Procesamiento del Lenguaje Natural, Revista nº 60, marzo de 2018, pp. 29-36 recibido 05-11-2017 revisado 07-01-2018 aceptado 02-02-2018

A Supervised Central Unit Detector for Spanish

Un detector de la unidad central para textos en castellano

Kepa Bengoetxea and Mikel Iruskieta
IXA Group. University of the Basque Country
{kepa.bengoetxea,mikel.iruskieta}@ehu.eus

Resumen: En este artículo presentamos el primer detector de la Unidad Central (CU) de resúmenes científicos en castellano basado en técnicas de aprendizaje automático. Para ello, nos hemos basado en la anotación del *Spanish RST Treebank* anotado bajo la Teoría de la Estructura Retórica o *Rhetorical Structure Theory* (RST). El método empleado para detectar la unidad central es el modelo de bolsa de palabras utilizando clasificadores como Naive Bayes y SVM. Finalmente, evaluamos el rendimiento de los clasificadores y hemos creado el detector de CUs usando el mejor clasificador.

Palabras clave: Unidad central, RST, clasificación, minería de datos, Naive Bayes, SVM

Abstract: In this paper we present the first automatic detector of the Central Unit (CU) for Spanish scientific abstracts based on machine learning techniques. To do so, learning and evaluation data was extracted from the *RST Spanish Treebank* annotated under the *Rhetorical Structure Theory* (RST). We use a bag-of-words model based on Naive Bayes and SVM classifiers to detect the central units of a text. Finaly, we evaluate the performance of the classifiers and choose the best to create an automatic CU detector.

Keywords: Central unit, RST, classification, data mining, Naive Bayes, SVM

1 *Introduction*

Knowing what is the most important sentence of a text and the intention in which this was uttered is a crucial task for language learners to understand a text.

Following Iruskieta, Diaz de Ilarraza, and Lersundi (2014) the central unit (CU) is an elementary discourse unit (EDU) and the most salient text-span of a rhetorical structure. Rhetorical structures or the RST diagrams are represented as trees (RS-trees) and there is at least one text-span[1] that is not modified by any other EDU through any mononuclear relation. On the contrary, this text span functions as the central node of the tree.

Determining first the most important segment of a discourse in a text is crucial also to annotate the rhetorical structure of a text (Iruskieta, de Ilarraza, and Lersundi, 2014), but also for some advanced NLP tasks such as sentiment analysis, summarization tasks and question answering, among others.

Automatic classification is a learning process, during which a program recognizes the features that distinguish each category from others and constructs a classifier when given a set of training examples with class labels. Application of this approach to the CUs can help in automatic detection on the basis of similarity of their content. In this research we classify CUs using the bag of words model. Algorithms used in classification are Naive Bayes (NB) (McCallum, Nigam, and others, 1998) and Support Vector Machine (SVM) (Cortes and Vapnik, 1995) that were successfully used in previous researches in text classification (Schneider, 2005).

Some CU's detectors were developed for Basque (Bengoetxea, Atutxa, and Iruskieta, 2017) and for Brazilian Portuguese (BP),[2] but there is no tool to detect the CU for Spanish.

To fulfill this gap, the main aim of this paper is to built an automatic Central Unit detector for Spanish scientific abstracts.

[1] If the relation at the top is a multinuclear one, there are more than one EDU functioning as CU.

[2] The demos of these two tools can be tested at http://ixa2.si.ehu.es/CU-detector/ for Basque (reliability of 0.57 F1) and http://ixa2.si.ehu.es/clarink/tools/BP-CU-detector/ for BP (reliability of 0.657 F1).

ISSN 1135-5948. DOI 10.26342/2018-60-3

Although this tool can be used in different approaches, it was developed under Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) that is a descriptive, language-independent theory of the organization of texts, which characterizes the text structure primarily in terms of the hierarchical relations that hold between discourse segments (EDUs).

Following Iruskieta et al. (2013), the rhetorical analysis of a text includes three phases: *i*) text segmentation (EDUs), *ii*) CU annotation and *iii*) description of relations between EDUs and groups of EDUs linked to the CU, building a hierarchical tree (RS-tree).

The results shows that we can build an entirely automatic CU detector with a good performance if we depart from an annotated RST corpus.

2 Related Work

Using different techniques some CU detectors were developed following the findings by Iruskieta, Diaz de Ilarraza, and Lersundi (2014) that show the importance of annotating the CU before rhetorical relations: *i*) Rule based detectors use features that were design by linguists for Basque[3] and Brazilian Portuguese (Bengoetxea, Atutxa, and Iruskieta, 2017; Iruskieta, Antonio, and Labaka, 2016). *ii*) Machine learning techniques using features developed by linguists for Basque (Bengoetxea, Atutxa, and Iruskieta, 2017). In both approaches, evaluation measures are based in annotated data, where the CU was considered in annotation guidelines.

In these works, authors found that annotating or detecting the CUs is a genre and domain oriented classification task. Some features which work very well for scientific abstracts genre, do not work for argumentative answer texts genre, or vice versa. Therefore, developing a general good CU detector is a complicated task, because following these approaches a linguist is needed to annotate the corpus and to extract the features manually for each genre or domain (and language).

The work presented here is different from the previous works, because of these two reasons: 1) the features to detect the CU are extracted automatically and 2) the corpus, the RST Spanish Treebank, employed in this work was annotated with rhetorical relations, following typical two step annotation methodology: *i*) EDU segmentation and *ii*) rhetorical relation labeling (da Cunha, Torres-Moreno, and Sierra, 2011). Therefore, it was annotated without taking into account the CU constraints in the annotation guidelines.[4]

The method employed in this work will be useful to detect the CU in other languages, genres and domains with less effort, if RST annotated data is available. The CU detector can be useful in several NLP tasks, such as sentiment analysis (to identify the most important evaluative sentence (Alkorta et al., 2017)), annotation of the rhetorical RS-trees (Iruskieta, de Ilarraza, and Lersundi, 2014) or to improve some parsers or prototypes (da Cunha et al., 2012).

3 Methodology

As we noted previously, there is not an annotated corpus with CUs for Spanish, but we extract the root of the rhetorical trees and label as CU. So, in order to build the CU detector we follow the subsequent phases.

3.1 Source for corpus compilation

The corpus we have used for such task is the RST Spanish Treebank (da Cunha, Torres-Moreno, and Sierra, 2011) which is the first corpus annotated with rhetorical relations for Spanish. The corpus is annotated with specialized texts of 9 domains: *i*) Astrophysics, *ii*) Earthquake Engineering, *iii*) Economy, *iv*) Law, *v*) Linguistics, *vi*) Mathematics, *vii*) Medicine, *viii*) Psychology and *ix*) Sexuality.

3.2 Selected corpus

To ensure the compilation of the corpus we check if every text was organized as follows: *i*) If all the text has a title at the beginning of the document. *ii*) If the text was long enough (most of the texts of the same domain has more than 4 EDUs). *iii*) If the extracted CU from the RS-tree is reliable.

We found that a lot of texts of different domains do not fulfill these constraints, so we selected the best two domains that fulfill these constraints: *i*) Psychology and *ii*) Linguistics.

[3] Basque corpus is composed with different domains, in the same genre.

[4] In the studies previously mentioned, the CU constraints were considered in the annotation process.

In one of these domains, we detect that the linguistic texts lack the title (4 of them) and some CU (7 text of 45) were wrongly annotated (and wrongly extracted),[5] once we compared with our CU annotation guidelines (Iruskieta, de Ilarraza, and Lersundi, 2014).[6]

Therefore, when an inconsistency in the annotation was found, the entry was fully examined, the title was added and the extracted CU was changed in our database.

After this process, the corpus description used in this paper is presented in Table 1 describing the two domains (Dom.): Psychology (PS) and Linguistics (LI), texts (T), words (W), Elementary Discourse Units (EDU) and Central Units (CU).

Dom.	T	W	EDU	CU
PS	28	4409	274	36
LI	45	11176	599	51
Total	73	15585	873	87

Table 1: Corpus description

The gold standard we created contains 873 EDUs and 73 texts, each with its CU.

The amount of texts of this study is smaller than previously used for similar tasks (Bengoetxea, Atutxa, and Iruskieta, 2017; Iruskieta, Antonio, and Labaka, 2016; Burstein et al., 2001).

3.3 Preprocessing

The steps to preprocess the data are the following:

i) Data. We extract EDUs and CUs from the annotated Spanish RST Treebank (da Cunha, Torres-Moreno, and Sierra, 2011). The gold standard segmented corpus was annotated automatically with morphosyntactic information using FreeLing (Carreras et al., 2004).

ii) Database. The database was created with the gold standard files.

iii) Data-sets. This corpus was divided into 2 non-overlapping datasets as we show in Table 2: 60 texts as a training dataset (Train) and 13 texts as test dataset (Test). So, we used 20% of the data for testing and rest of the 80% for training. To estimate the performance of our systems and to select the best classifier, we use a 10-fold cross-validation procedure: the 60 texts of the train dataset were partitioned randomly into 10 groups and we train 10 times on 9/10 of the labeled data and we evaluate the performance on the other 1/10 of the data.
Table 2 reports some information about the 2 non-overlapping datasets, measures (T for texts, EDUs, CUs) and difficulty (Diff.), multiple CUs (M) and texts where the CU is in the first EDU (F).[7]

Set	T	EDU	CU	Diff.	M	F
Train	60	621	69	0.111	8	25
Test	13	183	14	0.076	1	6
Total	73	804	83			

Table 2: Data-set information

The task's difficulty to find the CU has been calculated as follows: $Difficulty = \frac{CUs}{EDUs}$ where the nearer it is from 1 the easier it is to determine the CU.
Test dataset is more difficult, because difficulty is farther from 1 to determine the CU. While the proportion of multiple CUs (M) and the EDU position of the CUs (F) are similar in both dataset.

iv) Classification tasks. All the data we prepare was performed using Perl scripts and Weka workbench (automatic feature extraction with bag of words).

- We converted each segment words into a set of attributes representing word occurrence information and we created a set of 1000, 5000 and 15000 words (attributes) using the training data. We represented each segment by an array of lemmas.
- We convert all letters to lower case.
- We followed bag of words approach and used tokens (unigrams, bigrams and trigrams) as features, where a classification instance is a vector of tokens appearing in the segmented text.[8]
- We also added EDU position and title word occurrence information to the feature vector. Thus, there was

[5] We think that this is due to that the CU was not considered in the annotation guidelines.

[6] The psychology texts were formated as well as we need.

[7] Multiple CUs (M) are the most difficult to detect by automatic means, whereas texts that the CU is in the first EDU (F) are the easiest to detect.

[8] We tried removing all words without linguistic meaning using a list of Spanish stop words (this list can be consulted at http://members.unine.ch/jacques.savoy/clef/), but the results were worse.

no attempt to remove or normalize them. Using weka's "string to word vector", text was converted into feature vector using TF-IDF (Manning, Raghavan, and Schtze, 2008) as feature value.

- Finally, the training set dictionary obtained using this scheme contains 1000 features; the same dictionary was used for the test set. TF-IDF feature valued representation was selected for Sequential Minimal Optimization (SMO) (Platt, 1998) and Multinomial Naive Bayes (MNB) (McCallum, Nigam, and others, 1998) systems, and boolean feature valued representation for Bernoulli Naive Bayes (BNB) (John and Langley, 1995) system.

3.4 Automatic feature selection

Feature selection is classic refinement method in classification. It is an effective dimensionality reduction technique to remove noise feature. In general, the basic idea is to search through all possible combinations of attributes in the data to find which subset of features works best for prediction. Removal is usually based on some statistical measures, such as segment frequency, information gain, chi-square or mutual information.

In this research, we have tested the two most effective feature selection methods: *i*) chi-square and *ii*) information gain using different set of attributes: 50, 100, 500 and 1000. Finally we performed all the classifiers using chi-square with a set of 100 attributes.

3.5 Classification

Classification was perform using WEKA workbench, to choose the best system. In our experiment we used 3 types of classifiers: *i*) Sequential Minimal Optimization (SMO),*ii*) Multinomial Naive Bayes (MNB) and *iii*) Bernoulli Naive Bayes (BNB).

In machine learning, Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between features.

The reasons to choose Naive Bayes models are:

- They only require a small amount of training data to estimate the parameters necessary for classification.
- They have been used successfully in similar tasks: for identifying thesis statements (Burstein et al., 2001) or for classifying short texts (McCallum, Nigam, and others, 1998).
- They can be used as predictive and descriptive model.

We have implemented three different ML methods:

- MNB. Multinomial Naive Bayes implements the naive Bayes algorithm for multinomially distributed data, and it is one of the two classic naive Bayes variants used in text classification (where the data is typically represented as word vector counts, although TF-IDF vectors are also known to work well in practice).
- BNB. Bernoulli Naive Bayes approach is the other classic naive Bayes variant. BNB trains classifiers on the absence and presence of features and using this information we can build a model to classify or select from a text the EDU that is the most likely candidate to be labeled as CU.
- SMO. Sequential Minimal Optimization is an optimization technique for solving quadratic optimization problems, which arise during the training of SVM and it has better generalization capability. Another reason for SMO is the high classification accuracy on different tasks reported in the literature (Schuller et al., 2012; Mairesse et al., 2007; Kermanidis, 2012) on personality traits recognition.

3.6 Evaluation

As a performance measure we used the average performance of our classifier using traditional recall (Rec.), precision (Prec.), and F-score (F_1) metrics. F-score was calculated with the standard measures as follows:

$$Prec. = \frac{correct_{CU}}{correct_{CU} + excess_{CU}}$$

$$Rec. = \frac{correct_{CU}}{correct_{CU} + missed_{CU}}$$

$$F_1 = \frac{2 * Prec. * Rec.}{Prec. + Rrec.}$$

where $correct_{CU}$ is the number of correct central units, $excess_{CU}$ is the number of overpredicted central units and $missed_{CU}$ is the

System	Data	C	E	M	P	R	F_1
Baseline	Train	34	26	35	0.492	0.566	0.527
	Test	6	7	8	0.428	0.461	0.444
BNB	Cross	51	39	18	0.566	0.739	0.641
	Test	11	6	3	0.647	0.785	0.709
MNB	Cross	58	22	11	0.725	0.841	0.779
	Test	11	3	3	0.785	0.785	0.785
SMO	Cross	50	5	19	0.909	0.725	0.806
	Test	11	4	3	0.733	0.786	0.759

Table 3: Results obtained on cross-validation and test sets

number of central units the system missed to tag.

We have compared the results of 3 systems against a simple baseline to detect the CU. This baseline is based on the position of the given EDU into the whole document.[9] The position is an important indicator, because we found that the likelihood of a CU occurring at the beginning of the text was 49.27% in the training set. So we consider that the first segment is the only CU of the text as our baseline.

The choice of algorithms is driven by their different properties for classification. Results were calculated as average of 10 experiments using 10-fold cross-validation and we compare the results of all the system in a box plot.[10] After that, we use the best system to extract the CUs of the test dataset. Results and error analysis are evaluated in this test dataset (see Subsection 4.2).

4 Results

Table 3 shows the results obtained using *i*) a baseline, *ii*) three different machine learning methods: BNB, MNB and SMO.

We can observe that SMO and MNB systems are better than baseline and BNB systems. The best model of the Table 3 is SMO which provides 0.806 in cross-validation and 0.759 in test.

Table 3 shows that SMO system is better than MNB system in 0.027 points in cross-validation, but in test dataset SMO system is worse than MNB system in 0.026 points. In the next subsection we compare all the systems in more detail.

4.1 A comparison using box plot

To show how robust the systems are on the dataset we run 10-fold cross-validation 10 times. The training dataset was randomly broken into 10 partitions using 10 random seeds. We have calculated 10 means of the F-score value for each 10-fold cross-validation (see Figure 1).

To visualize the performance of the 4 systems (Baseline, BNB, MNB and SMO), we have summarized the distribution of F-score values using box plots (Chambers, 1983).

Figure 1 shows the following main results:

- SMO and MNB classifiers show a greater F-score median value than BNB and Baseline F-score value.
- The best systems are SMO and MBM systems which has the same median value of F-score.
- And finally we can see that SMO is slightly better than MBM system because the upper and lower quartiles are slightly upper.

To understand how the CU detector works, we present the results obtained in the test dataset and an error analysis in the following subsection.

4.2 Error analysis

We analyze in Table 4 the results obtained with the best system from manual segmentation (SMO Gold) extracted from RST Spanish Treebank and from automatic segmentation (SMO Auto) performed with DiSeg (da Cunha et al., 2010) and we describe why SMO does not detect correctly some of these CUs from the test dataset.

The SMO Gold system has selected 5 TP (true positive) at the beginning of the text,

[9] Other baselines with linguistic features can be tested but we excluded then, because this is out of the objectives assigned to the study.

[10] A box plot consists of a box summarizing 50% of the data. The upper and lower ends of the box are the upper and lower quartiles, while a thick line within the box encodes the median. Dashed appendages summarize the spread and shape of the distribution, and dots represent outside values (see Figure 1).

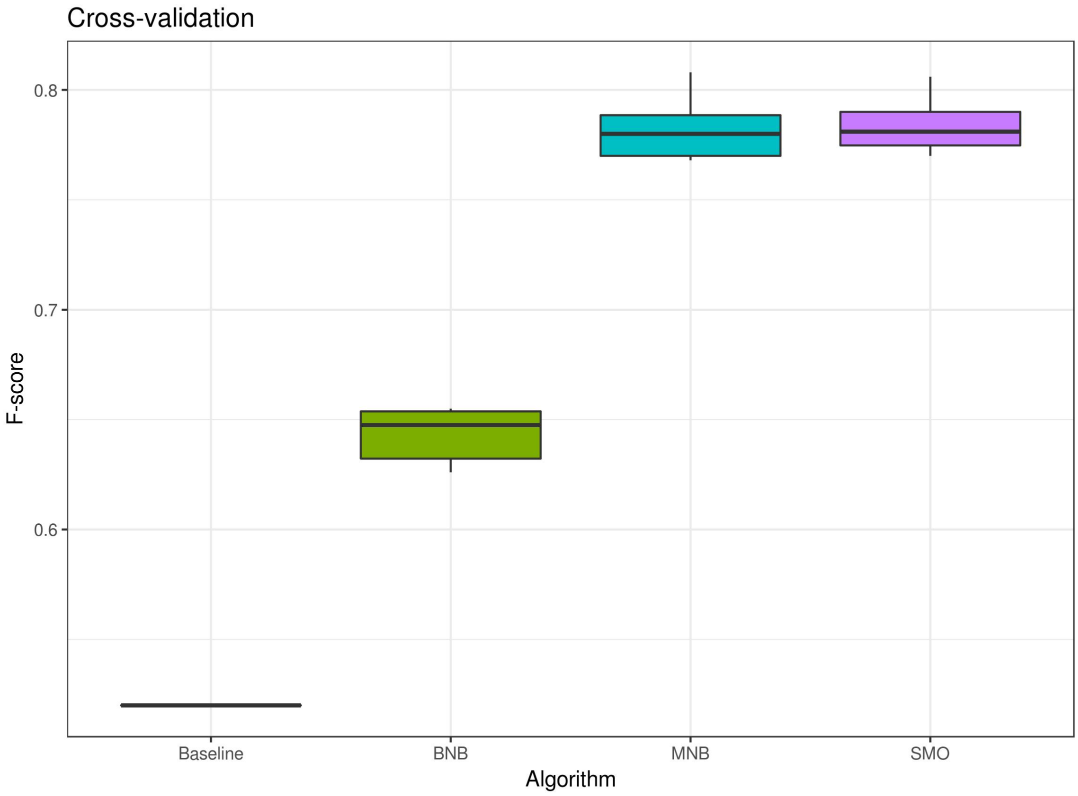

Figure 1: Exploring F-score distribution on the 10-fold cross-validation using 10 random seeds with Box Plot

System	Correct	Something wrong	
	Total agreem.	Partial agreem.	Total disagr.
SMO Gold	8	3	2
SMO Auto	8	2	3

Table 4: SMO's error analysis of the test dataset

2 TP at the middle and 1 TP at the end. Example 1 shows one CU that was found by the system.

(1) [el propósito de esta comunicación es hacer una reflexión sobre los retos a que se está enfrentando la neología terminológica en la realidad actual ;]$_{EDU2}$

Regarding the partial agreements, *i*) the system did not detect properly a CU at the end of the text, because it has selected 1 TP and another EDU as CU candidate (1 FP, false positive), that was some EDUs before, towards the middle of the text. *ii*) Another example that the system did not detect properly was a CU at the beginning of the text, because it has selected the CU (1 TP) and also other two false candidates (EDUs) at the end of the text (2 FPs). *iii*) The last one that the system has detected 1 TP of a multiple CU and did not detect the other EDU as a CU candidate (1 TN, true negative). This example of a partial agreement is presented in Example 2, in where the $EDU4$ was detected, but not the $EDU5$, which is in a clear conjunction.

(2) [el objetivo de el presente artículo es ; a_través_de un instrumento de evaluación de papel y lápiz ; evaluar el tipo de vínculo en la adolescencia]$_{EDU4}$ [y hacer correlaciones entre las calificaciones de la niñez y la adolescencia con_respecto_a el tipo de vínculo y las relaciones de pareja ;]$_{EDU5}$

Finally, the total disagreements were because the system could not detect a CU that was not indicated or written in a proper way. *i*) One of them, is at the end of the text and the CU is an intrasentential EDU. *ii*) The other has to objectives and the CU is an intrasentential EDU. Example 3 shows an example where the ML techniques ($EDU3$) disagree with the Gold Standard ($EDU9$).

(3) [el objetivo de nuestro proyecto es crear herramientas de aprendizaje de la lengua para estudiantes de formación profesional en las áreas de informática ; secretariado y electrónica]$_{EDU3}$ (...)
[nuestro artículo propone una metodología para la creación de una terminología plurilingüe]$_{EDU9}$

The results with the segmenter (SMO Auto) are only slightly worse (Table 4) and, therefore, we think that are acceptable. The small difference is that the system could not choose one CU that was partially correct in SMO Gold.

5 Discussion

An interesting point of this work is that in the annotation process of the Spanish RST Treebank (similar to the annotation of other RST Treebanks, such as Marcu (2000)) the CU was not considered during the annotation process. This will support, in such a sense, the claim that the CU is crucial point in RS-tree annotation, even when it is not considered in annotation guidelines.

In this paper we have introduced the first CU detector for Spanish[11] using SMO machine learning techniques without any linguistic design of features or rules in two subcorpus of the Spanish RST Treebank. The limitation of this work is that we could not use all the Spanish RST Treebank, due to some corpus formating constraints we think that are necessaries to develop CU detectors: *i*) text size and *ii*) title-body format of texts.

The experiments carried out on the corpus show competitive and promising results given the simplicity of the proposed method, which can be applied to different domains, if we have annotated RST treebank or a corpus partially annotated with discourse segments (EDUs) and CUs.

We are currently working to achieve the following aims:

- To reuse these techniques with other annotated data in different languages.
- To integrate the segmenter Diseg (da Cunha et al., 2010) and the CU detector for Spanish and follow up to detect some signaled discourse relations, to parse plain texts in Spanish and other languages as Basque, for example.

[11] A demo of the system can be tested here: http://ixa2.si.ehu.es/clarink/tools/ES-CU-detector/.

References

Alkorta, J., K. Gojenola, M. Iruskieta, and M. Taboada. 2017. Using lexical level information in discourse structures for basque sentiment analysis. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms, pages 39–47, Santiago de Compostela, Spain, September 4 2017. ACL.*

Bengoetxea, K., A. Atutxa, and M. Iruskieta. 2017. Un detector de la unidad central de un texto basado en técnicas de aprendizaje automático en textos científicos para el euskera. *Procesamiento del Lenguaje Natural*, 58:37–44.

Burstein, J., D. Marcu, S. Andreyev, and M. Chodorow. 2001. Towards automatic classification of discourse elements in essays. In *Proceedings of the 39th annual Meeting on Association for Computational Linguistics*, pages 98–105. ACL.

Carreras, X., I. Chao, L. Padró, and M. Padró. 2004. Freeling: An open-source suite of language analyzers. In *LREC*.

Chambers, J. M. 1983. *Graphical methods for data analysis*. Wadsworth Belmont, CA.

Cortes, C. and V. Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

da Cunha, I., E. SanJuan, J.-M. Torres-Moreno, M. T. Cabré, and G. Sierra. 2012. A symbolic approach for automatic detection of nuclearity and rhetorical relations among intra-sentence discourse segments in spanish. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 462–474. Springer.

da Cunha, I., E. SanJuan, J.-M. Torres-Moreno, M. Lloberas, and I. Castellón. 2010. Diseg: Un segmentador discursivo automático para el español. *Procesamiento del Lenguaje Natural*, 45:145–152.

da Cunha, I., J.-M. Torres-Moreno, and G. Sierra. 2011. On the Development of the RST Spanish Treebank. In *5th*

Linguistic Annotation Workshop (LAW V '11), pages 1–10, Portland, USA, 23 June. ACL.

Iruskieta, M., J. Antonio, and G. Labaka. 2016. Detecting the central units in two different genres and languages: a preliminary study of brazilian portuguese and basque texts. *Procesamiento de Lenguaje Natural*, 56:65–72.

Iruskieta, M., M. Aranzabe, A. Diaz de Ilarraza, I. Gonzalez, M. Lersundi, and O. L. de la Calle. 2013. The RST Basque TreeBank: an online search interface to check rhetorical relations. In *4th Workshop "RST and Discourse Studies"*, Brasil, October 21-23.

Iruskieta, M., A. D. de Ilarraza, and M. Lersundi. 2014. The annotation of the central unit in rhetorical structure trees: A key step in annotating rhetorical relations. In *COLING*, pages 466–475.

John, G. H. and P. Langley. 1995. Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 338–345. Morgan Kaufmann Publishers Inc.

Kermanidis, K. L. 2012. Mining authors' personality traits from modern greek spontaneous text. In *Proc. of Workshop on Corpora for Research on Emotion Sentiment & Social Signals, in conjunction with LREC*, pages 90–93. Citeseer.

Mairesse, F., M. A. Walker, M. R. Mehl, and R. K. Moore. 2007. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of artificial intelligence research*, 30:457–500.

Mann, W. C. and S. A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Manning, C. D., P. Raghavan, and H. Schtze. 2008. Relevance feedback and query expansion. *Introduction to Information Retrieval. Cambridge University Press, New York.*

Marcu, D. 2000. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448.

McCallum, A., K. Nigam, et al. 1998. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Madison, WI.

Platt, J. 1998. Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report. MSR-TR-98-14. Microsoft Research.

Schneider, K.-M. 2005. Techniques for improving the performance of naive bayes for text classification. *Computational Linguistics and Intelligent Text Processing*, pages 682–693.

Schuller, B. W., S. Steidl, A. Batliner, E. Nöth, A. Vinciarelli, F. Burkhardt, R. Van Son, F. Weninger, F. Eyben, T. Bocklet, G. Mohammadi, and B. Weiss. 2012. The interspeech 2012 speaker trait challenge. In *Interspeech*, volume 2012, pages 254–257.