Portuguese Corpus-Based Learning Using ETL

Ruy Luiz Milidiú[1], Cícero Nogueira dos Santos[1] and Julio Cesar Duarte[1,2]

[1]Departamento de Informática,
Pontifícia Universidade Católica – PUC-Rio
Rua Marquês de São Vicente, 225, Gávea
Phone: +55 (21) 3527-1500
Cep 22453-900, Rio de Janeiro - RJ, Brazil
{milidiu | nogueira | jduarte}@inf.puc-rio.br

[2]Centro Tecnológico do Exército
Av. das Américas, 28705, Guaratiba
Phone: +55 (21) 2410-6200
Cep 23020-470, Rio de Janeiro - RJ, Brazil
jduarte@ctex.eb.br

Received 22 July 2008; accepted 30 November 2008

Abstract

We present Entropy Guided Transformation Learning models for three Portuguese Language Processing tasks: Part-of-Speech Tagging, Noun Phrase Chunking and Named Entity Recognition. For Part-of-Speech Tagging, we separately use the Mac-Morpho Corpus and the Tycho Brahe Corpus. For Noun Phrase Chunking, we use the SNR-CLIC Corpus. For Named Entity Recognition, we separately use three corpora: HAREM, MiniHAREM and LearnNEC06.

For each one of the tasks, the ETL modeling phase is quick and simple. ETL only requires the training set and no handcrafted templates. ETL also simplifies the incorporation of new input features, such as capitalization information, which are sucessfully used in the ETL based systems. Using the ETL approach, we obtain state-of-the-art competitive performance in all six corpora-based tasks. These results indicate that ETL is a suitable approach for the construction of Portuguese corpus-based systems.

Keywords: Entropy Guided Transformation Learning, transformation-based learning, decision trees, natural language processing.

1. Introduction

Since the last decade, Machine Learning (ML) has proven to be a very powerful tool to help in the construction of Natural Language Processing systems, which would otherwise require an unfeasible amount of time and human resources. When applying supervised learning schemes to language processing tasks, a corresponding annotated corpus is required. Corpus-based learning is a very attractive strategy, since it efficiently uses fast growing data resources [15, 13, 14, 22]. For the Portuguese language, many tasks have been approached using ML techniques, such as: Part-of-Speech Tagging [1, 10], Noun Phrase Chunking [27], Named Entity Recognition [20, 17], Machine Translation [19] and Text Summarization [11].

Portuguese tagged corpora is a scarce resource. Therefore, we focus on tasks where there are available corpora. Hence, we select the following three Portuguese Language Processing tasks: Part-of-Speech Tagging (POS), Noun Phrase Chunking (NP) and Named Entity Recognition (NER). These tasks have been considered fundamental for more advanced computational linguistic tasks [26, 33, 34, 32]. Observe that usually these three tasks are sequentially solved. First, we solve POS tagging. Next, using POS as an additional input feature, we solve NP chunking. Finally, using both the POS tags and NP chunks as additional input features, we solve NER.

In Table 1, we enumerate the six Portuguese corpora used throughout this work. For each corpus, we indicate its corresponding task and size. This work extends our previous findings on Portuguese Part-of-Speech Tagging [28].

Table 1. Corpus sizes.

Corpus	Task	Sentences	Tokens
Mac-Morpho	POS	53,374	1,221,465
Tycho Brahe	POS	40,932	1,035,592
SNR-CLIC	NP	4,392	104,144
HAREM	NER	8,142	165,102
MiniHAREM	NER	3,393	66,627
LearnNEC06	NER	2,100	44,835

For both the Mac-Morpho Corpus and the Tycho Brahe Corpus, the best result is reported by Milidiú et al. [28]. Their best non ETL approach is based on Transformation Based Learning. Their system shows 96.60% and 96.63% accuracy for the Mac-Morpho Corpus and the Tycho Brahe Corpus, respectively. For Portuguese Noun Phrase Chunking, a state-of-the-art system based in Transformation Based Learning is reported by Santos & Oliveira [27]. Applying their system to the SNR-CLIC Corpus, we achieve a $F_{\beta=1}$ of 87.85. For the HAREM Corpus, as far as the we know, there is no reported result using the same corpus configuration explored in this work. For the MiniHAREM Corpus, the best result is reported by Aranha [16]. Aranha reports a $F_{\beta=1}$ of 61.57 for the CORTEX system. The CORTEX system uses handcrafted rules that jointly work with a rich knowledge base for the NER task. For the LearnNEC06 Corpus, the best result is reported by Milidiú et al. [20]. Their best approach is based on Support Vector Machines (SVM). Their SVM system achieves a $F_{\beta=1}$ of 88.11. In Table 2, we summarize these performance results.

Table 2. System performances.

Corpus	State-of-the-art		
	Approach	Performance	ETL
Mac-Morpho	TBL	96.60	96.75
Tycho Brahe	TBL	96.63	96.64
SNR-CLIC	TBL	87.85	88.61
HAREM	-	-	63.27
MiniHAREM	CORTEX	61.57	63.04
LearnNEC06	SVM	88.11	87.71

In this work, we apply Entropy Guided Transformation Learning (ETL) to the six corpora-based tasks. ETL is a new ML strategy that combines the feature selection characteristics of Decision Trees (DT) and the robustness of Transformation Based Learning (TBL) [21]. The main purpose of ETL is to overcome the human driven construction of good template sets, which is a bottleneck on the effective use of the TBL approach. ETL produces transformation rules that are more effective than decision trees and also eliminates the need of a problem domain expert to build TBL templates. A detailed description and discussion of the ETL approach can be found in [21], where there is also an application to the multilanguage phrase chunking task.

Using the ETL approach, we obtain competitive performance results in all six corpora-based tasks. In Table 2, we show ETL performance. To assess the ETL modeling power, we also show that, using POS tags and NP chunks provided by ETL based systems, we can build a competitive ETL based NER system. Furthermore, ETL modeling is quick, since it only requires the training set and no handcrafted templates. The observed ETL training time for the Mac-Morpho corpus, using template evolution, is bellow one hour running on an Intel Centrino Duo 1.66GHz laptop. These results indicate that ETL is a competitive modeling approach for the construction of Portuguese language processing systems based on annotated corpus.

The remainder of this paper is organized as follows. In section 2, the ETL strategy is described. In section 3, we show how to design an efficient ETL model for each one of the six corpora-based tasks. Finally, in section 4, we present our concluding remarks.

2. Entropy Guided Transformation Learning

Information Gain, which is based on the data Entropy, is a key strategy for feature selection. The most popular Decision Tree learning algorithms [24, 31] implement this strategy. Hence, they provide a quick way to obtain entropy guided feature selection.

Entropy Guided Transformation Learning is a new machine learning strategy that combines the advantages of Decision Trees and Transformation-Based Learning [21]. The key idea of ETL is to use decision tree induction to obtain templates. Next, the TBL strategy is used to generate transformation rules. The ETL method is illustrated in the Figure 1.

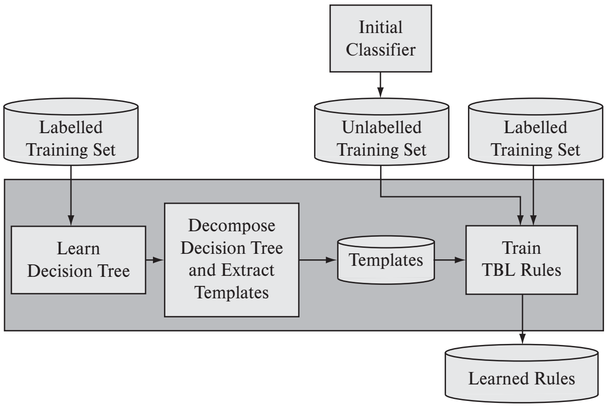

Figure 1. ETL - Entropy Guided Transformation Learning.

ETL method uses a very simple DT decomposition scheme to extract templates. The decomposition process includes a depth-first traversal of the DT. For each visited node, a new template is created by combining its parent node template with the feature used to split the data at that node. We use pruned trees in all experiments shown in section 3.

TBL training time is highly sensitive to the number and complexity of the applied templates. On the other hand, ETL provides a new training strategy that accelerates transformation learning. This strategy is based in an evolutionary template approach as described in [4]. The basic idea is to successively train simpler TBL models using subsets of the template set extracted from the DT. Each template subset only contains templates that include feature combinations up to a given tree level. In this way, only a few templates are considered at any point in time. Nevertheless, the descriptive power is not significantly reduced. We call this training strategy of Template Evolution.

The next two sections briefly review the DT learning algorithm and the TBL algorithm.

2.1. Decision Trees

Decision Tree learning is one of the most widely used machine learning algorithms. It performs a partitioning of the training set using principles of Information Theory. The learning algorithm executes a general to specific search of a feature space. The most informative feature is added to a tree structure at each step of the search. Information Gain Ratio, which is based on the data Entropy, is normally used as the informativeness measure. The objective is to construct a tree, using a minimal set of features, that efficiently partitions the training set into classes of observations. After the tree is grown, a pruning step is carried out in order to avoid overfitting.

One of the most used algorithms for DT induction is the C4.5 [24]. We use Quinlan's C4.5 system to obtain the required entropy guided selected features.

2.2. Transformation-Based Learning

Transformation Based error-driven Learning (TBL) is a successful machine learning algorithm introduced by Eric Brill [3]. It has since been used for several Natural Language Processing tasks, such as part-of-speech (POS) tagging [3, 5], noun-phrase and text chunking [25, 27], spelling correction [12], appositive extraction [7], and named entity extraction [6].

The TBL algorithm generates an ordered list of rules that correct classification mistakes in the training set, which have been produced by an initial classifier. The requirements of the algorithm are:

- two instances of the training set, one that has been correctly labeled, and another that remains unlabeled;
- an initial classifier, the *baseline system*, which classifies the unlabeled training set by trying to apply the correct class for each sample.
- a set of rule templates, which are meant to capture the relevant feature combinations that would determine the sample's classification.

The TBL algorithm can be formulated as follows:

1. The the baseline system is applied to unlabeled version of the training set, in order to obtain an initial classification;
2. The resulting classification is compared with the correct one and, whenever a classification error is found, all the rules that can correct it are generated by instantiating the templates. Usually, a new rule will correct some errors, but will also generate some other errors by changing correctly classified samples;
3. The rule scores (errors repaired - errors created) are computed. If there is not a rule with a score above an arbitrary threshold, the learning process is stopped;
4. The best scoring rule is selected, stored in the set of learned rules and applied to the training set;
5. Return to step 2.

When classifying a new sample item, the resulting sequence of rules is applied according to its generation order.

3. The Six Portuguese Corpora-Based Tasks with ETL

This section presents the application of the ETL approach to three Portuguese Language Processing tasks: part-of-speech tagging, noun phrase chunking and named entity recognition. We generate six different ETL systems, since we have six different corpora. For each one of them we report its performance. To highlight ETL learning power we also show the corresponding baseline system performance. In order to show that there is no significant performance loss when using ETL automatic templates, we also show the performance on the task of TBL with hand-crafted templates. Furthermore, we show the performance on the task of the state-of-the-art system.

The three tasks are modeled as token classification problems. For each token, its context is given by the features of its adjacent tokens. The number of adjacent tokens defines the size of what is called the context window. We have tried several context window sizes when modeling with ETL. In these models, WS=X subscript means that a context window of size X is used for the given model. For instance, $ETL_{WS=3}$ corresponds to ETL trained with a size three window, that is, the current token, the previous and the next ones.

3.1. Part-of-Speech tagging

Part-of-Speech (POS) tagging is the process of assigning a POS or another lexical class marker to each word in a text [9]. POS tags classify words into categories, based on the role they play in the context in which they appear. The POS tag is a key input feature for NLP tasks like phrase chunking and named entity recognition.

This section presents the application of the ETL approach to Portuguese POS tagging. We generate ETL systems for two Portuguese corpora: Mac-Morpho [2] and Tycho Brahe [18]. Table 3 shows some characteristics of these corpora. The Mac-Morpho Corpus is tagged with 22 POS tags, while the Tycho Brahe Corpus is tagged with 383 POS tags. The Tycho Brahe Corpus uses more POS tags because these tags also identify morphological aspects such as word number and gender. Each corpus is divided into training and test sets. These training and test set splits are the same as reported in [21].

Table 3. Part-of-Speech Tagging Corpora.

	Training Data		**Test Data**	
Corpus	Sentenc.	Tokens	Sentenc.	Tokens
Mac-Morpho	44,233	1007,671	9,141	213,794
Tycho Brahe	30,698	775,601	10,234	259,991

3.1.1. POS Tagging modeling

A word that appears in the training set is called a *known word*. Otherwise, it is called an *unknown word*. Our POS modeling approach follows the two stages strategy proposed by Brill [3]. First, morphological rules are applied to classify the unknown words. Next, contextual rules are applied to classify known and unknown words.

The morphological rules are based on the following token features:

- up to *c* characters long word prefixes and suffixes;
- specific character occurence in a word;
- adding (or subtracting) a *c* characters long prefix (or suffix) results in a known word;
- occurence of the word before (or after) a specific word *W* in a given long list of word bigrams. For instance, if the word appears after “to”, then it is likely to be a verb in the infinitive form.

In our experiments, we set the parameter *c* equal to 5.

With a very simple template set [3], one can effectively perform the morphological stage. For this stage, it is enough to use one feature or two feature templates. The one feature templates use one of the current token features. The two feature templates use one of the current token features and the current token POS.

The contextual rules use the context window features word and POS. We use the ETL strategy for learning contextual rules only.

3.1.2. ML Modeling

The following ML model configurations provide our best results.

BLS: The baseline system assigns to each word the POS tag that is most frequently associated with that word in the training set. If capitalized, an unknown word is tagged as a proper noun, otherwise it is tagged as a common noun.

ETL: The results for the TBL approach refer to the contextual stage trained using the lexicalized template set proposed in [3]. This template set uses combinations of words and POS tags in a context window of size 7.

TBL: In the ETL learning, we use the features word and POS. In order to overcome the sparsity problem, we only use the 200 most frequent words to induce the DT. In the DT learning step, the POS tag of the word is the one applied by the initial classifier (BLS). On the other hand, the POS tag of the neighbor words are the true ones. We report results for ETL trained with all the templates at the same time and also using template evolution.

3.1.3. Mac-Morpho Corpus

According to [28], a TBL system obtains state-of-the-art performance for the Mac-Morpho Corpus. Therefore, for the Mac-Morpho Corpus, we only report the performance of ETL, TBL and BLS systems.

In Table 4, we summarize the performance results of the three systems. The best ETL system uses a context window of size 7. Both ETL and TBL systems reduce the BLS system’s error in at least 64%. The $ETL_{WS=7}$ system’s accuracy is similar to the one of TBL. The $ETL_{WS=7}$ system’s accuracy, 96.75%, is equivalent to the best one reported so far for the Mac-Morpho Corpus.

Table 4. POS Tagging of the Mac-Morpho Corpus.

System	Accuracy (%)	# Templates
$ETL_{WS=7}$	96.75	72
TBL	96.60	26
BLS	90.71	–

Using the template evolution strategy, the training time is reduced in nearly 73% and there is no loss in the ETL$_{WS=7}$ system's performance. This is a remarkable reduction, since we use an implementation of the fastTBL algorithm [23] that is already a very fast TBL version. Training time is a very important issue when modeling a system with a corpus-based approach. A fast ML strategy enables the testing of different modeling options, such as different feature sets.

3.1.4. Tycho Brahe Corpus

According to [28], a TBL system obtains state-of-the-art performance for the Tycho Brahe Corpus. Therefore, for the Tycho Brahe Corpus, we only report the performance of ETL, TBL and BLS systems.

In Table 5, we summarize the performance results of the three systems. The best ETL system uses a context window of size 7. Both ETL and TBL systems reduce the BLS system's error in 62%. ETL and TBL systems achieved similar performance. Therefore ETL has state-of-the-art performance for the Tycho Brahe Corpus.

Using the template evolution strategy, the training time is reduced in nearly 63% and there is no loss in the ETL$_{WS=7}$ system's performance.

Table 5. POS Tagging of the Tycho Brahe Corpus.

System	Accuracy (%)	# Templates
ETL$_{WS=7}$	**96.64**	43
TBL	96.63	26
BLS	91.12	–

3.2. Noun Phrase Chunking

Noun Phrase Chunking consists in recognizing text segments that are Noun Phrases (NP). NP chunking provides a key feature that helps on more elaborated NLP tasks such as parsing and information extraction. In the example that follows, we use brackets to indicate the four noun phrase chunks in the sentence.

[He] reckons [the current account deficit]
will narrow to [only # 1.8 billion] in [September]

This section presents the application of the ETL approach to Portuguese NP Chunking. We use the SNR-CLIC Corpus described in [8]. Table 6 shows some characteristics of this corpus, which is tagged with both POS and NP chunk tags.

Table 6. Noun Phrase Chunking Corpus.

Corpus	Sentences	Tokens	Noun Phrases
SNR-CLIC	4,392	10,4144	17,795

3.2.1. NP Chunking Modeling

We approach the NP Chunking as a token classification problem, in the same way as in the CONLL-2000 shared task [26]. We use the *IOB*1 tagging style, where: *O*, means that the word is not a NP; *I*, means that the word is part of a NP and *B* is used for the leftmost word of a NP beginning immediately after another NP. The tagging style is shown in the following example.

He/I reckons/O the/I current/I account/I deficit/I
will/O narrow/O to/O only/I #/I 1.8/I billion/I
in/O September/I

3.2.2. ML Modeling

The following ML model configurations provide our best results.

BLS: The baseline system assigns to each word the NP tag that was most frequently associated with the part-of-speech of that word in the training set. The only exception was the initial classification of the prepositions, which is done on an individual basis: each preposition has its frequency individually measured and the NP tag is assigned accordingly, in a lexicalized method.

ETL: In the TBL system we use a template set that contains the templates proposed by Ramshaw & Marcus [25] and the set of six special templates proposed by Santos & Oliveira [27]. The Ramshaw & Marcus's template set contains 100 handcrafted templates which make use of the features *word*, POS and *NP tags*, and use a context window of seven tokens. The Santos & Oliveira's six templates are designed to reduce classification errors of preposition within the task of Portuguese noun phrase chunking. These templates use special handcrafted constraints that allow to efficiently check the feature *word* in up to 20 left side adjacent tokens.

TBL: In the ETL learning, we use the features word, *POS* and *NP tags*. Additionally, we introduce the feature *left verb*. This feature assumes the word feature value of the nearest predecessor verb of the current token. In the DT learning step: only the 200 most frequent words are used; the NP tag of the word is the one applied by the initial classifier; and, the NP tag of neighbor words are the true ones. We report results for ETL trained with all the templates at the same time as well as using template evolution.

We use 10-fold cross-validation to assess the performance of the trained systems.

3.2.3. SNR-CLIC Corpus

According to [27], the TBL system configuration used here obtains state-of-the-art performance for Portuguese Noun Phrase Chunking. Therefore, for the SNR-CLIC Corpus, we only report the performance of ETL, TBL and BLS systems.

In Table 7, we summarize the performance results of the three systems. The best ETL system uses a context window of size 11. The ETL$_{WS=11}$ system achieves a F$_{\beta=1}$ of 88.61, which is equivalent to the best result reported so far for a 10-fold cross-validation using the SNR-CLIC Corpus. ETL$_{WS=11}$ increases the BLS system's F$_{\beta=1}$ by 30.6%. The F$_{\beta=1}$ of the ETL$_{WS=11}$ system is similar to the one of TBL. These results indicate that the templates obtained by entropy guided feature selection are very effective to learn transformation rules for this task.

Table 7. Portuguese noun phrase chunking.

System	**Acc.** (%)	**Prec.**(%)	**Rec.** (%)	**F$_{\beta=1}$**	**# T**
TL$_{WS=11}$	**97.88**	**88.32**	**88.90**	**88.61**	55
TBL	97.66	87.32	88.38	87.85	106
BLS	92.54	62.84	73.78	67.87	-

Using the template evolution strategy, the training time is reduced in nearly 51%. On the other hand, there is a decrease of 0.6 in the ETL system's F$_{\beta=1}$.

In the application of ETL to the SNR-CLIC corpus, we can notice one of the ETL advantages. Using ETL, it is possible to explore larger context window sizes without being concerned with the number of feature combinations that it could produce. In the case of NP chunking, where we use four features per token, a context window of size 11 results in 44 candidate features. The manual creation of feature combinations in such a case would be a hard task. In [27] a special type of template unit is introduced, the *constrained atomic term*, that enables TBL to check a feature in a wide context window. On the other hand, this kind of template unit requires the creation of a task specific handcrafted constraint. Furthermore, when creating a complete template, it is necessary to combine this template unit with other contextual features.

3.3. Named Entity Recognition

Named Entity Recognition (NER) is the problem of finding all proper nouns in a text and to classify them among several given categories of interest or to a default category called Others. Usually, there are three given categories: Person, Organization and Location. Time, Event, Abstraction, Thing, and Value are some additional, but less usual, categories of interest. In the example that follows, we use brackets to indicate the four Named Entities in the sentence.

[*PER* Wolff], currently a journalist in [*LOC* Argentina], played with [PER Del Bosque] in the final years of the seventies in [ORG Real Madrid]

This section presents the application of the ETL approach to Portuguese NER. We evaluate the performance of ETL over three Portuguese corpora: the HAREM Corpus, the MiniHAREM Corpus and the LearnNEC06 Corpus. Table 8 shows some characteristics of these corpora.

Table 8. Named Entity Recognition Corpora.

Corpus	**Sentences**	**Tokens**	**Named Entities**
HAREM	8,142	16,5102	8,624
MiniHAREM	3,393	66,627	3,641
LearnNEC06	2,100	44,835	3,325

The HAREM Corpus is a golden set for NER in Portuguese [29]. This corpus is annotated with ten named entity categories: Person (PESSOA), Organization (ORGANIZACAO), Location (LOCAL), Value (VALOR), Date (TEMPO), Abstraction (ABSTRACCAO), Title (OBRA), Event (ACONTECIMENTO), Thing (COISA) and Other (VARIADO). Additionally, we automaticaly generate two input features. Using the ETL$_{WS=7}$ POS Tagger trained with the Mac-Morpho Corpus we generate the POS feature. Using the ETL$_{WS=11}$ NP Chunker trained with the SNR-CLIC Corpus we generate the Noun Phrase feature.

The MiniHAREM Corpus is a subset of the HAREM Corpus [29].

The LearnNEC06 Corpus [20] is annotated with only three categories: Person, Organization and Location. This corpus is already annotated with golden POS tags and golden noun phrase chunks.

3.3.1. NER Modeling

We approach the NER task as a token classification problem, in the same way as in the CONLL-2002 shared task [33]. We use the *IOB*1 tagging style, where: *O*, means that the word is not a NE; *I* - *XX*, means that the word is part of NE type *XX* and *B* - *XX* is used for the leftmost word of a NE beginning immediately after another NE of the same type. The *IOB*1 tagging style is shown in the following example.

Wolff/I-PER ,/O currently/O a/O journalist/O in/O
Argentina/O ,/O played/O with/O Del/I-PER
Bosque/I-PER in/O the/O final/O years/O of/O
the/O seventies/O in/O Real/I-ORG Madrid/I-ORG

The NER task can be subdivided [29] into two subtasks: *identification*, where the objective is the correct delimitation of named entities; and *classification*, where the objective is to associate categories to identified named entities. For the HAREM Corpus, we carry out experiments for these two subtasks. However, in the classification task, we use ETL to classify only the five most frequent categories: Person, Organization, Location, Date and Value.

We apply a 10-fold cross-validation to assess the trained models efficacy. For each experiment, we only report the ETL model that gives the best cross-validation result. In the HAREM Corpora experiments, we use the evaluation tools described in [29].

3.3.2. ML Modeling

The following ML model configurations provide our best results.

BLS: For the LearnNEC06 Corpus, we use the same baseline system proposed in [20], which makes use of location, person and organization gazetteers, as well as some simple heuristics. for the HAREM and MiniHAREM corpora, we apply a BLS that makes use of gazetteers only. We use the gazetteers presented in [20], as well as some sections of the REPENTINO gazetteer [30]. From the REPENTINO gazetteer we use only the categories Beings, Location and Organization. We use only some subcategories of the REPENTINO gazetteer. From category Beings we use subcategory Human. From Location we use Terrestrial, Town, Region and Adm. Division. From Organization we use all subcategories. For some subcategories an extra processing is also required. From Human, we extract only first names. From the Organization category, we use full company names and extract the top 100 most frequent words. We also use a month name gazetteer and a list of lower case words that can start a NE

ETL: In the ETL learning, we use the basic features *word*, *pos*, *noun phrase tags* and *ne tags*. Additionally, we introduce two new features: *capitalization information* and *dictionary membership*. The capitalization information feature provides a token classification, assuming one the following categorical values: First Letter is Uppercase, All Letters are Uppercase, All Letters are Lowercase, Number, Punctuation, Number with "/" or "-" inside, Number ending with *h* or *hs* or Other. Similarly, the dictionary membership feature assumes one the following categorical values: Upper, Lower, Both or None. In the DT learning step, only the 100 most frequent words are used, the named entity tag of the word is the one applied by the initial classifier, and the named entity tags of neighbor words are the true ones.

TBL: The reported results for the TBL approach refer to TBL trained with the 32 handcrafted template set proposed in [20].

In order to improve the ETL based NER system efficacy for the HAREM and MiniHAREM Corpora, we use the LearnNEC06 Corpus as an extra training set for the learning of the three categories: Person, Organization and Locations. In this case, we train the ETL based NER system using a two phase strategy: (1) first, we train an ETL classifier for the categories Person, Organization and Locations, using the two corpora; (2) then, we train a second ETL classifier for the five categories: Person, Organization, Location, Date and Value, using either the HAREM or MiniHAREM Corpus. The second ETL classifier uses the first one as the initial classifier (baseline system). It is important to note that in the experiments using the HAREM Corpus, the LearnNEC06 Corpus is included only in the training set of each cross-validation fold. The increase in the training corpus produced better results for the classification task only.

3.3.3. HAREM Corpus

As in [29], we report the category classification results in two scenarios: Total and Selective. In the *Total Scenario*, all the categories are taken into account when scoring the systems. In the *Selective Scenario*, only the five chosen categories (Person, Organization, Location, Date and Value) are taken into account.

In Table 9, we summarize the 10-fold cross-validation performance results of ETL, TBL and BLS systems for the Identification task. ETL$_{WS=5}$ system doubles the BLS's F$_{\beta=1}$ and is expressively better than TBL using handcrafted templates. The TBL result is very poor due to its template set, which is the same used with the LearnNEC06 Corpus proposed in [20]. This is the only TBL template set that we have found for the NER task. Since this template set contains some pre-instantiated tests, apparently it is very restricted to the kind of named entity structures that appear in the LearnNEC06 Corpus.

Table 9. NE Identification for the HAREM Corpus.

System	# T	Acc. (%)	Prec. (%)	Rec. (%)	F$_{\beta=1}$
ETL$_{WS=5}$	143	**97.87**	**82.55**	84.11	**83.32**
TBL	32	95.62	64.79	56.78	60.52
BLS	–	92.10	54.25	34.03	41.82

In the Identification task, the use of the template evolution strategy reduces the training time in nearly 70% and maintains the same ETL system efficacy.

In Tables 10 and 11, we summarize the 10-fold cross-validation performance results of the systems for the Category Classification task. In both scenarios Total and Selective, the $ETL_{WS=5}$ almost doubles the BLS's $F_{\beta=1}$. In the Total Scenario, the $F_{\beta=1}$ of the $ETL_{WS=5}$ classifier is 13.67 higher than the one of TBL. In the classification task, the use of the template evolution strategy reduces the training time in nearly 70%. On the other hand, there is a reduction of 1.05 in the $F_{\beta=1}$.

Table 12 shows the ETL results, broken down by named entity type, for the classification task using the HAREM Corpus.

Table 10. Category classification in the Total Scenario for the HAREM Corpus.

System	# T	Acc. (%)	Prec. (%)	Rec. (%)	$F_{\beta=1}$
$ETL_{WS=5}$	160	**94.85**	**71.49**	**56.75**	**63.27**
TBL	32	93.60	55.60	44.77	49.60
BLS	-	92.10	46.19	29.00	35.63

Table 11. Category classification in the Selective Scenario for the HAREM Corpus.

System	# T	Acc. (%)	Prec. (%)	Rec. (%)	$F_{\beta=1}$
$ETL_{WS=5}$	160	96.48	71.35	68.74	70.02
TBL	32	95.52	55.59	54.24	54.91
BLS	-	94.13	46.18	35.13	39.90

Table 12. ETL results by entity type for the HAREM Corpus.

	Precision (%)	Recall (%)	$F_{\beta=1}$
Location	73.02	68.19	70.52
Organization	58.94	60.55	59.73
Person	74.44	67.40	70.75
Date	81.45	82.89	82.17
Value	75.99	68.83	72.23
Overall	71.35	68.74	70.02

3.3.4. MiniHAREM Corpus

The ETL and TBL systems shown in this section are trained using the HAREM Corpus part that does not include the MiniHAREM Corpus. We assess the performances of these systems by applying each one to the MiniHAREM Corpus. As in the previous subsection, we report the category classification results in two scenarios: Total and Selective.

For the MiniHAREM Corpus, the CORTEX system [16] shows the best result reported so far. The CORTEX system relies in the use of handcrafted rules that jointly work with a rich knowledge base for the NER task [16]. Here, we also list the CORTEX system performance reported in [16].

In Table 13, we summarize the performance results of the four systems for the Identification task. For this task, the CORTEX system is the best one. As we can see in Table 12, $ETL_{WS=5}$ system doubles the BLS's $F_{\beta=1}$ and is expressively better than TBL using handcrafted templates.

In Tables 14 and 15, we summarize the performance results of the four systems for the Category Classification task. In both scenarios Total and Selective, $ETL_{WS=5}$ achieves the best $F_{\beta=1}$. Therefore, as far as we know, ETL results are the best reported so far for Category Classification of the MiniHAREM Corpus. This is a remarkable result, since the $ETL_{WS=5}$ system does not use any handcrafted rules or templates.

In Table 16, we show the $ETL_{WS=5}$ system results, broken down by named entity type, for the MiniHAREM Corpus.

Table 13. NE Identification for the MiniHAREM Corpus.

System	Recision (%)	Recall (%)	$F_{\beta=1}$
CORTEX	79.77	87.00	83.23
$ETL_{WS=5}$	81.51	82.42	81.96
TBL	65.14	56.12	60.29
BLS	54.26	32.33	40.52

Table 14. Category classification in the Total Scenario for the MiniHAREM Corpus.

System	Precision (%)	Recall (%)	$F_{\beta=1}$
$ETL_{WS=5}$	71.73	**56.23**	**63.04**
CORTEX	**77.85**	50.92	61.57
TBL	57.78	45.20	50.72
BLS	47.87	28.52	35.74

Table 15. Category classification in the Selective Scenario for the MiniHAREM Corpus.

System	Precision	Recall	$F_{\beta=1}$
$ETL_{WS=5}$	71.53	**68.04**	**69.74**
CORTEX	**77.86**	60.97	68.39
TBL	57.76	54.69	56.19
BLS	47.85	34.52	40.11

Table 16. ETL results by entity type for the MiniHAREM Corpus.

	Precision (%)	Recall (%)	$F_{\beta=1}$
Location	77.65	67.48	72.21
Organization	55.45	60.01	57.64
Person	75.45	67.62	71.32
Date	79.41	80.29	79.85
Value	73.82	65.27	69.28
Overall	71.53	68.04	69.74

Like any other ML based strategy, one advantage of ETL based systems over other non-ML based systems, such as CORTEX, is its versatility. All the used resources, but the training set and gazetteers, are language independent. We can quickly create an ETL based NER system for any language that has an available training set. Nowadays, we can find NER training sets [33, 34] for many languages, such as: Dutch, English, German, Hindi and Spanish.

3.3.5. LearnNEC06 Corpus

According to [20], the SVM algorithm obtains state-of-the-art performance for the LearnNEC06 Corpus. Therefore, for the LearnNEC06 Corpus, we also list the SVM system performance reported in [20].

In Table 17, we summarize the performance results of the four systems. The best ETL system uses a context window of size 9. Both $ETL_{WS=9}$ and TBL systems increase the BLS system's $F_{\beta=1}$ by at least 15%. The $ETL_{WS=9}$ system slightly outperforms the TBL system. $ETL_{WS=9}$ results are very competitive with the ones of SVM.

Table 17. NER of the LearnNEC06 Corpus.

System	Acc. (%)	Prec. (%)	Rec. (%)	$F_{\beta=1}$	# T
SVM	98.83	86.98	89.27	88.11	-
$ETL_{WS=9}$	98.80	86.89	88.54	87.71	102
TBL	98.79	86.65	88.60	87.61	32
BLS	97.77	73.11	80.21	76.50	-

Using the template evolution strategy, the training time is reduced in nearly 20% and the learned transformation rules maintain the same performance. In this case, the training time reduction is not very significant, since the training set is very small.

Although the ETL's $F_{\beta=1}$ is only slightly better than the one of TBL with handcrafted template, it is an impressive achievement, since the handcrafted template set used in [20] contains many pre-instantiated rule tests that carry a lot of domain specific knowledge.

In Table 18, we show the $ETL_{WS=9}$ system results, broken down by named entity type, for the LearnNEC06 Corpus.

Table 18. ETL results by entity type for the LearnNEC06 Corpus.

	Precision (%)	Recall (%)	$F_{\beta=1}$
Location	93.96	81.78	87.45
Organization	84.00	89.77	86.79
Person	85.75	91.93	88.73
Overall	86.89	88.54	87.71

4. Conclusions

We present Entropy Guided Transformation Learning models for three Portuguese Language Processing tasks. Six different annotated Portegese corpora are used: Mac-Morpho, Tycho Brahe, SNR-CLIC, HAREM, MiniHAREM and LearnNEC06.

ETL modeling is simple. It only requires the training set and no handcrafted templates. ETL also simplifies the incorporation of new input features such as capitalization information, which are sucessfully used in the ETL based NER systems. The critical learning resource is the annotated corpus. Therefore, the availability of new Portuguese annotated corpora, combined with ETL modeling, is an effective strategy to advance Portuguese language processing. ETL is versatile, since we are able to solve the NER problem using ETL systems only.

ETL training is reasonably fast. The observed ETL training time for the Mac-Morpho Corpus, using template evolution, is bellow one hour running on an Intel Centrino Duo 1.66GHz laptop.

Using the ETL approach, we obtain state-of-the-art competitive performance in all six corpora-based tasks. These results indicate that ETL is a suitable approach for the construction of Portuguese corpus-based systems.

References

[1] R. V. X. Aires, S. M. Aluísio, D. C. S. Kuhn, M. L. B. Andreeta, O. N. Oliveira-Jr. Combining Classifiers to Improve Part of Speech Tagging: A Case Study for Brazilian Portuguese. In *Proceedings of IBERAMIA-SBIA*, pages 227-236, 2000.

[2] S. M. Aluísio, J. M. Pelizzoni, A. R. Marchi, L. Oliveira, R. Manenti, V. Marquiafável. An Account of the Challenge of Tagging a Reference Corpus for Brazilian Portuguese. In *Proceedings of PROPOR*, Faro, pages 110-117, 2003.

[3] E. Brill. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Comput. Linguistics.* 21(4):543-565, 1995.

[4] J. R. Curran, R. K. Wong. Formalisation of Transformation-based Learning. In *Proceedings of the ACSC*, Canberra, Australia, pages 51-57, 2000.

[5] M. Finger. Técnicas de Otimização da Precisão Empregadas no Etiquetador Tycho Brahe. In *Proceedings of PROPOR*, São Paulo, pages 141-154, 2000.

[6] Radu Florian. Named Entity Recognition as a House of Cards: Classifier Stacking. In *Proceedings of 4th conference on Computational Natural Language Learning - CONLL*, pages 175-178, 2002.

[7] M. C. Freitas, J. C. Duarte, C. N. dos Santos, R. L. Milidiú, R. P. Renteria, V. Quental. A Machine Learning Approach to the Identification of Appositives. In *Proceedings of Ibero-American AI Conference*, Ribeirão Preto, 2006.

[8] M. C. Freitas, M. Garrao, C. Oliveira, C. N. dos Santos, M. Silveira. A anotação de um corpus para o aprendizado supervisionado de um modelo de SN. In *Proceedings of the III TIL / XXV Congresso da SBC*, São Leopoldo, 2005.

[9] D. Jurafsky, J. H. Martin. Speech and Language Processing. Prentice Hall, 2000.

[10] F. N. Kepler, M. Finger. Comparing Two Markov Methods for Part-of-Speech Tagging of Portuguese. In *Proceedings of IBERAMIA-SBIA*, Ribeirão Preto, pages 482-491, 2006.

[11] D. S. Leite, L. H. M. Rino. Combining Multiple Features for Automatic Text Summarization through Machine Learning. In *Proceedings of PROPOR*, Aveiro, Portugal, pages 122-132, 2008.

[12] L. Mangu, E. Brill. Automatic Rule Acquisition for Spelling Correction. In *Proceedings of The Fourteenth ICML*, São Francisco, pages 187-94, 1997.

[13] European Language Resources Association. http://catalog.elra.info/, Sept 24, 2008.

[14] Linguateca. www.linguateca.pt/, Sept 24, 2008.

[15] Linguistic Data Consortium. www.ldc.upenn.edu/, Sept 24, 2008.

[16] C. N. Aranha. Reconhecimento de entidades mencionadas em português. *O Cortex e a sua participação no HAREM*, Linguateca, Portugal, 2007.

[17] O. Ferrández, Z. Kozareva, A. Toral, R. Muñoz, A. Montoyo. Reconhecimento de entidades mencionadas em português, *Tackling HAREM's Portuguese Named Entity Recognition task with Spanish resources*, Linguateca, Portugal, 2007.

[18] IEL-UNICAMP; IME-USP. Corpus Anotado do Português Histórico Tycho Brahe. http://www.ime.usp.br/tycho/corpus/, Jan 23, 2008.

[19] R. S. Martnez, J. P. Neto, D. Caseiro. Statistical Machine Translation of Broadcast News from Spanish to Portuguese. In *Proceedings of PROPOR*, Aveiro, Portugal, pages 112-121, 2008.

[20] R. L. Milidiú, J. C. Duarte, R. Cavalcante. Machine learning algorithms for portuguese named entity recognition. In *Proceedings of Fourth Workshop in Information and Human Language Technology*, Ribeirão Preto, 2006.

[21] R. L. Milidiú, C. N. dos Santos, J. C. Duarte. Phrase Chunking using Entropy Guided Transformation Learning. In *Proceedings of ACL2008*, Columbus, Ohio, 2008.

[22] The Lacio Web Project. www.nilc.icmc.usp.br/lacioweb/ferramentas.htm, Jan 23, 2008.

[23] G. Ngai, R. Florian. Transformation-Based Learning in the Fast Lane. In *Proceedings of North Americal ACL*, pages 40-47, June 2001.

[24] J. R. Quinlan. C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, 1993.

[25] L. Ramshaw, M. Marcus. Text Chunking Using Transformation-Based Learning. In *Proceedings of* S. Armstrong, K. W. Church, P. Isabelle, S. Manzi, E. Tzoukermann, D. Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, Kluwer, 1999.

[26] E. F. T. K. Sang, S. Buchholz. Introduction to the CoNLL-2000 shared task: chunking. In *Proceedings of the 2nd workshop on Learning language in logic and the 4th CONLL*, Morristown, USA, pages 127-132, 2000.

[27] C. N. dos Santos, C. Oliveira. Constrained Atomic Term: Widening the Reach of Rule Templates in Transformation Based Learning. *EPIA*, Covilhã, Portugal, pages 622-633, 2005.

[28] C. N. dos Santos, R. L. Milidiú, R. P. Rentera. Portuguese Part-of-Speech Tagging Using Entropy Guided Transformation Learning. In *Proceedings of PROPOR*, Aveiro, Portugal, pages 143-152, 2008.

[29] D. Santos, N. Cardoso. Reconhecimento de entidades mencionadas em português. Linguateca, Portugal, 2007.

[30] L. Sarmento, A. Sofia, L. Cabral. REPENTINO - A Wide-Scope Gazetteer for Entity Recognition in Portuguese. In *Proceedings of 7th Workshop on Computational Processing of Written and Spoken Portuguese*, Itatiaia, pages 31-40, 2006.

[31] J. Su, H. Zhang. A Fast Decision Tree Learning Algorithm. *AAAI*, University of New Brunswick, NB, Canadá, 2006.

[32] M. Surdeanu, R. Johansson, A. Meyers, L. Màrquez, J. Nivre. The CoNLL 2008 Shared Task on Joint Parsing of Syntactic and Semantic Dependencies. *CoNLL 2008.* In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, Coling 2008 Organizing Committee, Manchester, England, pages 159-177, 2008.

[33] T. K. Sang, F. Erik. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of CoNLL-2002*, Taipei, Taiwan, pages 155-158, 2002.

[34] T. K. Sang, F. Erik, F. De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In Walter Daelemans and Miles Osborne, editors, In *Proceedings of CoNLL-2003*, Edmonton, Canada, pages 142-147, 2003.