## Capítulo 10

# MALINCHE: A NER system for Portuguese that reuses knowledge from Spanish

Thamar Solorio

Diana Santos e Nuno Cardoso, editores, *Reconhecimento de entidades mencionadas em português: Documentação e actas do* HAREM*, a primeira avaliação conjunta na área*, Capítulo 10, p. 123–136, 2007.

Due to the many potential uses of named entities (NE) in higher level NLP tasks, a lot of work has been devoted to developing accurate NE recognizers. Earlier approaches were primarily based on hand-coded knowledge, lists of gazetteers, and trigger words (Appelt et al., 1995; Krupka e Hausman, 1998; Black et al., 1998; Téllez et al., 2005). More recently, as machine learning has increased its popularity within the NLP community, NER systems are taking advantage of machine learning algorithms (Arévalo et al., 2002; Bikel et al., 1997, 1999; Borthwick, 1999; Carreras et al., 2002, 2003b; Madrigal et al., 2003; Petasis et al., 2000; Sekine et al., 2002; Zhou e Su, 2002). However, lists of trigger words and gazetteers remain a key component of these systems.

Newer approaches try to avoid limitations of language dependency by tackling NER on a multilingual setting (Carreras et al., 2003a; Curran e Clark, 2003; Florian et al., 2003; Maynard et al., 2003b), and although it is very unlikely that a general NER system performing well across all languages will exist in the near future, recent systems have successfully achieved higher portability than that of previous attempts. The main goal of this research work is to provide a representation of the learning task that increases coverage of a hand-coded NE tagger and evaluate its effectiveness and portability to different collections and languages. Our approach needs to be flexible and easy to port so that an average user can adapt the system to a particular collection or language. In a previous work we presented results of extending the coverage of a hand-coded tagger for Spanish to different texts (Solorio, 2005). Here we show how the same representation can be used to perform NE extraction in Portuguese without needing to adapt the task to Portuguese. Results presented here show that it is possible to perform NE extraction on both languages, Spanish and Portuguese, using the same design for the learning task.

The next section describes our framework for NE extraction. Section 10.2 presents the results of performing NE extraction on Portuguese using the framework previously described. The paper concludes by summarizing our findings.

## 10.1 The MALINCHE System

Similar to the strategy used by other researchers in previous approaches, we divide the NER problem into two sub-tasks that are solved sequentially:

1. We first determine which words, or sequences of words, are likely to be NEs. This task is called Named Entity Delimitation (NED).

2. Once we have extracted possible NEs from documents, we then try to categorize each NE into one of the following classes: `PERSON`, `ORGANIZATION`, `LOCATION` and `MISCELLANEOUS`. This task is called Named Entity Classification (NEC).

We decided to divide the problem in this way considering the unbalanced distribution of data. Normally, in a given document around 10%, or at most 18%, of words are NE.

This unbalanced distribution can cause trivial classifiers to achieve accuracies of up to 85% by tagging all words in the document as non-NE. We can circumvent this problem by carefully selecting the learning algorithm, or by assigning a cost matrix to the classification errors. Some authors, working with classification problems with similar conditions, have used the solution of selecting the training instances in an attempt to give the learner a well balanced training set. This can be achieved by means of *over-sampling*, where instances of the ill-represented classes are randomly selected and added to the training set (Ling e Li, 1998), or *under-sampling*, where random instances of the over represented class are removed to balance the distribution (Zhang e Mani, 2003). Whatever the alternative taken, we can not be certain that the bias for selecting the most frequent tag can be completely removed. Moreover, according to a study performed by Japkowicz (2003), when class imbalances cause low classification accuracies it is best to tackle the small disjunct problem (Holte et al., 1989) than to attempt to rectify the imbalances. Thus, even though for some works this condition does not seem to be a problem, for example (Borthwick, 1999), we opted for the strategy of performing NED first and then NEC. This separation of tasks will allow for different attributes for each task, and thus, we can tackle each subproblem using a different strategy.

The methods we developed for NED and NEC are very similar in spirit. In both cases we take advantage of the tags assigned by the hand-coded tagger[1] and use them together with some lexical features to train a learning algorithm. Our goal is to allow the classifier to take advantage of the knowledge the hand-coded tagger has about the NER task. Going a step further, we want the classifier to learn from the hand-coded tagger mistakes. This is why a key component in our method is precisely the output of the hand-coded tagger, because we believe it provides valuable information. In the following sections we describe in more detail the NED and NEC methods.

### 10.1.1 Named Entity Delimitation

As mentioned earlier, in this task we are concerned with extracting from documents the words, or sequences of words, that are believed to be NE. This extraction process can be performed by means of classifying each word in the document with a tag that discriminates NE. In our classification setting we use the BIO scheme, where each word is labelled with one of three possible tags, according to the following criteria:

- The B tag is for words that are the beginning of a NE.
- The I tag is assigned to words that belong to an NE, but they are not at its beginning.
- The O tag is for all other words that do not satisfy any of the previous two conditions. All words not belonging to NE are assigned the O tag.

[1] The hand-coded system used in this work was developed at the TALP research center by Carreras e Padró (2002).

| Word | BIO Class |
|---|---|
| Monaco | B |
| was | O |
| in | O |
| mourning | O |
| for | O |
| the | O |
| death | O |
| of | O |
| Prince | B |
| Rainier | B |
| III | I |

Table 10.1: An example of NED using the BIO classification scheme

Let $D_R$ be the set of labelled documents that will be used for training
Let $D_T$ be the set of test documents
TRAINING
1. Label $D_R$ with PoS and NE tags using the hand-coded tagger
2. Transform the PoS tags assigned by the hand-coded tagger to the compact set of tags
3. Transform the NE tags from the output of the hand-coded tagger to BIO format
4. Build the training instances adding to the output of the hand-coded tagger the training attributes
5. Give the learning algorithm the training instances and perform training

TESTING
1. Label $D_T$ with PoS and NE tags using the hand-coded tagger
2. Transform the PoS tags assigned by the hand-coded tagger to the compact set of tags
3. Build the test set adding to the output of the hand-coded tagger the training attributes
4. Transform the NE tags from the output of the hand-coded tagger to BIO format
5. Let the trained classifier label the test instances

Table 10.2: The NED algorithm.

An example of a possible output for this classification setting is shown in Table 10.1. Here we present a sentence where each word is classified under the BIO scheme.

The algorithm for NED is summarized in Table 10.2. As we can see, the only processing we need to perform are two transformations of the output of the hand-coded system. One postprocessing step was needed in order to reduce the set of PoS tags. The hand-coded tagger has a set of tags that gives detailed information about each word. That is, in addition to giving the word category, it also gives information concerning the type, mode, time, person, gender, and number, whenever possible. Then, for the category verb there are around 600 possible tags. We decided to eliminate some of this information and retain only what we consider most relevant. For all categories we kept only information regarding their main PoS category, a detailed description of the reduced list can be found

| Word | Hand-coded tag | BIO tag |
|---|---|---|
| La | O | O |
| Comisión | ORG | B |
| Nacional | ORG | I |
| del | ORG | I |
| Agua | ORG | I |
| alertó | O | O |
| el | O | O |
| desbordamiento | O | O |
| del | O | O |
| río | O | O |
| Cazones | LOC | B |

Table 10.3: An example of how the tags assigned by the hand-coded tagger to the sentence are translated to the BIO scheme.

in Solorio (2005). The other postprocessing step is required to map the NE tags from the hand-coded tagger to the BIO tags; the hand-coded tagger does not assign BIO tags, instead it recognizes the NE in the documents and classifies them according to our predefined set. A very simple program analyzes these tags and translates them to the BIO scheme. Table 10.3 shows an example, where the hand-coded tagger tags are translated to the BIO scheme for the sentence *La Comisión Nacional del Agua alertó el desbordamiento del río Cazones* translated to English as *The National Commission of Water warned the flooding of the Cazones river*.

This NED algorithm is independent of the learning algorithm used to build the classifier. We can use the algorithm of our preference, provided it is well suited for this kind of learning task. In our evaluation we have used as learning algorithm Support Vector Machines (SVM) (Vapnik, 1995; Stitson et al., 1996). We give a brief description of this learning strategy on Subsection 10.1.4.

### 10.1.2 The features

The representation of instances of the learning concept is one of the most important considerations when designing a learning classification task. Each instance is represented by a vector of attribute values. For our problem, each word $w_i$ is described by a vector of five attributes, $\langle a_1, a_2, \ldots, a_5 \rangle$ , where $a_1$ to $a_3$ are what we call internal, or lexical, features: the word $w_i$, the orthographic information, and the position of the word in the sentence, respectively. Attributes $a_4$ and $a_5$ are the PoS tag and the BIO tag, both assigned by the hand--coded tagger. These two attributes are considered as external features, given that they are acquired from external sources, while the internal features are automatically gathered from the documents. In addition to this, we use for each word $w_i$ the attributes of the two words surrounding $w_i$; that is, the attributes for words $w_{i-2}$, $w_{i-1}$, $w_{i+1}$ and $w_{i+2}$. The final

feature vector for a given word $w_i$ is the following:

$$w_i = [a_{1_{w_{i-2}}}, ..., a_{5_{w_{i-2}}}, a_{1_{w_{i-1}}}, ..., a_{5_{w_{i-1}}}, a_{1_{w_i}}, ..., a_{5_{w_i}}, a_{1_{w_{i+1}}}, ..., a_{5_{w_{i+1}}}, a_{1_{w_{i+2}}}, ..., a_{5_{w_{i+2}}}, c_i] \quad (10.1)$$

where $c_i$ is the real class for word $w_i$.

To illustrate this, consider the sentence *El Ejército Mexicano puso en marcha el Plan DN-III*, the attribute vector for word *Mexicano* is the following:

$w_{Mexicano}$ = [*El*, 3, 1, DA, O,
*Ejército*, 3, 2, N C, B,
*Mexicano*, 3, 3, N C, I,
*puso*, 2, 4, V M, O,
*en*, 2, 5, SP, O,
I]

Within the orthographic information we consider 6 possible states of a word. A value of 1 in this attribute means that the letters in the word are all capitalized. A value of 2 corresponds to words where all letters are lower case. Value 3 is for words that have the initial letter capitalized. A 4 means the word has digits, 5 is for punctuation marks and 6 refers to marks representing the beginning and end of sentences.

Note that the attributes $a_{5_{w_i}}$ and $c_i$ will differ only when the base hand-coded tagger misclassifies a named entity, whereas by erroneously mixing the *B* and *I* tags; or by failing to recognize a word as an NE, in this case tags *B* and *I* will be misclassified by the hand-coded tagger as *O*. Intuitively, we may consider the incorrectly classified instances as noisy. However, we believe that by having the correct NE classes available in the training corpus, the learner will succeed in generalizing error patterns that will be used to assign the correct NE. If this assumption holds, that learning from other's mistakes is helpful, the learner will end up outperforming the initial hand-coded tagger.

The idea of the BIO labelling scheme, which uses three tags: *B*, *I* and *O*, for delimiting NE follows the work by Carreras et al. (2003a,b). The differences between their approach and the one proposed here lie in the representation of the learning task and the classification process. Concerning the attributes in the representation of problem instances, Carreras et al. include chunk tags of window words, chunk patterns of NE, trigger words, affixes and gazetteer features, none of them were used in our work. Their classification process is performed by selecting the highest confidence prediction from three binary AdaBoost classifiers, one for each class. In contrast, our classifier is a multi class adaptation of SVM.

### 10.1.3 Named Entity Classification

NE Classification is considered to be a more complex problem than NED. This may be due to the fact that orthographic features are less helpful for discriminating among NE classes.

| Internal Features | | | External Features | | Real class |
|---|---|---|---|---|---|
| Word | Caps | Position | POS tag | NEC tag | |
| El | 3 | 1 | DA | O | O |
| Ejército | 3 | 2 | NC | ORG | ORG |
| Mexicano | 3 | 3 | NC | ORG | ORG |
| puso | 2 | 4 | VM | O | O |
| en | 2 | 5 | SP | O | O |
| marcha | 2 | 6 | NC | O | O |
| el | 2 | 7 | DA | O | O |
| Plan | 3 | 8 | NC | O | MISC |
| DN-III | 1 | 9 | NC | ORG | MISC |

Table 10.4: An example of the attributes used in the learning setting for NEC in Spanish for the sentence *El Ejército Mexicano puso en marcha el Plan DN-III* (The Mexican Army launched the DN-III plan).

The majority of NE seem to have very similar surface characteristics, and as a consequence envisioning good attributes for the task becomes more challenging. A common strategy to achieve good accuracy on NEC is to use linguistic resources such as word lists, dictionaries, gazetteers or trigger words. These resources are very helpful, and many of them are easily built because they are available in machine-readable format. However for most languages these resources have not been created yet, plus they can become obsolete quite rapidly. In this work, we try to use features without restricted availability, so we restrained the source of features to the information in the documents themselves.

The final set of features used in the NEC task includes all the attributes described in the NED task. Originally we thought it would be necessary to add other attributes for this task, as NEC poses a greater challenge to the learner. It turned out that the original set of features was good enough, and we will discuss this in more detail in the following section. Then, for a given word $w$ we have as internal features the word itself (attribute $a_1$), orthographic information, ($a_2$), and the position in the sentence of word $w$ ($a_3$). The external features also remained unchanged for the NEC task. We use the PoS tags and the NE tags from the hand-coded tagger. In Table 10.4 we present the features that describe each instance in this NEC task.

A summary of the NEC algorithm is presented in Table 10.5. Note, however, that concerning the output of the hand-coded tagger, the NE tags remain unchanged for this task.

### 10.1.4 The machine learning algorithm

The methods proposed in this work to solve the NER problem are used in combination with a machine learning algorithm. Note, however, that they are not designed to work with a specific learning algorithm. Rather, we can select the most appropriate algorithm considering the type of the learning task, the computing resources, namely CPU and me-

Let $D_R$ be the set of labelled documents that will be used for training
Let $D_T$ be the set of test documents
TRAINING
1. Label $D_R$ with PoS and NE tags using the hand-coded tagger
2. Transform the PoS tags assigned by the hand-coded tagger to the compact set of tags
3. Build the training instances adding to the output of the hand-coded tagger the internal attributes
4. Give the learning algorithm the training instances and perform training

TESTING
1. Label $D_T$ with PoS and NE tags using the hand-coded tagger
2. Transform the PoS tags assigned by the hand-coded tagger to the compact set of tags
3. Build the test set adding to the output of the hand-coded tagger the internal attributes
4. Let the trained classifier label the test instances

Table 10.5: The NEC algorithm

mory, and the amount of time we are willing to spend on the training and testing of the algorithm.

In this work we selected for our experiments Support Vector Machines as the learning strategy. However it is worth mentioning that due to computer resources constraints we did not carry out experiments with other learning schemes. For instance, ensemble methods are a promising alternative, as it is well known that they are a powerful learning strategy that usually outperforms the individual classifiers that make up the ensemble (Dietterich, 2000). Our main concern in this work is not to find the best learning algorithm for NER, but come up with a good representation of the learning problem that could be exploited in conjunction with any powerful learning algorithm. Thus, we selected the best algorithm that we could afford experimenting with and we consider the results reported throughout this document as a lower bound on classification measures. With a more powerful learning strategy, such as ensembles, and a larger training set, results could be improved considerably.

**Support Vector Machines**

Given that Support Vector Machines have proven to perform well over high dimensionality data, they have been successfully used in many natural language related applications, such as text classification (Joachims, 1999, 2002; Tong e Koller, 2001) and NER (Mitsumori et al., 2004). This technique uses geometrical properties in order to compute the hyperplane that best separates a set of training examples (Stitson et al., 1996). When the input space is not linearly separable SVM can map, by using a kernel function, the original input space to a high-dimensional feature space where the optimal separable hyperplane can be easily calculated. This is a very powerful feature, because it allows SVM to overcome the limitations of linear boundaries. They also can avoid the over-fitting problems of neural

| Class | Instances |
|---|---|
| B | 648 |
| I | 293 |
| O | 7,610 |

Table 10.6: Distribution of examples in the Portuguese corpus for the NED task.

networks as they are based on the structural risk minimization principle. The foundations of these machines were developed by Vapnik, and for more information about this algorithm we refer the reader to Vapnik (1995) and Schölkopf e Smola (2002).

In our work, the optimization algorithm used for training the support vector classifier is an implementation of Platt's sequential minimal optimization algorithm (Platt, 1999). The kernel function used for mapping the input space was a polynomial of exponent one. We used the implementation of SVM included in the WEKA environment (Witten e Frank, 1999).

## 10.2 Named Entity Recognition in Portuguese

We believe that the portability of our method is very important, even though we know that our method will not be completely language independent. There are important differences across languages that do not allow for a general NLP tool to be built, and the same applies to an NE tagger. We can aim at developing tools that will be useful for similar languages, which is a reasonable and practical goal, and is one of our goals in this research work. We are not expecting that our method will perform well on languages such as English or German, but we can expect it to be useful for other languages similar to those used in the current study, such as Italian, Portuguese or even Romanian. Considering that our method is based on an existing tagger for Spanish, it is reasonable to expect better results for Spanish than for any other language. However, if our method is capable of achieving good results for a different language, then we can claim it is a portable method, and it can be exploited to perform NER on several languages without any modifications.

In this Section, we evaluate the classification performance of our method on Portuguese. For this we used the training corpus provided by HAREM (see Chapter 1)[2]. This corpus contains documents of various literary genres. The corpus has 8,551 words with 648 NE. The following sections present our experimental results.

| | | | | Attributes | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Hand-coded tagger | | | Internal | | | External | | | Internal & External | | |
| Class | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* |
| B | 60.0 | 68.8 | 0.641 | **82.4** | 85.8 | 0.841 | 75.9 | 81.0 | 0.784 | 82.1 | **87.8** | **0.849** |
| I | 64.5 | 73.3 | 0.686 | 80.1 | 76.8 | 0.784 | 73.8 | 70.3 | 0.720 | **80.9** | **77.8** | **0.793** |
| O | 97.2 | 95.5 | 0.964 | 98.7 | 98.5 | 0.986 | 98.1 | 97.7 | 0.979 | **98.8** | 98.4 | 0.986 |
| Overall | 73.9 | 79.2 | 0.763 | 87.0 | 87.0 | 0.870 | 82.6 | 83.0 | 0.827 | 87.2 | 88.0 | 0.876 |

Table 10.7: Experimental results for NED in Portuguese.

### 10.2.1 Results on NED

In this section we report our results of NED in Portuguese. We describe the distribution of instances over classes for the Portuguese corpus in Table 10.6. As the goal is to explore to what extent our method can be applied to similar languages, we did not make any particular changes to our system. The method is applied in the same way as it was applied previously to Spanish, results for Spanish can be found at Solorio (2005). Experimental results on NED are presented in Table 10.7. These results are averaged using a 10-fold cross validation[3]. We can observe that the hand-coded tagger achieved surprisingly high classification measures, it reached an F measure of 0.763. We believe that these results reveal that the two languages share some characteristics, among them the orthographic features: in Portuguese it is also conventional to write proper names with the first letter in uppercase. On the other hand, note also that the behavior of the two types of features differs greatly from that observed for Spanish. The internal features have better results than the external, for Spanish we observed that external features achieved better results than the internal ones. A plausible explanation to this is that, given that the hand-coded tagger misclassified more instances in the Portuguese case, then it is harder for the SVM, trained with the output of the hand-coded tagger, to learn the task in this somehow noisier setting. Nonetheless, SVM did improve the accuracy of the hand-coded tagger, and even more relevant for us, the combination of the two types of features yielded the best results. In this setting, our method is still the best option to achieve higher precision and recall on NED in Portuguese.

### 10.2.2 Results on NEC in Portuguese

We have shown that our proposed solution works well for Portuguese NED, now we need to evaluate how well this solution works for NEC in Portuguese. In this case the classifi-

[2] **Editors' note.** Note that the author does not apply in the chapter the measures used for HAREM elsewhere in this book, but rather defines her own, such as accuracy per word. Also she uses a small subset of the first golden collection, not the full golden collection.

[3] Since this is a classification task where we need to assign to every word one out of three possible classes, we measure per word accuracies.

| Class | Instances |
|---|---|
| PESSOA | 237 |
| COISA | 4 |
| VALOR | 68 |
| ACONTECIMENTO | 14 |
| ORGANIZACAO | 195 |
| OBRA | 56 |
| LOCAL | 187 |
| TEMPO | 112 |
| ABSTRACCAO | 55 |
| VARIADO | 13 |

Table 10.8: Distribution of examples in the Portuguese corpus for the NEC task.

cation task is more difficult due to several factors, among them, those we have discussed previously (Subsection 10.1.3). Another relevant factor is that the Portuguese corpus has a different set of NE classes than that of the hand-coded tagger. This Spanish tagger discriminates only among four different classes, namely PERSON, ORGANIZATION, LOCATION and MISCELLANEOUS. For the Portuguese set the classifier needs to assign NE tags from a set of 10 classes, these are PESSOA (person), COISA (object), VALOR (quantity), ACONTECIMENTO (event), ORGANIZACAO (organization), OBRA (artifact), LOCAL (location), TEMPO (date/time expression), ABSTRACCAO (abstraction) and VARIADO (miscellaneous). This will require the SVM to discover a function for mapping from the reduced set of classes to the larger set. Yet another complicating factor is the distribution of examples in the Portuguese set, which is shown in Table 10.8. We can observe that there are several classes for which we have very few examples, then there is little information for the classifier to learn these classes well. The following experimental results will show that these are not issues to be concerned of, the classifier does learn this type of target function. However, it is evident that more examples of the poorly represented classes can make a considerable difference in the classification performance.

Table 10.9 presents the results of NEC in Portuguese. Here again, we compared the four sets of results: the hand-coded tagger for Spanish, the internal features only, the external features only and the combination of both features. Similarly as in the NEC experiments we measured per word accuracies, but independently from the NED task[4]. The hand-coded tagger performed poorly, the overall $F$ measure barely reaches the 0.10, and naturally it has an $F$ measure of 0 on all the instances belonging to the classes not included on its set of classes. However, the hand-coded tagger has also an $F$ measure of 0 for the VARIADO (miscellaneous) class even though for Spanish the hand-coded tagger was able to label corrECTLY some of the instances in this class.

[4] These results are optimistic since we are assuming a perfect classification on the NED task. On a real scenario the errors on NED classification would be carried on to the NEC task, degrading the performance of the NEC task.

| | | | | Attributes | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Hand-coded tagger | | | Internal | | | External | | | Internal & External | | |
| Category | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* |
| PESSOA | 34.8 | 72.5 | 0.466 | 87.7 | 92.9 | 0.9023 | 47.7 | 74.0 | 0.58 | 83.3 | 89.6 | 0.864 |
| COISA | 0 | 0 | 0 | 0 | 0 | 0 | 50.0 | 25.0 | 0.333 | 0 | 0 | 0 |
| VALOR | 0 | 0 | 0 | 89.0 | 79.9 | 0.842 | 76.9 | 78.4 | 0.777 | 87.1 | 89.7 | 0.884 |
| ACONTECIMENTO | 0 | 0 | 0 | 1 | 76.2 | 0.864 | 83.3 | 9.5 | 0.169 | 1 | 38.1 | 0.550 |
| ORGANIZACAO | 41.4 | 38.4 | 0.393 | 83.4 | 88.9 | 0.849 | 46.5 | 48.0 | 0.472 | 79.7 | 85.5 | 0.825 |
| OBRA | 0 | 0 | 0 | 94.0 | 91.4 | 0.927 | 57.0 | 21.2 | 0.309 | 92.3 | 82.1 | 0.869 |
| LOCAL | 52.5 | 16.5 | 0.248 | 79.8 | 80.8 | 0.803 | 53.8 | 46.2 | 0.497 | 75.9 | 77.6 | 0.767 |
| TEMPO | 0 | 0 | 0 | 85.2 | 88.0 | 0.866 | 85.5 | 81.3 | 0.833 | 87.7 | 87.7 | 0.877 |
| ABSTRACCAO | 0 | 0 | 0 | 86.9 | 71.0 | 0.782 | 26.3 | 4.4 | 0.075 | 81.8 | 67.9 | 0.742 |
| VARIADO | 0 | 0 | 0 | 63.9 | 18.2 | 0.280 | 0 | 0 | 0 | 33.3 | 3.03 | 0.056 |
| Overall | 12.8 | 12.7 | 0.110 | 77.0 | 68.7 | 0.712 | 52.7 | 38.8 | 0.404 | 72.1 | 62.1 | 0.643 |

Table 10.9: NEC performance on the Portuguese set.

SVM trained with only the external features achieved impressive improvements, it is surprising to see how good this classifier performs, especially on the classes where the hand-coded tagger had errors of 100%. Consider for example, the case of the classes COISA and OBRA, the error reductions of these classes are quite large, external features achieved F measures of over 0.30, we were able to reduce the classification errors for more than 30%. We consider this an excellent achievement of this method.

On the other hand, internal features helped SVM to outperform the results of external ones, reaching F measures as high as 0.927 on the OBRA class. The set of results attained by the internal features are the best overall, leaving the SVM classifier combining both internal and external features as the second best. It is interesting to observe how, the internal features helped boost classification performance of the SVM trained with the external features, when both are combined. Regarding the performance of the SVM with internal features, we cannot assert the same, given that in this case the internal features performed better than the combination. It seems that, for Portuguese, combining both types of features was beneficial only in one direction.

As we mentioned at the beginning of this section, the hand-coded tagger classifies NE only into four categories. Considering this, it might be a little unfair to compare our method against the performance of the hand-coded tagger, as presented on Table 10.9. However, we believe that this comparison is important to show the flexibility of our method. We performed a different experiment in order to present a comparison with equal conditions for both taggers. In this experiment, we transformed the Portuguese corpus so that it fits the classification setting of the hand-coded tagger. First, we removed from the corpus instances belonging to classes VALOR and TEMPO. These classes were removed because the hand-coded tagger does not consider them as NE. Then, instances from classes

| Class | Transformation | Description |
|---|---|---|
| PESSOA | PESSOA → PESSOA | remains unchanged |
| COISA | COISA → VARIADO | relabelled as VARIADO |
| VALOR | VALOR → ∅ | eliminated from corpus |
| ACONTECIMENTO | ACONTECIMENTO → VARIADO | relabelled as VARIADO |
| ORGANIZACAO | ORGANIZACAO → ORGANIZACAO | remains unchanged |
| OBRA | OBRA → VARIADO | relabelled as VARIADO |
| LOCAL | LOCAL → LOCAL | remains unchanged |
| TEMPO | TEMPO → ∅ | eliminated from corpus |
| ABSTRACCAO | ABSTRACCAO → VARIADO | relabelled as VARIADO |
| VARIADO | VARIADO → VARIADO | remains unchanged |

Table 10.10: Modifications of the Portuguese corpus to fit the classification setting of the hand-coded tagger.

| | | | | Attributes | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Hand-coded tagger | | | Internal | | | External | | | Internal & External | | |
| Category | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* | *P* (%) | *R* (%) | *F* |
| PESSOA | 35.6 | 72.3 | 0.477 | 86.7 | 91.0 | 0.888 | 48.9 | 72.3 | 0.583 | 87.3 | 91.0 | 0.891 |
| ORGANIZACAO | 41.8 | 37.8 | 0.397 | 84.4 | 89.4 | 0.868 | 47.3 | 44.5 | 0.459 | 82.2 | 87.0 | 0.845 |
| LOCAL | 68.0 | 17.2 | 0.274 | 85.4 | 82.7 | 0.840 | 56.3 | 51.2 | 0.536 | 79.9 | 79.9 | 0.799 |
| VARIADO | 0 | 0 | 0 | 90.0 | 77.3 | 0.832 | 31.7 | 12.6 | 0.180 | 83.6 | 70.7 | 0.766 |
| Overall | 36.3 | 31.8 | 0.287 | 86.6 | 85.1 | 0.857 | 46.0 | 45.1 | 0.440 | 83.3 | 82.1 | 0.825 |

Table 10.11: NEC performance on the modified Portuguese set.

COISA,ACONTECIMENTO,OBRA and ABSTRACCAO were relabelled as VARIADO, which is equivalent to class MISC. The remaining instances, belonging to classes PESSOA, ORGANIZACAO and LOCAL, were left unchanged.

In Table 10.10 we summarize the transformation process. Classification results of this experiment are presented in Table 10.11. These results are similar to those on Table 10.9. The hand-coded tagger had the lowest classification measures, reaching an *F* measure of 0.287; despite this poor behavior of the hand-coded tagger, we were able to improve NEC performance by a large margin, a combination of features yielded an F measure of 0.825. SVM trained on internal features attained the best results overall, although for class PESSOA the combination of internal and external features outperformed SVM trained only with internal features.

## 10.3 Final remarks

We are pleased to see the outcome of these experiments. Although the test set is small, we still consider these results very promising. We posed this problem as a machine learning task, then we trained a learning algorithm with the data available. Thus, a reasonable ex-

pectation of having more data available is that of expecting the classifier to learn better the target function, since for a learning algorithm the more data the better they will perform, provided the new data is not noisy.

We were able to reach excellent results on both NE tasks showing that our method can be applied to the task of NER on Portuguese and achieve high accuracies. We succeeded on our goal of increasing the coverage of a hand-coded named entity tagger in a different domain. The hand-coded system was developed for Spanish, then its coverage on Portuguese texts was very low. Nevertheless, by using our representation of the learning task, the coverage was increased tremendously, in some cases error reductions were as high as 80%; see classification measures for classes `VALOR`, `TEMPO` and `ABSTRACCAO` on Table 10.9. It is not surprising that internal features deliver better results in the majority of the cases, however the combination of features deliver competitive results. The important contribution from this work is that we can have the same method, using exactly the same representation, to perform NER on Spanish and Portuguese, without any manual tuning.

Our system entered the HAREM evaluation contest and it ranked #12 from 22 runs on the global results, and as high as #8 on the literary genre for NED.

Our design of the learning task has shown that it is possible to build good NE taggers without the need of complex and language-dependent features that are commonly used for NER. The method is flexible that we do not even need the hand-coded tagger: the internal features proved to be sufficient by themselves, leaving the use of a hand-coded tagger as optional.

An important characteristic of our method is its flexibility. We showed results proving that the method can be applied to a language other than Spanish with excellent results. Additionally, the method performed equally well on simulated speech transcripts, thus it is very flexible. Moreover, the method is flexible also regarding the classification setting of NE. Recall that the hand-coded tagger can only classify NE into a set of four categories. However, as the Portuguese data set has 10 different categories, it was unclear, at first, if this wider classification represented a problem for our method. This turned out not to be a problem, as it achieved impressively high accuracies. We can conclude that the method is not restricted in this respect, it can be applied to different categorizations of NE, regardless of the ones determined by the hand-coded tagger.

## Acknowledgements

We would like to thank the different reviewers of this chapter for their thoughtful comments and suggestions. We would also like to thank Nuno Cardoso and Diana Santos for their great job on this book.

This work was done while the first author was at the National Institute of Astrophysics, Optics and Electronics, Puebla, Mexico.