Detecting Acronyms from Capital Letter Sequences in Spanish

Rubén San-Segundo[1], *Juan M. Montero*[1], *Verónica López-Ludeña*[1], *Simon King*[2]

[1]Speech Technology Group, ETSI Telecomunicación. UPM. Spain.
[2]Centre for Speech Technology Research, University of Edinburgh, UK.
lapiz@die.upm.es

Abstract

This paper presents an automatic strategy to decide how to pronounce a Capital Letter Sequence (CLS) in a Text to Speech system (TTS). If CLS is well known by the TTS, it can be expanded in several words. But when the CLS is unknown, the system has two alternatives: spelling it (abbreviation) or pronouncing it as a new word (acronym). In Spanish, there is a high relationship between letters and phonemes. Because of this, when a CLS is similar to other words in Spanish, there is a high tendency to pronounce it as a standard word. This paper proposes an automatic method for detecting acronyms. Additionaly, this paper analyses the discrimination capability of some features, and several strategies for combining them in order to obtain the best classifier. For the best classifier, the classification error is 8.45%. About the feature analysis, the best features have been the Letter Sequence Perplexity and the Average N-gram order.

Index Terms: Capital letter sequence pronunciation, Speech synthesis, Spelling, Spanish, Acronyms, Abbreviations

1. Introduction

When developing a Text to Speech (TTS) system, Capital Letter Sequences (CLS) can be pronounced in several ways. In many TTS, there is a list with the most frequent CLSs and their corresponding expansion. In this case, the TTS expands the CLS into several words (i.e. MIT: the Massachusetts Institute of Technology). The problem arises when the CLS is unknown. In this case, the system has two alternatives. The general one is to spell the CLS letter by letter (i.e. FBI: F B I). On the other hand, in some cases, the CLS can be pronounced as a new word (i.e. NATO), the CLS is an acronym. In Spanish, due to the high relationship between letters and phonemes, there is a high tendency to pronounce any CLS as a standard word when the CLS is similar to other words in Spanish. In this sense, there is not a general rule to decide when pronouncing it as word or letter by letter. This paper proposes an automatic method for selecting the best alternative when dealing with unknown CLSs. The main target is to detect acronyms (that can be pronounced as a standard word) rejecting the abbreviations (that must be spelled). This paper includes also an analysis of the discrimination power of several features considered in the classification task.

The rest of the paper is organized as follows. Section 2 summaries the state of the art in acronym pronunciation. Section 3 describes the database used in this work. Section 4 presents the main features proposed in this study. Section 5 presents the feature analysis using this database. Finally, section 6 shows the main classification experiments and section 7 summaries the main conclusions of this work.

2. State of the art

The problem addressed in this paper is one of the problems included in the research line of text normalization [1]. In this reference, the whole text normalization problem is described. Sproat and al present a Non-Standard Word (NSW) taxonomy with several proposals of automatic strategies for classifying every NSW within one of the taxonomy classes. According to acronyms, Sproat and al propose to consider acronyms as any other standard words and to use a dictionary (list of standard words) for discriminating between standard words and NSWs. This paper proposes an automatic method for selecting the best prononciation of unknow CLSs (not included in the dictionary).

In the literature, there are some research efforts focused on how to extract acronyms from raw text automatically [2] [3] [4]. In this studies, the main target is to increase the list of acronyms, in order to reduce the probability of being unknown. On the other hand, there are several works trying to model the creation of acronyms [5] or abbreviations [6]. From these efforts, it is possible to conclude that the syllable and letter structure are important aspect in the creation of acronyms and abbreviations. Based on these ideas, this paper will consider features based on a letter language model.

3. Database description

Table 1 summaries the main characteristics of the database considered in this work. This database has been extracted from several months of the "El Mundo" news-

paper [7]. The extraction process has been performed automatically (looking for words in capital letters) with a posterior manual revision.

Table 1: *Database characteristics.*

Characteristics	Acronyms	Abbreviations
Number of examples	653	684
Percentage	48.8%	51.2%
Average length	3.8	3.3
Maximum length	8	5

The first characteristic is that the percentage of acronyms is very high, very close to the percentage of abbreviations. This is due to the high relationship between letters and phonemes in Spanish. Another characteristic is that the average length of acronyms is higher. A smaller letter sequence (2 or 3 letters) has less probability of generating an acronym.

4. Main features based on letter sequences

This work has considered several features based on a letter language model (LLM) in order to decide if a Capital Letter Sequence (CLS) must be spelled or pronounced as a standard word. This language model has been generated considering the acronyms from the training set (in a Cross-Validation strategy described in section 6). Considering the maximun lengths reported in table 1, a 6-gram letter language model has been considered. This LLM has been trained using the IRSTLM toolkit [8]. In order to train this model, two marks for indicating the beginning and the end of the letter sequence have been considered (*ini* and *end*). For example, the letter sequence corresponding to the abbreviation IND, is "*ini* I N D *end*". This way, it is possible to incorporate information about the boundaries. The features derivated from the LLM are:

- Letter Sequence Perplexity (LS Perplexity). This feature is the perplexity of a CLS given a LLM. Considering that the LLM was generated using acronyms from the training set, it is expected that acronyms have lower perplexity than abbreviations.
- Minimum Probability (Min Prob). In some cases, part of the CLS can be very common in Spanish words while there are other parts forbidden in standard Spanish words. The perplexity, in these cases, can be small but the CLS should be rejected (not considered as an acronym). For example: MNAC, NAC has a very high probability and only the sequence MN generates a low probability. Considering the whole perplexity some local details are missed. Computing the minimum probability along the letter sequence can report additional information.
- Maximum Probability (Max Prob). Similar to the previous feature, in this case, the maximum probability computed along the letter sequence is considered.
- Average N-gram (Ave N-gram). This measurement is the average N-gram order for computing the probability of every letter. This feature tries to complement the information reported with the perplexity. Acronyms are expected to have a higher average N-gram order than abbreviations. For example, the N-gram orders for computing the probabilities in the MNAC sequence are: *ini* M(2-gram) N(1-gram) A(2-gram) C(3-gram) *end*(2-gram). In this case, the Average N-gram order is 10 / 5 = 2.
- Minimum N-gram (Min N-gram). Similar to the minimum probability, the minimum N-gram order is considered for reporting more details about local behaviours.
- Maximum N-gram (Max N-gram). In order to complete the analysis, the maximum N-gram order is also considered.

5. Feature Analysis

Fig. 1 shows the ROC curves for all the features. False acceptation is the percentage of cases where the system considers an abbreviation as an acronym by mistake, trying to pronounce it as a standard word. False rejection is the percentage of cases where the system classifies (wrongly) an acronym as an abbreviation, spelling it instead of pronouncing it as a standard word. As it is shown, the best features are the Letter Sequence Perplexity and the Average N-gram order. In both cases, it is possible to reach a 10% EER (Equal Error Rate). Minimun and Maximun N-gram order features have a discrete number of possible values, so the ROC curves present straight lines in some places.

Additional to the ROC curves, table 2 shows the Information Gain for each feature. Similar to the ROC curves, the best features are LS Perplexity and Ave N-gram. The worst ones are the Max Prob and the Max N-gram. This analysis has been performed using the WEKA toolkit [9].

Table 2: *Information Gain Analysis*

Feature	Information Gain
Ave N-gram	**0.67**
LS Perplexity	**0.66**
Min Prob	0.47
Min N-gram	0.42
Max N-gram	0.40
Max Prob	0.36

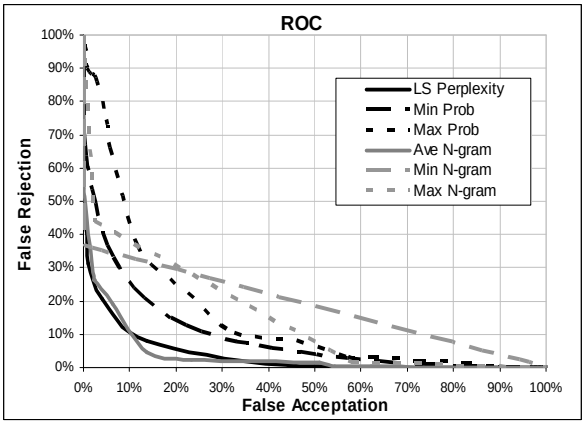

Figure 1: *ROC for the different features.*

6. Classification experiments

For the classification experiments, a Cross-Validation strategy has been considered. The whole database has been ramdonly divided into 10 subsets: eight for training the LLM, one for tunning the classifier parameters and one for testing. The experiments are repeated 10 times and the results are the average of all runs.

6.1. Grouping Features

The first analysis is about how to group the different features. Several alternatives have been considered and evaluated in table 3. These grouping alternatives are: the two best features, the LLM probability based features, the N-gram order based features, and considering all of them. Table 3 shows the classification error and the ROC area (area under the ROC curve) for the different alternatives. In this case, a simple classifier has been considered: a naive Bayes classifier. This classifier is a simple probabilistic classifier based on applying Bayes' theorem assuming independence between features.

Table 3: *Classification experiments considering several grouping alternatives*

Features group	ROC area	Class. error
Ave N-gram + LS Pp	0.968	12.04%
LS Pp, Min + Max Prob	0.962	11.90%
Ave + Min + Max N-gram	0.966	13.98%
All the features	**0.968**	**11.20%**

As it is shown in table 3, LLM probability based features perform better than N-gram based ones. The best results are obtained considering all the features.

6.2. Using Several Classifiers

This sub-section presents experiments considering different classifiers for combining all the features. Table 4 shows the classification error and the ROC area (area under the ROC curve) for the different alternatives:

- Naive Bayes. Same as the classifier considered in the previous section.
- Bayesian Network. This classifier is based on Bayesian Networks. A Bayesian Network is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG).
- Multi-Layer Perceptron. A Multi Layer Perceptron is a feedforward artificial neural network consisting of multiple layers of nodes in a directed graph (3 layers in this case), with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function (sigmoide).
- Decision Tree using the C4.5 algorithm [10].
- Attribute Selected Classifier. This is meta-classifier that takes a search algorithm and evaluator next to the base classifier. In this case, the base classifier is the decision tree generated using the C4.5 algorithm.

Table 4: *Classification experiments with different classifiers*

Features group	ROC area	Class. error
Naive Bayes	0.968	11.20%
Bayesian Network	0.969	10.40%
Multi-Layer Perceptron	0.967	9.57%
Decision Tree C4.5	0.940	8.67%
Attribute Selected Classifier	**0.943**	**8.45%**

In these experiments (table 4), the best classifier has been the decision tree obtained with the C4.5 algorithm. In this case, the classification error is 8.67%. Using the meta-classifier (Attribute Selected Classifier) the classification error is 8.45%.

Fig. 2 shows the top nodes of the decision tree obtained (using WEKA). As it is shown, decision over Average N-gram, LS perplexity and Min N-gram features are situated at the top nodes.

Fig. 3 shows the ROC curves for all the classifiers. False acceptation represents the percentage of abbreviations classified as acronym by mistake, and the false rejection is the percentage of acronyms classified as abbreviations. When classifying an acronym as an abbreviation (false rejection), the system will spell the CLS instead of pronouncing it as a standard word. On the other hand, when the system classifies an abbreviation as an acronym (false acceptance), the system will try to pronounce it as

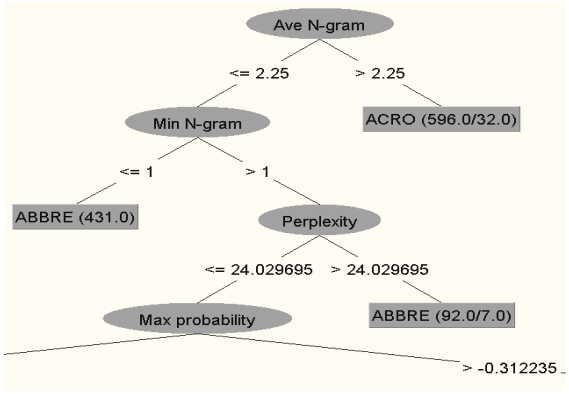

Figure 2: *Top nodes of the decision tree*

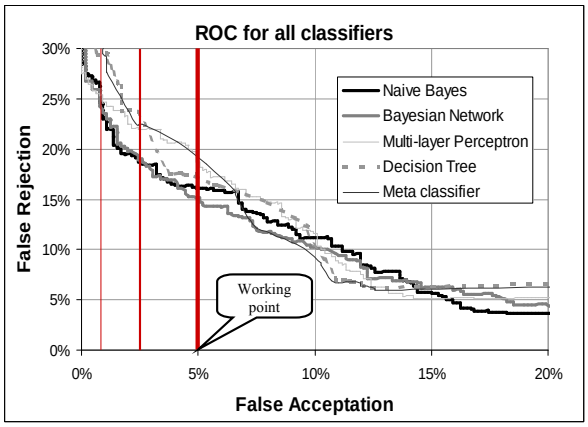

Figure 3: *ROC curves considering the different classifiers.*

a standard word generating a wrong pronunciation. This second type of error is the most dangerous one. Because of this, in order to define a working point, the false acceptance must be lower than 5%. Table 5 shows the false rejection for several working points (false acceptation percentages), considering the best system in this range: the Bayesian Network.

Table 5: *Different working points*

False Acceptance	False Rejection
5.0%	15.1%
2.5%	18.5%
1.0%	23.5%

7. Conclusions

This paper has presented an automatic strategy to classify Capital Letter Sequences (CLSs) as abbreviation or acronyms, when these CLSs are unknown by the TTS (they are not in the system vocabulary). The abbreviations will be spelled while the acronyms will be pronounced as standar words. In Spanish, there is a high percentage of acronyms that makes this work very interesting. This paper has also analysed the discrimination capability of some features based on a Letter Language Model (LLM): letter sequence perplexity, minimum and maximum probability and, average, minimum and maximum N-gram order when computing the probability along the letter sequence. The best features have been the letter sequence perplexity and the average N-gram order. Finally, the paper has evaluated several classifiers for combining the differents features. The lowest classification error (8.45% in table 4) has been obtained with a meta classifier that uses a decision tree based on the C4.5 algorithm (this has been the best meta classifier in this task). When analysing the working point (a false acceptance lower than 5%), the best classifier, in this range, is the Bayesian Network.

8. Acknowledgements

The work leading to these results has received funding from the European Union under grant agreement n 287678. It has also been supported by TIMPANO (TIN2011-28169-C05-03), INAPRA (MICINN, DPI2010-21247-C02-02) and MA2VICMR (CAM, S2009/TIC-1542) projects. Authors also thank the other members of the Speech Technology Group and Simple4All project for the continuous and fruitful discussion on these topics.

9. References

[1] Sproat, R., et al. ”Normalization of non-standard words.” Computer Speech and Language, 15(3), 287-333, 2001.

[2] Larkey, Leah, Paul Ogilvie, Andrew Price and Brenden Tamilio. ”Acrophile: An Automated Acronym Extractor and Server”, In Proceedings of the ACM Digital Libraries conference, pp. 205-214, 2000.

[3] Stuart Yeates. ”Automatic Extraction of Acronyms from Text”. New Zealand Computer Science Research Students’ Conference. 1999.

[4] J.T. Chang, H Schtze, and R.B. Altman. 2002. ”Creating an Online Dictionary of Abbreviations from MEDLINE” JAMIA.

[5] Cannon, Garland. 1989. ”Abbreviations and acronyms in English word-formation.” American Speech, 64:99127.

[6] Pennell and Liu. 2011. ”Toward text message normalization: Modeling abbreviation generation.” Proceedings of the IEEE. pp. 5364-5367.

[7] El Mundo Newspaper (1996-2006). New from the ”El Mundo” newspaper during several years.

[8] Federico, M. and Cettolo, M. (2007) Efficient Handling of N-gram Language Models for Statistical Machine Translation Proc. of ACL Workshop on SMT. pages 8895, Prague, Czech Republic.

[9] Witten, I.H. and Frank, E. (2005) Data Mining: Practical machine learning tools and techniques. 2nd edition Morgan Kaufmann, San Francisco.

[10] Quinlan, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, 1993.