Multi-Document Summarization Using Complex and Rich Features

Maria Lucía del Rosario Castro Jorge, Verônica Agostini,
Thiago Alexandre Salgueiro Pardo

Núcleo Interinstitucional de Linguística Computacional (NILC)
Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo
Avenida Trabalhador são-carlense, 400. 13566-590 - São Carlos/SP, Brazil

{mluciacj,agostini,taspardo}@icmc.usp.br

Abstract. Multi-document summarization consists in automatically producing a unique informative summary from a collection of texts on the same topic. In this paper we model the multi-document summarization task as a problem of machine learning classification where sentences from the source texts have to be classified as belonging or not to the summary. For this aim, we combine superficial (e.g., sentence position in the text) and deep linguistic features (e.g. semantic relations across documents). In particular, the linguistic features are given by CST (Cross-document Structure Theory). We conduct our experiments on a CST-annotated corpus of news texts. Results show that linguistic features help to produce a better classification model, producing state-of-the-art results.

1. Introduction

Recently, new technologies have made available a large amount of textual information in digital format. Consequently, the capability to process all this information is reduced and for this reason Multi-Document Summarization (MDS) can be a useful resource. MDS is conceived as the automatic production of a summary given a group of texts on the same topic (McKeown and Radev, 1995; Radev and McKeown, 1998). The idea of this task was imagined as an extension of a previous area called single-document summarization or simply Automatic Summarization (AS), which consists in generating a summary from one text.

In both AS and MDS, the summary should ideally contain the most relevant information of the topic that is being discussed in the source texts. As an example, Figure 1 shows a summary extracted from the work of Jorge and Pardo (2010b). This summary was automatically produced from three texts telling the victory of Brazil at the volleyball world league. These texts are part of CSTNews corpus (Aleixo and Pardo, 2008; Maziero et al., 2010), which contains news texts written in Brazilian Portuguese. The summary was translated from Portuguese.

> *The Brazilian volleyball team won on Friday the seventh consecutive victory in the World League, defeating Finland by 3 sets to 0 - partials of 25/17, 25/22 and 25/21 - in a match in the Tampere city, Finland. The first set remained balanced until the middle, when André Heller went to serve. In the last part, Finland again paired the game with Brazil, but after a sequence of Brazilians points Finland failed to respond and lost by 25 to 21. The Brazilian team won five times the World League in 1993, 2001, 2003, 2004 and 2005.*

Figure 1. Multi-document Summarization example (Jorge and Pardo, 2010b)

Moreover, MDS should not only focus on the extraction of relevant information, but also deal with challenges, such as redundancy, complementary and contradictory information, different writing styles and varied referential expressions. These are factors that are more likely to happen when there are various sources of information.

There are two ways of approaching MDS as described by Mani and Maybury (1999). The first one is the superficial approach in which statistical or little linguistic information is used to build the summary. Classical examples of this approach include methods based on word counting. The second approach is deep and is characterized by the usage of deep linguistic knowledge such as grammars, semantic or discourse information. The deep approach is said to produce summaries of higher quality in terms of information, coherence and cohesion, but it demands various high-cost resources. On the other hand, superficial approach requires low-cost processing, but summaries produced by models under this approach tend to have lower quality. In summarization, there are various researches for both approaches, varying from simple sentence selection based on word frequency (Luhn, 1958; Edmundson, 1969) to more complex methods that combine machine learning techniques and textual features (Kupiec et al., 1995), or methods that include more sophisticated linguistic knowledge such as CST (Cross-document Structure Theory) (Radev, 2000; Zhang et al. 2002; Otterbacher et al. 2002; Afantenos et al. 2004; Jorge and Pardo 2009, 2010a, 2010b).

According to Mani and Maybury, MDS should also take three phases: the first one is analysis, in which an intern representation of the texts is created; the second phase is called transformation, in which relevant content is selected given the representation in the previous phase; finally, the third phase is synthesis, where all the selected content is expressed in natural language. Given this context, this work is focused on the second MDS phase, where content selection is done. Particularly, we present a machine learning approach for content selection by representing this task as a supervised classification problem. For this aim, sentences are treated as instances for classification, in which classes indicate whether a sentence should be included in the summary or not.

We combine simple features, such as sentence position and sentence size, with more sophisticated linguistic features, such as semantic relations between sentences from different texts. In our case, these semantic relations are given by the CST model. CST is a multi-document model that provides a set of 24 semantic relations that aim to be applied across documents on the same topic. Our proposal is a novel approach for MDS, since it is the first work that treats MDS as a classification problem by including deep features. It is important to point out that, at the moment, CST information has been shown to be the most sophisticated information for MDS of texts written in Brazilian Portuguese (Jorge and Pardo, 2009; 2010b).

In order to manage the CST information, sentences and their relations across texts are organized as graphs, were sentences are represented as nodes and CST relations as edges. Having this representation, it is also easy to extract mathematical measures from the graph and treat them like features. A well studied way to extract these mathematical measures is through the concept of complex networks (Costa et al., 2007). In this work we will use the clustering coefficient, which has been previously used by Antiqueira and Nunes (2010) for single-document summarization and showed to be good for sentence extraction. The main hypothesis in this work is that deep linguistic knowledge features combined with simple and complex features will help to predict more accurately sentence classes.

Experiments are conducted over CSTNews corpus, which is a corpus of Brazilian Portuguese news texts already annotated with CST. The corpus also provides human summaries, which will determine the correspondent classes of the sentences in the original texts of the corpus, as well as be the basis for evaluation. Results show a good performance using traditional machine learning classifiers such as Decision Trees, Naïve Bayes, Support Vector Machines and One Rule, producing state-of-the-art results and demonstrating the importance of linguistic features.

This paper is organized as follows: in Section 2, we will make a review on the main related work for MDS and CST; in Section 3, the methodology used to drive the experiments will be described; in Section 4, the evaluation method will be described; in Section 5, results will be discussed and; in Section 6, some final remarks will be made.

2. Multi-Document Summarization

One of the first researches on MDS was the one proposed by McKeown and Radev (1995, 1998) where the authors represent in template format (like an attribute-value table) the information contained in texts. They relate those templates with semantic relations like the ones in CST, and then their system activates operators that combine information according to relations and attributes grades of importance to that information. The most important information is selected for the summary.

Mani and Bloedorn (1997) proposed a method for modeling the multi-document problem consisting in a word graph construction from a set of texts. The authors model, through the graph, similarities, differences and contradictions in texts, and then they walk on the graph to find relevant information according to the topic.

Another important research was introduced by Radev et al. (2000, 2001), who proposed the system MEAD, which creates a rank of sentences by giving a grade according to three basic features: lexical distance to the first sentence of the documents, lexical distance to the title of the documents and lexical distance to the centroid of the documents (which is the sentence or set of sentences that better represent the focus of the topic of the texts). Sentences with the best grades are candidates to compose the final summary.

Radev (2000), besides proposing CST model, also proposed a 4-stage summarization process: in the first stage, clustering of similar texts is done; in the second stage, texts may be internally structured (which can be a syntactic, semantic or discourse structure); in the third stage, CST relations are established among texts units; finally, in the forth stage, the content for the summary is selected by exploring CST relations.

After CST was proposed, some important works used the model for MDS. Zhang et al. (2002) proposed an improvement of MEAD system by re-ranking sentences according to the number of CST relations. Otterbacher et al. (2002) used CST relations to improve cohesion in summaries. Afantenos et al. (2004) proposed a new classification of CST relations considering two main categories: synchronic and diachronic relations. Synchronic relations describe the same event discussed in various sources, and diachronic relations describe the evolution of an event through time. The authors also proposed a summarization method based on pre-defined templates and ontologies. Jorge and Pardo (2009, 2010a, 2010b) explored CST knowledge by applying operators for content selection based on user preferences. Operators apply rules that map user preferences to CST relations in order to build the summary. Results of this work showed to be good for Brazilian Portuguese texts, in terms of informativity, redundancy reduction, cohesion and coherence. Maziero et al. (2010) proposed a refinement of CST relations by reducing the number of relations to 14. The authors also proposed a classification of these 14 relations, which we reproduce in Figure 2. This classification has two main categories: Content and Presentation/form. Content category involves all the relations that refer mainly to the information within two text units. On the other hand, the Presentation/Form category includes relations that refer mainly to the way text units are written. Content Category is divided in three subcategories: redundancy, complement and contradiction, while Presentation/Form category is divided in two subcategories: style and authorship.

For MDS there have not been significant researches based on machine learning using deep linguistic features. For single-document summarization there are some important works that use machine learning to create automatic models for summarization. One interesting work in this line is proposed by Kupiec et al. (1995), where the authors treat the content selection task as a problem of statistical classification. Given texts and their corresponding summaries, a classification function will determine which sentences have more probability to be in the summary. The authors consider features such as sentence size, most frequent words, paragraph position, uppercase words, and fixed phrase attributes. A Naïve Bayes algorithm is used for classification. Results showed that fixed phrase and paragraph position attributes give a better performance rather than word frequency based attributes.

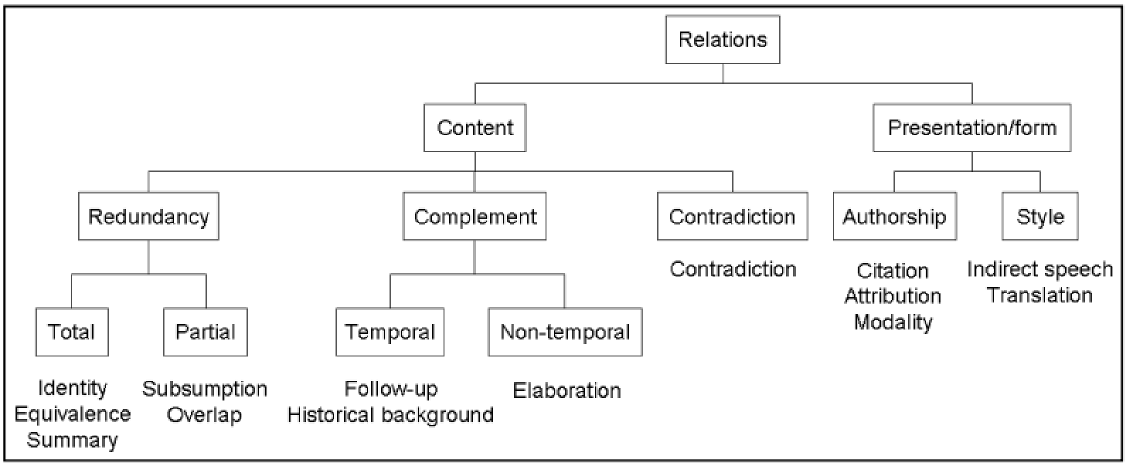

Figure 2. Maziero et al. (2010) classification of CST relations

Mani and Bloedorn (1998) proposed a learning method for generic and user-focused summaries. They used three types of features: location, thematic and cohesion. The authors use C4.5 (Quinlan, 1992) and AQ15c (Wnek et al., 1995) methods. The results showed that rules learned for user focused summaries use more keyword specific features and, for generic summaries, rules tend to use more location features. Chuang and Yang (2000) used two types of features: structured and non-structured features. Structured features correspond to rhetorical relations (Mann and Thompson, 1987) and non-structured features correspond to word frequencies, paragraph number, etc. The authors use decision trees, Naïve Bayes classifier and DistAl neural network (Yang et al., 1999). Results showed that the systems built with those learning algorithms outperformed the commercial Microsoft Word summarizer.

In the next section, we describe the methodology used in this work.

3. Methodology

As it has been mentioned before, the experiments of this work were conducted over the CSTNews corpus, which is composed of 50 news text clusters, and each cluster contains in average 3 documents discussing a common topic. Each cluster also provides the correspondent CST annotation of the sentences and a summary elaborated by humans. For the moment the annotation has been manually done. At the moment, an investigation for an automatic way of detecting CST relations is being developed (Maziero et al., 2010).

Our methodology consists of four stages: graph construction, feature selection, classification, and summary building, which will be described next.

3.1. Graph Construction

In this initial step we construct a graph containing all the CST information, where sentences are represented by nodes and relations are represented by edges. In Figure 3, we show a small example of a hypothetical graph containing CST information. This structure will help us to extract some important features as it will be detailed in Section 3.2.

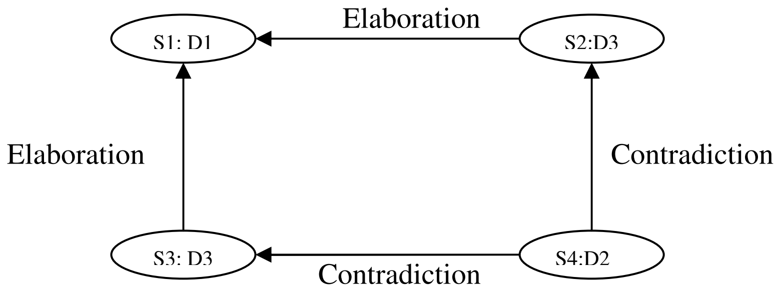

Figure 3. Illustration of a graph containing CST relations

In this hypothetical example, we see that sentence four from document two of the cluster, denoted as S4:D2, contradicts sentences two and three from document three (S2:D3 and S3:D3, respectively); and these two sentences elaborate sentence one from document one (S1:D1).

3.2. Feature Selection

We consider 9 features to describe the sentences of the corpus. We are considering sentences as the segments to be analyzed because the corpus CSTNews is already segmented and annotated at this level.

We divide features in two categories: superficial and deep features. We describe them below.

3.2.1. Superficial Features

This type of feature corresponds to the characteristics that refer to the structure and statistics rather than the semantics of the texts. We list and describe 2 superficial features: sentence size and sentence position:

- Sentence size: describes the size of a sentence in terms of the number of words it contains, excluding the stopwords. This attribute was normalized by diving all the size values of each sentence with the highest size value in the cluster. The final value of this feature results in a number between 0 and 1.

- Sentence position: refers to the position of the sentence in a particular text. This feature can assume three possible values: BEGIN, MIDDLE, and END. BEGIN value corresponds to the first sentence of any text; END value corresponds to the last sentence of any text; and MIDDLE value corresponds to the remaining sentences between BEGIN and END.

3.2.2. Deep Features

Deep features refer to semantic and discourse characteristics of the texts. In our case this characteristics are given by the information provided by CST. We consider seven deep features, three of them are metrics derived from the graph constructed in the first stage of the process; the other four features correspond to the type of relation according to Maziero et al. (2010) classification. All seven features are listed below:

- Centroid: this feature indicates whether a sentence is the most representative of the content of a text or not. According to this idea, this feature can assume two values: yes or no. To measure how representative a sentence can be, we calculate the grade of a node (sentence) in the graph constructed in stage 1. The grade is given by the number of edges (CST relations) that connects the given sentence (node). The sentence with the highest number of edges is considered the centroid.

- Similarity to centroid: once we know which sentence of the text is the centroid, we calculate how similar to the centroid the other sentences are. This similarity is measured by lexical distance, which is calculated by word overlap function. We consider this overlap function as the number of lexical matches between two sentences divided by the sum of the words in the two sentences, which results in a number between 0 and 1.

- Clustering coefficient (CC), which is a well known metric in the theory of complex networks that measures the level that nodes tend to cluster together. As an example, in Figure 4 it can be observed a graph that presents a highly connected group of nodes (nodes 1 to 5).

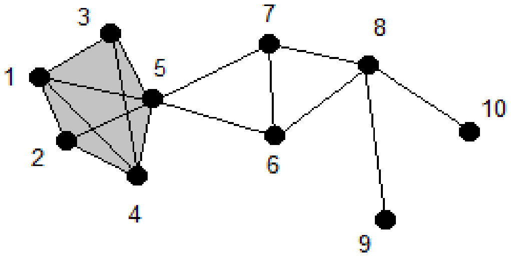

Figure 4. Example of cluster in a graph

In MDS, a graph that models CST information may have clusters in its topology and this can express highly elaborated topics in the texts, which we think are more likely to be in the summary. In other words, CC may be a good feature that can help to predict if a sentence should be or not in the final summary. We choose this measure based on the experiments made by Antiqueira and Nunes (2010) that show that CC is a good measure for AS using complex networks. CC is calculated using the next formula:

$$\text{CC}= \frac{3 \times \# triangles}{\# connected\ triples}$$

In this formula, triples refer to three nodes that are either connected by two edges or three edges. Particularly, triangles are defined as three nodes connected by three edges, while simple triples connect three nodes with only two edges. For example, considering the graph represented in Figure 4: it can be observed that nodes 8, 9, and 10 compose a simple triple, since 9 and 10 are not connected by any edge, but nodes 5, 6 and 7 compose a triangle because the three nodes are connected by an edge.

- Redundancy type: in this category are included all the relations that refer to redundant information, considering that redundancy can be partial or total. These relationships are: Identity, Equivalence, Overlap, Summary and Subsumption. Given a sentence in a CST graph, the value of the redundancy feature indicates the number of relations of this type that the sentence has.

- Complement type: in this category are included all the relations that refer to complementary information. This category includes: Historical background, Elaboration and Follow-up. Similarly to Redundancy feature, the value for complement feature indicates the number of relations of this type that the sentence has.

- Contradiction type: it only includes the contradiction relation. The value for this feature is calculated similarly to the redundancy and complement features.

- Authorship type: this category includes all relations that refer to the writing style and authorship information rather than the content of the text. The relations that are part of this category are: Citation, Attribution, Modality, Indirect-speech and Translation. The value for this feature is calculated the same way as the other CST features.

The value for the last 4 features is normalized by dividing the feature with the total number of relations within the cluster. As an example of how these features are calculated, consider a sentence S that is connected by a Subsumption relation and an Attribute relation in the graph: this sentence will have 1 relation of the redundancy category and 1 of the authorship category. Supposing that these are the only relations in the cluster, the feature vector of sentence S is showed in Figure 5. In this example, features A, R, Comp and Contr correspond to Authorship/style, Redundancy, Complement and Contradiction categories, respectively.

Position	Size	Centroid	CC	Similarity to centroid	A	R	Comp	Contr
…	…	…	…	…	0.5	0.5	0	0

Figure 5. Example of feature vector of Sentence S

The idea behind these features is also that, after training, we may know which types of CST relations help to predict more accurately the classes of the sentences, i.e., which types of relations are more likely to include sentences in the final summary.

In this initial approach, we only considered these 9 features because our goal is to focus on how CST information, combined with superficial information, helps to predict sentence classes. Of course other features might be included, considering, for instance, syntactic, semantic or even ontology-based information, but these will be explored in future works.

3.3. Classification

Having the set of features, we need to determine the correspondent class of each sentence of our corpus. Two possible classes can be assigned: 0 or 1. Sentences classified as 1 represent the ones that should be included in the final summary and sentences classified as 0 represent the ones that should not be included in the final summary. Since for each cluster in the corpus we have the correspondent human summary, we can determine which sentences are in that summary. Our limitation here is that human summaries (abstracts) are not necessarily composed by the same exact sentences of the original texts. In order to solve this limitation, we consider as belonging to class 1 all the sentences that have at least 70% of lexical similarity with any sentence of the human summary; otherwise, class 0 is assumed. This is an initial approach, but for future works we plan to evaluate which threshold is the most suitable when comparing sentences versus summary sentences.

To train our classification models, we used 4 methods: Decision Trees (particularly, algorithm J48), Naïve Bayes, Support Vector Machine (SVM) and One Rule. We use WEKA (Witten and Frank, 2005) for running all these methods.

Since our database is unbalanced (because the number of sentences of class 1 is a lot lower than sentences of class 0) we balance it manually by duplicating instances from the minor class of the database. The data set is also divided in two subsets: training set and test set. The training set contains 80% of the instances of the data, while the testing set contains 20% of it.

3.4. Summary Building

After we create our models with each classification algorithm, we tested those models using the test set. For each model, we select the instances that were classified with class “1”. Before building the summary, instances of class 1 are ranked according to the probability of being actually from class 1. Sentences are then included in the summary according to the order in which they are ranked until the size of the summary reaches its maximum size considering the specified compression rate. In this work we consider a 70% compression rate in relation to the longest text in the cluster (i.e., the summary may have up to 30% of the number of words of the longest text). We use this compression rate because is the same compression rate that was used to build the human summaries.

We also consider that sometimes there will not be enough instances classified as “1” to reach the specified summary size. In this case, we start including instances that were classified as “0” and had lower probability of being from that class (until de desired size is reached).

One important step that is taken into account is the redundancy treatment. As mentioned in Section 1, redundancy is something we have to deal with when we are working with several texts discussing the same topic. For this, we use the information provided by CST to eliminate redundancy from the final summary, by discarding sentences that have relations of the redundancy type. For example, if the relation between two sentences is Identity, one of them will be eliminated; if the relation is Equivalence, we will eliminate the longest sentence (considering the number of words of the sentence); if the relation is Subsumption, we eliminate the sentence that is subsumed.

One relation that is part of the redundancy type and is not treated here is the Overlap relation. We still do not include it in the process because eliminating redundancy in this case requires extra steps (sentence fusion, for instance) that we are not using at the moment, but certainly we will use in future works.

In the next section we will describe the evaluation methodology we use in this work.

4. Evaluation

Two types of evaluation were carried out in this work. The first evaluation corresponds to the traditional machine learning evaluation of accuracy: Precision, Recall, F-measure and ROC area. We used 10-fold cross-validation for this aim.

In the second type of evaluation, we wanted to measure how informative were the automatic summaries built from our models. For this, we used ROUGE (Lin and Hovy, 2003) which is an automatic measure that compares automatic and human summaries by considering the n-grams they have in common. There are various measures of ROUGE. In this work we will use two of them: ROUGE-1 and ROUGE-L. ROUGE-1 compares unigrams in both summaries; ROUGE-L compares the longest common subsequences between the summaries. Results for these measures are given in terms of Precision, Recall and F-measure. Note that these measures are different from the traditional machine learning ones. ROUGE considers Precision (P) and Recall (R) as shown in the formulas below:

$$P= \frac{\text{Number of common n-grams among human and automatic summaries}}{\text{Number of n-grams of the automatic summary}}$$

$$R= \frac{\text{Number of common n-grams among human and automatic summaries}}{\text{Number of n-grams of the human summary}}$$

In the next section the results will be discussed.

5. Experiments and Results

As mentioned before, we run our experiments considering 4 well known classification methods: Decision Trees (J48), Naïve Bayes (NB), SVM and One Rule (OneR). In Table 1 it is shown the results in terms of traditional Precision, Recall, F-Measure and ROC area for each class. In Table 2 it is shown more information from the confusion matrix for each class, such as True Positive rate (TP), False Positive Rate (FP), Correctly Classified Instances Rate (CCIR).

Table 1. Results in terms traditional Precision, Recall, F-measure and ROC area

	Class 1 (sentence should be in the summary)				Class 0 (sentence ignored)			
	Precision	Recall	F-measure	ROC	Precision	Recall	F-measure	ROC
J48	0.801	0.910	0.852	0.891	0.896	0.774	0.831	0.891
NB	0.712	0.306	0.428	0.687	0.558	0.877	0.682	0.687
SVM	0.694	0.538	0.606	0.651	0.623	0.764	0.686	0.651
OneR	0.690	0.749	0.718	0.706	0.726	0.664	0.693	0.706

Table 2. Confusion matrix information

	Class 1		Class 0		CCIR
	TP	FP	TP	FP	
J48	0.910	0.226	0.774	0.090	0.842
NB	0.306	0.123	0.877	0.694	0.591
SVM	0.538	0.236	0.764	0.462	0.650
OneR	0.749	0.336	0.664	0.251	0.706

It can be observed from the results of Table 1 and Table 2 that the best performance was achieved by algorithm J48, in terms of Precision, Recall, F-measure and ROC area. J48 has also the highest rate of correctly classified instances. We observed that, for this algorithm, deep features are considered good attributes for the tree division. Particularly, features of the type redundancy and complement are the ones that obtain more information gain values. To evaluate this, we applied an attribute evaluator from Weka, and the results showed that the three best attributes are: redundancy type, sentence position and complement type. This consists in important evidence that deep features such as relation type provide important information for the classification of sentences. This also corresponds to the intuition behind MDS that redundant information (that is repeated across texts) tends to be important. OneR also showed a good performance in terms of F-measure, ROC area and mean absolute error. This algorithm used the sentence size feature to build the rules.

We also wanted to see how a rule-based algorithm selects more than one feature to construct the rules. For this, we built a model using Prism algorithm (also part of WEKA), though we did not evaluate it. In this model, we observed that rules involving higher values for deep features require less feature combinations to determine the sentence class. On the other hand, rules involving lower values or superficial features require more feature combinations to complete the rule. For example, in Figure 6, the first learned rule expresses that, if a sentence has a complement feature value between 0.090 and 1, and a clustering coefficient value between 0.049 and 0.069, and a redundancy feature between 0.107 and 1, then the class for that sentence is 1. In other words, if the sentence has a considerable number of complement relations, and it is part of a well connected cluster in the graph, and the information of the sentence is repeated across various texts, then the sentence should be included in the summary. In the case of the second learned rule, the value of the complement feature is evaluated in the interval between 0 and 0.007, which is lower than 0.090 and 1. In other words, these rules express that if a sentence has higher values of deep features (like complement or redundancy), then the chance to be in class 1 is high, but if they have lower values for these features, the rule will have to include other conditions in order to determine the correct class.

SVM, which also has a good performance, gives higher weight to complement type and redundancy type features. For Naïve Bayes classifier, superficial features are more predictive, but this classifier has the lowest rate for correctly classified instances.

Learned rule 1:
If Complement feature = (0.090-1]
and Clustering coefficient = (0.049-0.069]
and Redundancy feature = (0.107-1] **then class 1**

Learned rule 2:
and Sentence size = (0-0.176]
and Similarity to centroid = (0-0.009]
and Redundancy feature = (0-0.007]
and Sentence position = MEDIO
and Centroid = NO
and Complement type = (0-0.007]
and Authorship type = (0-0.007] **then class 1**

Figure 6. Example of rules in Prism

Considering that J48 achieved the best performance, we used that model to build automatic summaries for 10 clusters of the corpus, which correspond to the 20% of the data that we selected as test set. These summaries have been evaluated using ROUGE, as described in Section 4. These results are compared with the ones obtained by Jorge and Pardo (2010b) and Radev et al. (2001) methods, with their generic CST operator and MEAD system, respectively. All these results are shown in Table 3. We show only ROUGE-1 and ROUGE-L results, which are among the most used measures.

Table 3. ROUGE results

	ROUGE-1			**ROUGE-L**		
	Precision	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
J48	0.537	0.550	0.542	0.509	0.521	0.514
Generic operator	0.572	0.523	0.543	0.538	0.492	0.512
MEAD	0.554	0.239	0.369	0.540	0.230	0.354

We can observe that, in terms of F-measure, model J48 and the generic CST operator proposed by Jorge and Pardo have a good performance when compared to MEAD. This confirms previous results (Jorge and Pardo, 2009, 2010b) that show that CST information helps to produce better summaries in terms of information quality. It is also interesting to see that some superficial features, like sentence size, also show to be important and to complement CST information.

To measure the confidence degree of these results, we used ANOVA test. The statistical test showed that the results for Precision, Recall and F-measure had a 95% confidence degree.

In future works we intend to investigate more deep features that may help to improve even more the performance of our model. These features may include syntactic and/or ontological information or even rhetorical information such as the proposed by Mann and Thompson (1987).

6. Final Remarks

This work presented a machine learning approach for multi-document summarization using deep features for the classification task. Results showed that these features are good predictors for sentence classes. At the moment the only deep features we are using are the ones that come from the information provided by CST model. For future work we intend to explore more linguistic features, such as syntactic, ontological or rhetorical information.

Acknowledgements

The authors are grateful to FAPESP, CAPES and CNPq for supporting this work.

References

Afantenos, S.D.; Doura, I.; Kapellou, E.; Karkaletsis, V. (2004). Exploiting Cross-Document Relations for Multi-document Evolving Summarization. In the *Proceedings of SETN*, pp. 410-419.

Aleixo, P. and Pardo, T.A.S. (2008). *CSTNews: Um Córpus de Textos Journalísticos Anotados segundo a Teoria Discursiva CST (Cross-Document Structure Theory).* Série de Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, no. 326. São Carlos-SP.

Antiqueira, L. and Nunes, M.G.V. (2010). Complex Networks and Extractive Summarization. In the *Extended Activities Proceedings of the 9th International Conference on Computational Processing of the Portuguese Language* – PROPOR. Porto Alegre/RS, Brazil.

Chuang W.T. and Yang J. (2000). Extracting sentence segments for text summarization: a machine learning approach. In the *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 152-159. Athens, Greece.

Costa, L.F.; Rodrigues, F.A.; Travieso, G.; Boas, P.R.V. (2007). Characterization of complex networks: A survey of measurements. *Advances in Physics*, Vol. 56, pp. 167-242.

Edmundson, H. P. (1969). New Methods in automatic extracting. *Journal of the ACM*, Vol. 16, pp. 264-285.

Jorge, M.L.C. and Pardo, T.A.S. (2009). Content Selection Operators for Multidocument Summarization based on Cross-document Structure Theory. In the *Proceedings of the 7th Brazilian Symposium in Information and Human Language Technology* – STIL, pp. 1-8. São Carlos/SP, Brazil.

Jorge, M.L.C. and Pardo, T.A.S. (2010a). Formalizing CST-based Content Selection Operations. In the *Proceedings of the International Conference on Computational Processing of Portuguese Language-* PROPOR. Porto Alegre/RS, Brazil.

Jorge, M.L.C. and Pardo, T.A.S. (2010b). Experiments with CST-based Multidocument Summarization. In the *Proceedings of the ACL Workshop TextGraphs-5: Graph-based Methods for Natural Language Processing,* pp. 74-82. Uppsala/Sweden.

Kupiec, J.; Pedersen, J.; Chen, F. (1995). A trainable document summarizer. In the *Proceedings of the 18th ACMSIGIR Conference on Research & Development in Information Retrieval*, pp. 68-73.

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

Luhn, H.P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development,* Vol. 2, pp. 159-165.

Mani, I. and Bloedorn, E. (1997). Multi-document summarization by graph search and matching. In the *Proceedings of the 14th National Conference on Artificial Intelligence* – AAAI, pp. 622-628.

Mani, I. and Bloedorn, E. (1998). Machine Learning of Generic and User-Focused Summarization. In the *Proceedings of the Fifteenth National Conference on Artificial Intelligence* – AAAI, pp. 821-826.

Mani, I. and Maybury, M.T. (1999). *Advances in automatic text summarization.* MIT Press, Cambridge, MA.

Mann, W.C. and Thompson, S.A. (1987). *Rhetorical Structure Theory: A Theory of Text Organization.* Technical Report ISI/RS-87-190.

Maziero, E.G.; Jorge, M.L.C.; Pardo, T.A.S. (2010). Identifying Multidocument Relations. In the *Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science-* NLPCS, pp.60-69. Funchal/Madeira, Portugal.

McKeown, K. and Radev, D.R. (1995). Generating summaries of multiple news articles. In the *Proceedings of the 18th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 74-82. Seattle, WA.

Otterbacher, J.C.; Radev, D.R.; Luo, A. (2002). Revisions that improve cohesion in multi-document summaries: a preliminary study. In the *Proceedings of the Workshop on Automatic Summarization,* pp 27-36. Philadelphia.

Quinlan, J.R. (1992). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco/CA, USA.

Radev, D.R. (2000). A common theory of information fusion from multiple text sources, step one: Cross-document structure. In the *Proceedings of the 1st ACL SIGDIAL Workshop on Discourse and Dialogue*. Hong Kong.

Radev, D.R. and McKeown, K. (1998). Generating natural language summaries from multiple on-line sources. *Computational Linguistics*, Vol. 24, N. 3, pp. 469-500.

Radev, D.R.; Jing, H.; Budzikowska, M. (2000). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation and user studies. In the *Proceedings of the ANLP/NAACL Workshop*, pp. 21-29.

Radev, D.R.; Blair-Goldensohn, S.; Zhang, Z. (2001). Experiments in single and multi-document summarization using MEAD. In the *Proceedings of the First Document Understanding Conference*. New Orleans, LA.

Witten, I.H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wnek, J. (1995). *DIAV 2.0. User Manual: Specification and Guide through the Diagrammatic Visualization System*. Reports of the Machine Learning and Inference Laboratory, George Mason University.

Yang, J.; Parekh, R.; Honavar V. (1999). DistAl: An inter-pattern distance-based constructive learning algorithm. *Intelligent Data Analysis*, Vol. 3, pp. 55-73.

Zhang, Z.; Goldenshon, S.B.; Radev, D.R. 2002. Towards CST-Enhanced Sumarization. In the *Proceedings of the 18th National Conference on Artificial Intelligence*. Edmonton/Canada.