows that TextRank can yield extracts nearly as informative as those produced by the former, when embedding stemming and stopwords removal. It can also perform AS with little computational effort and no training, when compared to the supervised SuPor-2. As a conclusion, we see that some linguistic knowledge may boost TextRank performance without too much effort, since language-dependent resources for preprocessing texts in natural language are usually available and easy to handle, concerning our addressed approach.

There are many experiments that may be derived from our discussion in this paper (1) Although the reported results suggest that linguistic knowledge does make a difference when embedded in language-free extractive summarizers, the performance of the top systems assessed through ROUGE should be more comprehensively licensed through additional assessment tasks. (2) These could also incorporate other graph-based algorithms than TextRank, such as the LexRank one, aiming at reassuring our claim and scaling up graph-based approaches. (3) Since we addressed language-independence (thus portability) versus language-dependence for informativeness, it would also be interesting to explore other domains or languages to support our claim or, at least, to look for other findings to confirm if linguistic knowledge indeed makes a difference. (4) Other TextRank variations could also be explored, to see if adding more features would make TextRank closer to SuPor-2.

Acknowledgements

This work has been supported by the Brazilian research funding agencies CNPq, CAPES and FAPESP.

References

B. C. Dias-da-Silva, M. F. Oliveira, H. R. Moraes, C. Paschoalino, R. Hasegawa, D. Amorin and A. C. Nascimento. 2000. Construção de um Thesaurus Eletrônico para o Português do Brasil. In *Proc. of the V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR 2000*), São Carlos, Brasil , 1-11.

C. Lin and E. H. Hovy. 2003. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *Proceedings of Language Technology Conference (HLT-NAACL 2003),* Edmonton, Canada.

D. Marcu. 1999. Discourse Trees Are Good Indicators of Importance in Text. In: Mani, I., Maybury, M. T. (Eds.). 1999. *Advances in Automatic Text Summarization*. MIT Press.

D. S. Leite and L. H. M. Rino. 2006. Selecting a Feature Set to Summarize Texts in Brazilian Portuguese. In: J. S. Sichman et al. (eds.): *Proc. of 18th. Brazilian Symposium on Artificial Intelligence (SBIA'06) and 10th. Ibero-American Artificial Intelligence Conference (IBERAMIA'06)*. Lecture Notes on Artificial Intelligence, No. 4140, Springer-Verlag, 462-471.

G. Erkan and D R. Radev. 2004. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research* 22:457-479

G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. Miller. 1990. Introduction to WordNet: An Online Lexical Database. *International Journal of Lexicography* 3(4):235-244

G. Salton, and C. Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management* 24 : 513-523.. Reprinted in: K. Sparck-Jones and P. Willet (eds.). 1997. *Readings in Information Retrieval*, Morgan Kaufmann, 323-328.

H. Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development* 2:159-165

H. P. Edmundson. 1969. New methods in automatic extracting. *Journal of the Association for Computing Machinery* 16:264-285.

I. Witten and E. Frank. 2005. *Data Mining: Practical machine learning tools and techniques*, 2nd ed. Morgan Kaufmann, San Francisco.

I. Mani. 2001. *Automatic Summarization*. John Benjamin's Publishing Company.

I. Mani and M. T. Maybury. 1999. *Advances in Automatic Text Summarization*. MIT Press.

J. Caldas Junior, C. Y. M. Imamura and S. O. Rezende. Avaliação de um Algoritmo de Stemming para a Língua Portuguesa. In *Proc. of the 2nd Congress of Logic Applied to Technology (LABTEC'2001)*, vol. II. Faculdade SENAC de Ciências Exatas e Tecnologia, São Paulo, Brasil (2001), 267-274.

J. M. Kleinberg. 1999. Authoritative sources in hyperlinked environment. *Journal of the ACM*, 46(5):604-632.

J. Kupiec, J. Pedersen and F. Chen. 1995. A trainable document summarizer. In: *Proc. of the 18th ACM-SIGIR Conference on Research & Development in Information Retrieval*, 68-73.

J. Larocca Neto, A. D. Santos, C. A. A. Kaestner and A. A. Freitas. 2000. Generating Text Summaries through the Relative Importance of Topics. *Lecture Notes in Artificial Intelligence*, No. 1952. Springer-Verlag, 200-309

M. A. Hearst. 1993. TextTiling: A Quantitative Approach to Discourse Segmentation. Technical Report 93/24. University of California, Berkeley.

M. F. Porter. 1980. An Algorithm for Suffix Stripping. *Program*, 14 (3) : 130-137

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing Order into Texts. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, Barcelona, Spain, July.

R. Mihalcea. 2005. Language Independent Extractive Summarization. In: *Proc. of the 43th Annual Meeting of the Association for Computational Linguistics*, Companion Volume (ACL2005), Ann Arbor, MI, June.

R. Barzilay and M. Elhadad. 1997. Using lexical chains for text summarization. In: *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97)*, ACL, Madrid, Spain.

S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30:1-7.

T. A. S. Pardo and L.H.M. Rino. 2003. TeMário: A corpus for automatic text summarization (in Portuguese). NILC Tech. Report NILC-TR-03-09