# A Comparison of Automatic Summarizers of Texts in Brazilian Portuguese

Lucia Helena Machado Rino[1], Thiago Alexandre Salgueiro Pardo[1],
Carlos Nascimento Silla Jr.[2], Celso Antônio Alves Kaestner[2], Michael Pombo[1]

[1] Núcleo Interinstitucional de Lingüística Computacional (NILC/São Carlos)
DC/UFSCar – CP 676, 13565-905 São Carlos, SP, Brazil
http://www.nilc.icmc.usp.br
thiago@nilc.icmc.usp.br;{michaelp; lucia}@dc.ufscar.br

[2] Pontifícia Universidade Católica do Paraná (PUC-PR)
Av. Imaculada Conceição 1155, 80215-901 Curitiba, PR, Brazil
{silla; kaestner}@ppgia.pucpr.br

**Abstract.** Automatic Summarization (AS) in Brazil has only recently become a significant research topic. When compared to other languages initiatives, such a delay can be explained by the lack of specific resources, such as expressive lexicons and corpora that could provide adequate foundations for deep or shallow approaches on AS. Taking advantage of having commonalities with respect to resources and a corpus of texts and summaries written in Brazilian Portuguese, two NLP research groups have decided to start a common task to assess and compare their AS systems. In the experiment five distinct extractive AS systems have been assessed. Some of them incorporate techniques that have been already used to summarize texts in English; others propose novel approaches to AS. Two baseline systems have also been considered. An overall performance comparison has been carried out, and its outcomes are discussed in this paper.

## 1 Introduction

We definitely live in the information explosion era. A recent study from Berkeley [12] indicates there were 5 million terabytes of new information created via print, film, magnetic, and optical storage media in 2002, and the www alone contains about 170 terabytes of information on its surface. This is about twice the data generated in 1999, given an increasing rate at about 30% each year. Conversely, to use this information is very hard. Problems like information retrieval and extraction, and text summarization became important areas in Computer Science research.

Especially concerning Automatic Summarization (AS), we focus on extractive methods in order to produce extracts of texts written in Brazilian Portuguese. Extracts, in this context, are summaries produced automatically on the basis of superficial, empirical or statistical, techniques, broadly known as extractive methods [15]. These actually aim at producing summaries that consist entirely of material copied, usually sentences, from the source texts. Typically, extracts or summaries automatically generated have 10 to 30% of

the original text length – being faster to read – but must contain enough information to satisfy the user's needs [13].

Five AS systems were assessed, all of them sharing the same linguistic resources, when applicable. Only precision (P) and recall (R) have been considered, for practical reasons: being extractive, all the summarizers under consideration could be automatically assessed to calculate P and R. The performance of those AS systems could thus be compared, in order to identify the features that apply better to a genre-specific text corpus in Brazilian Portuguese.

To calculate P and R, ideal summaries – extractive versions of the manual summaries – have been used, which have been automatically produced by a specific tool, a generator of ideal extracts (available in http://www.nilc.icmc.usp.br/~thiago). This tool is based upon the widely known vector space model and the cosine similarity measure [25], and works as follows: 1) for each sentence in the manual summary the most similar sentence in the text is obtained (through the cosine measure); 2) the most representative sentences are selected, yielding the corresponding ideal, extractive, summary. This procedure works as suggested by [14], i.e., it is based on the premise that ideal extracts should be composed of as many sentences (the most similar ones) as the ones in the corresponding manual summary.

As we shall see, some of the systems being assessed had to be trained. In this case, the very same pre-processing tools and data have been used by all of them. We chose TeMário [19] (available in http://www.linguateca.pt/Repositorio/TeMario), a corpus of 100 newspaper texts (c.a. 613 words, or 1 to 2 ½ pages long) that has been built for AS purposes, as the only input for the assessment reported here. Those texts have been withdrawn from online regular Brazilian newspapers, the Folha de São Paulo (60 texts) and the Jornal do Brasil (40 texts) ones. They are equally distributed amongst distinct domains, namely, those respecting to free author views, critiques, world, politics, and foreign affairs. The summaries that come along with this corpus are those hand-produced by the consultant on the Brazilian Portuguese language.

Details of the considered systems and their assessment are given below. In Section 2, we outline the main features of each system under focus. In Section 3 we describe the experiment itself and a thorough discussion on their overall rating. Finally, in section 4 we address the outcomes of the reported assessment, concerning the potentialities to apply AS for Brazilian Portuguese texts of a particular genre.


## 2  Extractive AS systems under focus

Each of the assessed AS systems tackles a particular AS strategy. Specially, three of them suggest novel approaches, as follows: (a) **Gist Summ**arizer (GistSumm) [20], focuses upon the matching of lexical items of the source text against lexical items of a gist sentence, supposed to be the sentence of the source text that best expresses its main idea, which is previously determined by means of a word frequency distribution; (b) **T**erm **F**requency-**I**nverse **S**entence **F**requency-based **Summ**arizer (TF-ISF-Summ) [9], adapts Salton's TF-IDF information retrieval measure [25] in that, instead of signaling the documents to retrieve, it pinpoints those sentences of a source text that must be included

in a summary; (c) **Neural Summ**arizer (NeuralSumm) [21] is based upon a neural network that, after training, is capable of identifying relevant sentences in a source text for producing the extract. Added to those, we employ a classification system (ClassSumm) that produces extracts based on a Machine Learning (ML) approach, in which summarization is considered as a classification task. Finally, we use Text **Su**mmarization in **Por**tuguese (SuPor) [17], a system aiming at exploring alternative methodologies that have been previously suggested to summarize texts in English. Based on a ML technique, it allows the user to customize surface and/or linguistic features to be handled during summarization, permitting one to generate diverse AS engines. In the assessment reported in this paper, SuPor has been customized to just one AS system.

All the systems consistently incorporate language-specific resources, aiming at ensuring the accuracy of the assessment. The most significant tools already available for Brazilian Portuguese are a part-of-speech tagger [1], a parser [16], and a stemmer based upon Porter's algorithm [3]. Linguistic repositories include a lexicon [18], and a list of discourse markers, which is derived from the DiZer system [22]. Additionally, a stoplist (i.e., a list of stopwords, which are too common and, therefore, irrelevant to summarization) and a list of the commonest lexical items that signal anaphors are also used. Apart from the discourse markers and the lexical items lists, which are used only by ClassSumm, and the tagger and parser, which are not used by GistSumm and NeuralSumm, the other resources are shared amongst all the systems.

Text pre-processing is also common to all the systems. It involves text segmentation, through delimiting sentences by applying simple rules based on punctuation marks, case folding and stemming, and stopwords removal. In the following we briefly describe each AS system.

### 2.1 The GistSumm System

GistSumm is an automatic summarizer based on a novel extractive method, called gist-based method. For GistSumm to work, the following premises must hold: (a) every text is built around a main idea, namely, its gist; (b) it is possible to identify in a text just one sentence that best expresses its main idea, namely, the gist sentence. Based on them, the following hypotheses underlie GistSumm methodology: (I) through simple statistics the gist sentence or an approximation of it is determined; (II) by means of the gist sentence, it is possible to build coherent extracts conveying the gist sentence itself and extra sentences from the source text that complement it.

GistSumm comprises three main processes: text segmentation, sentence ranking, and extract production. Sentence ranking is based on the keywords method [11]: it scores each sentence of the source text by summing up the frequency of its words and the gist sentence is chosen as the most highly scored one. Extract production focuses on selecting other sentences from the source text to include in the extract, based on: (a) gist correlation and (b) relevance to the overall content of the source text. Criterion (a) is fulfilled by simply verifying co-occurring words in the candidate sentences and the gist sentence, ensuring lexical cohesion. Criterion (b) is fulfilled by sentences whose score is above a threshold, computed as the average of all the sentence scores, to guarantee that only

relevant-to-content sentences are chosen. All the selected sentences above the cutoff are thus juxtaposed to compose the final extract.

GistSumm has already undergone several evaluations, the main one being DUC'2003 (Document Understanding Conference). According to this, Hypothesis I above has been proved to hold. Other methods than the keywords one were also used for sentence ranking. The keywords one outperformed all of them.

## 2.2 The TF-ISF-Summ System

TF-ISF-Summ is an automatic summarizer that makes use of the TF-ISF (Term-Frequency Inverse-Sentence-Frequency) metric to rank sentences in a given text and then extract the most relevant ones. Similarly to GistSumm, the approach used by this system has also three main steps: (1) text pre-processing (2) sentence ranking, and (3) extract generation. Differently from that, in order to rank the sentences, it calculates the mean TF-ISF of each sentence, as proposed in [9]: (1) each sentence is considered as a fragment of the text; (2) given a sentence, the TF-ISF metric for each term (similar to the TF-IDF metric [25]) is calculated: TF is the frequency of the term in the document and ISF is a function of the number of sentences in which the term appears; (3) finally, the TF-ISF for the whole sentence is computed as the arithmetic mean of all the TF-ISF values of its terms. Sentences with the highest mean-TF-ISF score and above the cutoff are selected to compose the output extract.

The method showed to be only as good as the random sentences approach in the experiments made by Larocca Neto [8] for documents in English.

## 2.3 The NeuralSumm System

NeuralSumm system makes use of a ML technique, and runs on four processes: (1) text segmentation, (2) features extraction, (3) classification, and (4) extract production. It is primarily unsupervised, since it is based on a self-organizing map (SOM) [6], which clusters information from the training texts. NeuralSumm produces two clusters: one that represents the important sentences of the training texts (and, thus, should be included in the extract) and another that represents the non-important sentences (and, thus, should be discarded). To our knowledge, it is the first time a SOM has been used to help determining relevant sentences in AS.

During AS, after analyzing the source text, features extraction focuses on each sentence, in order to collect the following features: (i) sentence length, (ii) sentence position in the source text, (iii) sentence position in the paragraph it belongs to, (iv) presence of keywords in the sentence, (v) presence of gist words in the sentence, (vi) sentence score by means of its words frequency, (vii) sentence score by means of TF-ISF and (viii) presence of indicative words in the sentence. It is worth noticing that keywords are limited to the two most frequent words in the text, gist words are the composing words of the gist sentence, and indicative words are genre-dependent and could be corresponding to, e.g., 'problem', 'solution', 'conclusion', or 'purpose', in scientific texts. Both feature (vi) and the gist sentence are calculated in the same way as they are in GistSumm. The

rationale behind incorporating these features in NeuralSumm may be found in [21]. Sentence classification is carried out by considering every feature of each sentence, which is given as input to the SOM. This finally classifies the sentences as important or non-important, the important ones being selected and juxtaposed to compose the final extract.

NeuralSumm SOM was already compared to other ML techniques. It proved to be better than Naïve Bayes, decision trees and decision rules methods, with an error decreasing rate to the worst case of c.a. 10% [21].

## 2.4 The ClassSumm System

The Classification System was proposed by Larocca Neto et al. [10] and uses a ML approach to determine relevant segments to extract from source texts. Actually, it is based on a Naïve Bayes classifier.

To summarize a source text, the system performs the same four processes that NeuralSumm, as previously explained. Text pre-processing is similar to the one performed by TF-ISF-Summ. Features extracted from each sentence are of two kinds: statistical, i.e., based on measures and counting taken directly from the text components, and linguistic, in which case they are extracted from a simplified argumentative structure of the text, produced by a hierarchical text agglomerative clustering algorithm. A total of 16 features are associated to each sentence, to know: (a) mean-TF-ISF, (b) sentence length, (c) sentence position in the source text, (d) similarity to title, (e) similarity to keywords, (f) sentence-to-sentence cohesion, (g) sentence-to-centroid cohesion, (h) main concepts – the most frequent nouns that appear in the text, (i) occurrence of proper nouns, (j) occurrence of anaphors, (k) occurrence of non-essential information. Features (d), (e), (f) and (g) use the cosine measure to calculate similarity; features (h) and (i) use the POS Tagger; finally, features (j) e (k) use fixed lists, as mentioned before. The remaining are linguistic features, based on the binary tree that represents the argumentative structure of the text, where each leaf is associated to a sentence and the internal nodes are associated to partial clusters of sentences. These features are: (l) the depth of each sentence in the tree, and (m) four features that represent specific directions in a given level of the tree (height 1,2,3,4) that indicate, for each depth level, the direction taken by the path from the root to the leaf associated with the sentence.

Extract generation is considered as a two-valued classification problem: sentences should be classified as relevant-to-extract or not. According to the values of the features for each sentence, the classification algorithm must "learn" which ones must belong to the summary. Finally, the sentences to include in the extract will be those above the cutoff and, thus, those with the highest probabilities of belonging to it.

In the experiment reported in this article, the only unused feature was the keywords similarity, because the TeMário corpus does not convey a list of keywords. Compared to the other systems, ClassSumm uses two extra lists: one with indicators of main concepts and another with the commonest anaphors. Although there are no such fixed lists to Brazilian Portuguese, we followed Larocca Neto's [8] suggestions, incorporating to the current version of the system the corresponding pronoun anaphors for English, such as 'this', 'that', 'those', etc.

ClassSumm was evaluated on a TIPSTER corpus of 100 news stories for training, and two test procedures, namely, one that has used 100 automatic summaries and another that has used 30 manual extracts [10], in which it outperforms the "from-top" – those from the beginning of the source text, and random order baselines.

## 2.5 The SuPor System

Similarly to some of the above systems, SuPor also conveys two distinct processes: training and extracting based on a Bayesian method, following [7]. Unlike them, it embeds a flexible way to combine linguistic and non-linguistic constraints for extraction production. AS options include distinct suggestions originally aimed at texts in English, which have been adapted to Brazilian Portuguese. To configure an AS strategy, SuPor must thus be customized by an expert user [17].

In SuPor, relevant features for classification are (a) sentence length (minimum of 5 words); (b) words frequency [11]; (c) signaling phrases; (d) sentence location in the texts; and (e) occurrence of proper nouns. As a result of training, a probabilistic distribution is produced, which entitles extraction in SuPor. For this, only features (a), (b), (d) and (e) are used, along with lexical chaining [2]. Adaptations from the originals have been made for Portuguese, to know: (i) for lexical chaining computation, a thesaurus [4] for Brazilian Portuguese is used; (ii) sentence location (10% of the first and 5% of the last sentences of a source text are considered); (iii) proper nouns are those that are not abbreviations, occur more than once in the source text and do not appear at the beginning of a sentence; (iv) a minimum threshold has been set for the selection of the most frequent words: each term of the source text is frequency-weighed, and the total weight of the text is produced; then the average weight, along with its standard deviation is taken as the cutoff of frequent words.

SuPor works in the following way: firstly, the set of features of each sentence are extracted. Secondly, for each of the sets, the Bayesian classifier provides its probability, which will enable top-sentences to be included in the output extract.

SuPor performance has been previously assessed through two distinct experiments that also focused on newspaper articles and their ideal extracts, produced by the generator of ideal extracts already referred to. However, testing texts had nothing to do with TeMário. Both experiments addressed the representativeness of distinct groupings of features. Overall, the features grouping that have been most significant included lexical chaining, sentence length and proper nouns (avg.F-measure=40%).

## 3 Experiments and Results

We proceeded to a blackbox-type evaluation, i.e., only comparing the systems outputs. The main limitation imposed to the experiment was making it efficient: to compare the performance of the five systems, evaluation should be entirely automatic. As a result, only co-selection measures [23], more specifically P, R, and F-measure were used. We could not compare either automatic extracts with TeMário manual summaries because they are hand-built and do not allow for a viable automatic evaluation. For this reason,

the corresponding ideal extracts were used, as described in Section 1. In relation to the systems that need training, to assure non-biasing, a 10-fold cross validation has been used (each fold comprising 10 texts).

We also included in the evaluation two baseline methods: the one based just upon the selection of from-top sentences and the other that chooses them at random (hereafter, From-top and Random order methods, respectively). Following the same approach, the extracts contain as many sentences as the cutoff allows in this case.

In the AS context, the metrics under focus here are defined as follows: (a) compression rate is 30%. It has been chosen to conform to the sizes of both, the manual summaries (length ranging from 25 to 30%) and ideal extracts; (b) Let N be the total number of sentences in the output extract; M be the total number of sentences in the ideal extract; NR be the number of relevant sentences included in the output extract, i.e., the number of coinciding sentences between the output and its corresponding ideal extract; (c) precision and recall are defined by P=NR/N and R=NR/M, and F-measure is the balance metric between P and R, F=2*P*R/(P+R).

All the systems were independently run. Table 1 shows the averaged precision, recall and F-measure metrics of each system obtained in the experiments, with last column indicating the relative performance of each system as the percentage over the random order baseline method, i.e. (F-measure/F-measure-random-baseline - 1).

**Table 1**. Systems performance (in %)

| Systems | Avg. P | Avg. R | Avg. F | % over random |
|---|---|---|---|---|
| SuPor | 44.9 | 40.8 | 42.8 | 38 |
| ClassSumm | 45.6 | 39.7 | 42.4 | 37 |
| From-top | 42.9 | 32.6 | 37.0 | 19 |
| TF-ISF-Summ | 39.6 | 34.3 | 36.8 | 19 |
| GistSumm | 49.9 | 25.6 | 33.8 | 9 |
| NeuralSumm | 36.0 | 29.5 | 32.4 | 5 |
| Random order | 34.0 | 28.5 | 31.0 | 0 |

Overall, the combination of features that lead to SuPor performance is [location, words frequency, length, proper nouns, lexical chaining]. SuPor performance may well be due to the inclusion of lexical chaining, since this is its most distinctive feature. Meaningfully, training has also counted on signaling phrases, which has been considered only in SuPor. This, added to lexical chaining, may well be one of the reasons for SuPor outperformance. Lexical chaining also has a close relationship to the innovative features added to ClassSumm, the second topmost system. Especially, it focuses on the strongest lexical chains, whilst ClassSumm focuses on sentence-to-sentence and sentence-to-centroid for cohesion.

Close performance between SuPor and ClassSumm can also be explained through the relationship between the following features combinations, respectively: [words frequency, signaling phrases] and [mean TF-ISF, indicator of main concepts, similarity to title]. This is justified by acknowledging that the mean TF-ISF is based on words frequency and main concepts and titles may signal phrases that lead to decision patterns.

Both topmost systems include features that have been formerly indicated for good performance, when individually taken (see the generalization of Edmundson's [5] para-

digm in [13]): sentences location and cue phrases (i.e., the referred signaling ones). Additionally, both have been trained through a Bayesian classifier, with a considerable overlap of features. Keywords, which have been considered the poorest in Edmundson's model [5], have not been considered in any of them. In all, they substantially differ only through the anaphors and non-essential information features, although location, in Class-Summ, addresses the argumentative tree of a source text, instead of its surface structure, as it is in SuPor.

TF-ISF-Summ, which has a worse performance than ClassSumm, coincides with that in the combination [words frequency, mean TF-ISF], for the same reasons given above. Although its performance is not substantially far from that of SuPor, its upperbound is a baseline. This may also suggest that what distinguishes SuPor is not the word frequency, neither is the mean TF-ISF measure in ClassSumm.

Not surprisingly, GistSumm performance is farther than the other systems referred to, for it is based mainly upon words distribution, which has been repeatedly evidenced as a non-expressive feature. However, evidences provided by the DUC'2003 evaluation show that GistSumm is effective in determining the gist sentence. In that evaluation, GistSumm scored 3.12 in a 0-4 scale for usefulness. This metric was presented to DUC judges in the following way: their score of any given summary should indicate how useful the summary was in retrieving the corresponding source text (0 indicating no use at all and 4, totally useful, i.e., as good as having the full text instead.[1] So, the problem must be in the extraction module instead. Although this system achieved the best P, its R is the worst, even worse than the baselines. Recall could be improved, for example, if gist words were spread over the whole source text, which does not seem to be the case in newspaper texts, where the gist is usually in the lead sentences.

Although NeuralSumm is based on a combination of most of the features embedded in SuPor and ClassSumm, its performance is much worse. This may be due to its training on SOM, as well as on the means training has been carried out (e.g., a non-significant corpus) or, ultimately, on the features themselves, which also include word frequency.

The From-top method occupies, as expected, the 3rd position in the F-measure scale. Being composed of newspaper texts of varied domains, the test corpus has an expressive feature: lead sentences usually are the most relevant ones. Distinction between that and the other 2 topmost systems may be due to the sophistication of combining distinctive features. Since most of them coincide, but cohesive indicators, lexical chaining (SuPor) and sentence-to-sentence or sentence-to-centroid cohesion (ClassSumm) seem to be the key parameters for our outperforming systems.

It is important to notice that the described evaluation is not noise free. The ways ideal extracts are generated bring about a problem to our evaluation: since the generator relies on the cosine similarity measure, and this does not take into account the sentence size, there is no way to guarantee that compression rate is uniformly observed. Actually, there are ideal extracts in our reference corpus that are considerably longer than the extracts automatically generated. This poses an evaluation problem in that the comparison between both penalizes recall, whilst increasing precision.

These results are relatively similar to the ones obtained in the literature for texts in English, such as the ones of Teufel and Moens [26] (P=65% and R=44%), Kupiec et al.

---

[1] For more details about this measure, see http://duc.nist.gov/.

(P=R=42%) and Saggion and Lapalme [24] (P=20% and R=23%). Although the direct comparison between the results is not fair, due to different training, test corpora, and even language, it may indicate the general state of the art in extractive AS.

## 4  Final Remarks

Clearly, considering linguistic features and, thus, knowledge-based decisions, indicates a way of improving extractive AS. It is also worthy considering that the topmost evaluated systems are based on training, which means that, with more substantial training data, performance may be improved. Limitations usually addressed in the literature refer to the impossibility of, e.g., aggregating or generalizing information. SuPor and ClassSumm evaluations suggest that, although those procedures keep been inexistent in extractive approaches, a way of surpassing those difficulties is still to address the semantic-level through surface manipulation of text components. Another significant way of improving SuPor and ClassSumm is to make the input reference lists (e.g., stoplists and discourse markers) more expressive, by adding more terms to them. Also, substituting the language-dependent repositories that have been currently adapted (e.g., the thesaurus in SuPor) or building an argumentative tree in ClassSumm by other means may improve performance, since that will be likely to tune better the systems to Brazilian Portuguese.

After all, the common evaluation presented here made it possible to compare different systems, allowing fostering AS research especially concerning texts in Brazilian Portuguese and, more importantly, delineating future goals to pursue.

## References

1. Aires, R.V.X., Aluísio, S.M., Kuhn, D.C.e.S., Andreeta, M.L.B., Oliveira Jr., O.N.: Combining classifiers to improve part of speech tagging: A case study for Brazilian Portuguese. In: Open Discussion Track Proceedings of the 15th Brazilian Symposium on AI. (2000) 227–236
2. Barzilay, R., Elhadad, M.: Using Lexical Chains for Text Summarization. In: Advances in Automatic Text Summarization. MIT Press (1999) 111–121
3. Caldas Jr., J., Imamura, C.Y.M., Rezende, S.O.: Evaluation of a stemming algorithm for the Portuguese language (in Portuguese). In: Proceedings of the 2nd Congress of Logic Applied to Technology. Volume 2. (2001) 267–274
4. Dias-da Silva, B., Oliveira, M.F., Moraes, H.R., Paschoalino, C., Hasegawa, R., Amorin, D., Nascimento, A.C.: The Building of an Electronic thesaurus for Brazilian Portuguese (in Portuguese). In: Proceedings of the V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada. (2000) 1–11
5. Edmundson, H.P.: New methods in automatic extracting. Journal of the Association for Computing Machinery **16** (1969) 264–285
6. Kohonen, T.: Self organized formation of topologically correct feature maps. Biological Cybernetics **43** (1982) 59–69

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

8. Larocca Neto, J.: Contribution to the study of automatic text summarization techniques (in Portuguese). Master's thesis, Pontifícia Universidade Católica do Paraná (PUC-PR), Graduate Program in Applied Computer Science (2002)

9. Larocca Neto, J., Santos, A.D., Kaestner, C.A.A., Freitas, A.A.: Document clustering and text summarization. In: Proc. 4th Int. Conf. Practical Applications of Knowledge Discovery and Data Mining. (2000) 41–55

10. Larocca Neto, J., Freitas, A.A., Kaestner, C.A.A.: Automatic text summarization using a machine learning approach. In: XVI Brazilian Symp. on Artificial Intelligence. Number 2057 in Lecture Notes in Artificial Intelligence (2002) 205–215

11. Luhn, H.: The automatic creation of literature abstracts. IBM Journal of Research and Development **2** (1958) 159–165

12. Lyman, P., Varian, H.R.: How much information. Retrieved from http://www.sims.berkeley.edu/how-much-info-2003 on [01/19/2004] (2003)

13. Mani, I.: Automatic Summarization. John Benjamin's Publishing Company (2001)

14. Mani, I., Bloedorn, E.: Machine learning of generic and user-focused summarization. In: Proc. of the 15th National Conf. on Artificial Intelligence (AAAI 98). (1998) 821–826

15. Mani, I., Maybury, M.T.: Advances in Automatic Text Summarization. MIT Press (1999)

16. Martins, R.T., Hasegawa, R., Nunes, M.G.V.: Curupira: a functional parser for Portuguese (in Portuguese). NILC Tech. Report NILC-TR-02-26 (2002)

17. Módolo, M.: Supor: an environment for exploration of extractive methods for automatic text summarization for portuguese (in Portuguese). Master's thesis, Departamento de Computação, UFSCar (2003)

18. Nunes, M.G.V., Vieira, F.M.V., Zavaglia, C., Sossolete, C.R.C., Hernandez, J.: The building of a Brazilian Portuguese lexicon for supporting automatic grammar checking (in Portuguese). ICMC-USP Tech. Report 42 (1996)

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

20. Pardo, T.A.S., Rino, L.H.M., Nunes, M.G.V.: GistSumm: A summarization tool based on a new extractive method. In: 6th Workshop on Computational Processing of the Portuguese Language – Written and Spoken. Number 2721 in Lecture Notes in Artificial Intelligence, Springer (2003) 210–218

21. Pardo, T.A.S., Rino, L.H.M., Nunes, M.G.V.: NeuralSumm: A connexionist approach to automatic text summarization (in Portuguese). In: Proceedings of the IV Encontro Nacional de Inteligência Artificial. (2003)

22. Pardo, T.A.S., Rino, L.H.M., Nunes, M.G.V.: DiZer: An automatic discourse analysis proposal to brazilian portuguese (in Portuguese). In: Proc. of the I Workshop em Tecnologia da Informação e da Linguagem Humana. (2003)

23. Radev, D.R., Teufel, S., Saggion, H., Lam, W., Blitzer, J., Qi, H., Çelebi, A., Liu, D., Drabek, E.: Evaluation challenges in large-scale document summarization. In: Proc. of the 41st Annual Meeting of the Association for Computational Linguistics. (2003) 375–382

24. Saggion, H., Lapalme, G.: Generating indicative-informative summaries with sumUM. Computational Linguistics **28** (2002) 497–526

25. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management **24** (1988) 513–523

26. Teufel, S., Moens, M.: Summarizing scientific articles: Experiments with relevance and rhetorical status. Computational Linguistics **28** (2002) 409–445