HAL
open science

Evaluating the Impact of External Lexical Resources into a CRF-based Multiword Segmenter and Part-of-Speech Tagger

Mathieu Constant, Isabelle Tellier

To cite this version:

Mathieu Constant, Isabelle Tellier. Evaluating the Impact of External Lexical Resources into a CRF-based Multiword Segmenter and Part-of-Speech Tagger. 8th International Conference on Language Resources and Evaluation (LREC'12), May 2012, Turkey. pp.646-650. hal-00790624

HAL Id: hal-00790624

https://hal.science/hal-00790624v1

Submitted on 20 Feb 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evaluating the Impact of External Lexical Resources into a CRF-based Multiword Segmenter and Part-of-Speech Tagger

Matthieu Constant*, Isabelle Tellier**

*Université Paris-Est, LIGM, CNRS, ** Université Paris 3 - sorbonne Nouvelle, LaTTiCe, CNRS
mconstan@univ-mlv.fr, isabelle.tellier@univ-paris3.fr

Résumé

This paper evaluates the impact of external lexical resources into a CRF-based joint Multiword Segmenter and Part-of-Speech Tagger. We especially show different ways of integrating lexicon-based features in the tagging model. We display an absolute gain of 0.5% in terms of f-measure. Moreover, we show that the integration of lexicon-based features significantly compensates the use of a small training corpus.

Keywords: Multiword Expressions & Collocations, Part Of Speech tagging, Statistical and machine learning methods, Conditional Random Fields, Morphosyntactic lexicons, lexical segmentation.

1. Introduction

Coupling external lexicons with an annotated corpus to train discriminative models is a recent trend in Natural Language Processing, e.g. (McCallum and Li, 2003) for Named Entity Recognition. Some studies have shown that it can significantly improve the accuracy of Part-of-Speech (POS) Tagging – e.g. (Denis and Sagot, 2009) for French – as it helps to deal with unknown words [1]. Similarly, Constant and Sigogne (2011) and Constant et al. (2011) described different methods to use external Multiword Unit (MWU) lexicons in order to improve the CRF [2]-based joint task of MWU segmentation and POS tagging. They show that it helps to recognize unknown segments. Neverthess, the different methods are sometimes hardly comparable because experiments have been conducted in different environments [3].

The objective of this paper is four-fold :

- synthesize the different possible methods for coupling external lexicons with an annotated corpus for the joint task of MWU segmentation and POS tagging ;
- evaluate them in a uniform environment on two different versions of the reference annotated corpus ;
- compare with a sequential approach (segmentation followed by tagging) ;
- release a fully parameterized LGLPL-licensed software resource 'lgtagger' (http ://igm.univ-mlv.fr/~mconstan/research/software) [4].

The first section describes the task and the resources used. We then present the different methods for coupling these resources to train a single linear CRF model. In the last section, we detail the experiments undertaken and comment the results. All experiments were carried out on French.

1. Words that are absent of the training corpus.
2. Conditional Random Fields (Lafferty et al., 2001)
3. Different CRF training softwares, different versions of the training corpus
4. Some CRF features implemented in this tool were directly inspired by the ones in the tool SEM : http ://www.univ-orleans.fr/lifo/Members/Isabelle.Tellier/SEM.html.

2. Task and resources

We describe here the joint task of MWU segmentation and POS tagging, and the resources we have used. We take advantage of the fact that French is a language for which various MWU lexicons are available, as well as a fully POS-annotated corpus where MWUs are marked.

2.1. Joint MWU Segmentation and POS tagging

Our joint task consists in segmenting and labelling lexical units including multiword ones. By using an `IOB` [5] scheme (Ramshaw and Marcus, 1995), it is equivalent to labelling simple tokens. Each token is labeled by a tag of the form `X+B` or `X+I`, where `X` is the POS label of the lexical unit and the suffix `B` indicates that the token is at the beginning of the lexical unit, while the suffix `I` indicates an internal position. Suffix `O` is useless as the end of a lexical unit corresponds to the beginning of another one (suffix `B`) or the end of a sentence. Such a procedure therefore determines lexical unit limits, as well as their POS. For instance,

Quant	PREP+B
à	PREP+I
la	DET+B
technique	CN+B
,	PUNCT+B
son	DET+B
verdict	CN+B
est	V+B
implacable	ADJ+B
.	PUNCT+B

(Concerning the technique, its verdict is implacable)

2.2. French Treebank

The French Treebank (FTB) (Abeillé et al., 2003) is a syntactically annotated corpus made of journalistic texts from *Le Monde* newspaper. We used two different versions : (i) the 2005 version ; (ii) the recent version dedicated to tagging and parsing experiments (Candito and Crabbé., 2009). We have uniformized the POS tagsets for both versions

5. I : Inside (segment) ; O : Outside (segment) ; B : Beginning (of segment)

with 29 tags. Version (i) is composed of 19,490 sentences and 569,080 units including 32,975 multiword ones (i.e. 5.8%). Version (ii) is smaller and has a lower proportion of MW units : it contains 12,351 sentences and 350,931 units including 10,785 multiword units (i.e. 3.1%). MW units are of different types : compound words – e.g. *parce que* (because), *flambant neuf* (brand new), *dans l' immédiat* (right now) – and named entities – e.g. *Europe de l'Est* (Eastern Europe) , *Jean-Pierre* (John Peter) –.

2.3. Lexical resources

The lexical resources used are of two types : morphological electronic dictionaries and strongly lexicalized local grammars. They are freely available in the software 'lgtagger' under LGPL-LR license. Morphological electronic dictionaries are lists of lexical entries of simple and compound forms. They contain : the general-language dictionary DELA of 976,000 entries (Courtois, 1990; Courtois et al., 1997), the general-language dictionary Lefff of 579,000 entries (Sagot, 2010) and the toponym dictionary Prolex of 123,000 entries (Piton et al., 1999). Strongly lexicalized local grammars are factorized sets of multiword entries in the form of finite-state graphs (Gross, 1997). For the purpose of this paper, we used 42 graphs recognizing numerical determiners and some Named Entities like organization or location names.

In our experiments, we concatenated the information coming from all these resources, whitout preserving the memory of their origin.

3. Coupling external lexicons and annotated corpus to train a CRF model

Linear CRFs are the best current statistical models to learn to annotate sequences. Their interest also relies in the fact that they allow to take into account a great number of various features. The features can be computable intrinsic properties of the data as well as external information (McCallum and Li, 2003). We detail different solutions to integrate them.

3.1. Linear CRF Models

Linear chain Conditional Ramdom Fields (CRF) are discriminative probabilistic models introduced by (Lafferty et al., 2001) for sequential labelling. Given an input sequence of tokens $x = (x_1, x_2, ..., x_N)$ and an output sequence of labels $y = (y_1, y_2, ..., y_N)$, the model is defined as follows :

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{N} \exp\Big(\sum_{k=1}^{K} \lambda_k f_k(t, x, y_t, y_{t-1}) \Big)$$

where $Z(x)$ is a normalization factor depending on x. It is based on K features, each of them being a binary function f_k depending on the current position t in x, the current label y_t, the preceding one y_{t-1} and the whole input sequence x. This means that any computable property x^i of x can be taken into account in features : the lexical value of x at any position (not only at position t), whether this value begins with an upper case, contains a number, etc. The feature is activated (i.e. $f_k(t, y_t, y_{t-1}, x) = 1$) if a given configuration

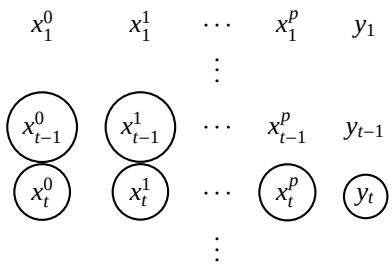

FIGURE 1 – A File of Labeled Examples with an instanciated Feature Template

concerning t, y_t, y_{t-1} and x is satisfied. Each feature f_k is associated with a weight λ_k. The weights are the parameters of the model, to be estimated. The features belong to the model and should be provided by the users. But softwares implementing CRF models help users : they usually only require to provide *feature templates* which are automatically instanciated into as many features as positions in the training data they can apply. We detail this point in the following subsection, because it is crucial to understand how this mechanism allows to integrate external information.

3.2. Feature templates

In CRF++ [6] and Wapiti [7], some of the best known (and, in the case of Wapiti, most efficient) implementations of linear CRFs, training data (x, y) must be provided in files of the form of Figure 1.

In this kind of "tabular-like" files, each line corresponds to a position t in a sentence. Sequences of tokens (i.e. labeled sentences) are just consecutive lines, two distinct sentences being separated by a blank line. At a position t, the first $p+1$ columns display the computable properties x_t^i, $0 \leq i \leq p$ of x_t which can be taken into account in features, and the last one contains the value y_t of the correct label.

Figure 1 also illustrates the notion of template. A template can be seen as a configuration of holes which can be positioned on any line of the file. At any given position, it is able to select the not-empty values of the files that appear in its holes. Each of these positions produces a feature, as a conjonction of all observed values. Of course, to respect the constraints of linear CRFs, the hole configurations can take any form on the first $p + 1$ columns, but can only integrate a unique value y_t ("unary features") or a sequence of two consecutive values y_{t-1} and y_t ("binary features") on the last one.

The template of Figure 1, applied on the example sentence of section 2.1. at the position $t = 6$ where x_t^0 is the lexical value of x at position t, x_t^1 the property for x_t of belonging to a punctuation list and $x_t^2 = x_t^p$ the property of beginning with an upper case, provides the following feature :

$f_k(t, y_t, y_{t-1}, x) = 1$ if (y_t =DET+B) $\wedge$ (x_t^0 ='son') $\wedge$ ($x_t^1 = 0$) $\wedge$ ($x_t^2 = 0$) $\wedge$ (x_{t-1}^0 =',') $\wedge$ ($x_{t-1}^1 = 1$) ;

$f_k(t, y_t, y_{t-1}, x) = 0$ otherwise

The usual trick of CRFs consists in using the same set of

6. http ://crfpp.googlecode.com/svn/trunk/doc/index.html
7. http ://wapiti.limsi.fr/

features for each position t, so the value of the position itself ($t = 6$) is not a criterion in the definition of the feature.

3.3. Integrating external resources into a CRF

To take into account external resources in a learning process using CRF models, many different solutions are possible. To illustrate them, we take again the labeled example of section 2.1.. A lexical resource will provide information about the set of every possible POS category that the lexical units of this sentence can have :

- quant à : prep
- quant : prep
- à : prep
- la : cn, det, pro
- technique : cn
- ...

The set of categories of the resource is not necessary the same as the one of the labeled example, this is why we write them with lower cases. Some of the solutions proposed further require that the set of labels match, others don't.

The first possible use of a resource is called *Filtering*. It requires that the set of labels match. The process of filtering consists in using the resource to limit the search space of possible labelings. Instead of searching for the y that maximizes $P_\lambda(y|x)$, we search for the best *y among those compatible with the resource* which maximizes this value. In other words, incompatible labelings are discarded. The filtering step is, in this case, independent of the learning step, and can be applied before of after it.

A second more interesting approach consists in taking into account the resource *during the learning step*, by integrating the information it contains into the tabular-like file. Even then, there are various possible options. Three of them have been identified :

- *Learn-concat* : a single column is added to the file, which contains a string concatenating in a fixed order each possible category associated with the token in the resource. This string thus becomes a new property of the token. This solution does not require at all that the sets of labels are the same. The training file will then become :

Quant	prep	PREP+B
à	prep_prep+I	PREP+I
la	cn_det_pro	DET+B
technique	cn	CN+B
...		

  Note that when the resource includes multiword units, they are treated the same way as in the rest of the corpus, i.e. whith an additional I label for internal units (the B label is implicit everywhere else).
- *Learn-bool* : each possible category mentioned in the resource is considered as a new boolean property taking the value 1 if it is a possible category for this token, 0 otherwise. The training file will then become :

	det	cn	prep	prep+I	...	
Quant	0	0	1	0		PREP+B
à	0	0	1	1		PREP+I
la	1	1	0	0		DET+B
technique	0	1	0	0		CN+B
...						

  This strategy does not require either that the sets of labels are the same. But the generation of every possible feature template in this case is potentially explosive, as every possible combination of columns should give rise to a distinct template.
- *Learn-ex* : each possible couple made of a token and a category seen in the resource can be considered as a new example of the training set. This strategy implements the idea that a resource is a collection of observed possible instances. In this case, the training set receives new lines of the form :

Quant	PREP+B
à	PREP+I
Quant	PREP+B
à	PREP+B
la	CN+B
la	DET+B
la	PRO+B
technique	CN+B
...	

This strategy requires that the sets of labels match. It has the advantage of adding possible instances of associations between a token and a label, which may not appear otherwise in the examples. But it can perturb the probability distributions of these associations, if the set of examples is small. By adding a large number of new examples, it may also slow down the learning step. Furtermore, feature templates with "holes" outside of the current position (in particular, every binary feature templates) will not apply, as each of these new examples is surrounded by blank lines.

4. Evaluation

We trained different CRF models with the software Wapiti (Lavergne et al., 2010) using the algorithm *rprop*. The standard feature templates (i.e. without lexicon-based ones) are defined in the table below : w_t stands for the token at the relative position t from the current token ; l_t is the label at the relative position t.

$w_t = X, t \in \{-2, -1, 0, 1, 2\}$	$\&l_0 = L$
Lowercase form of $w_0 = W$	$\&l_0 = L$
Prefix of $w_0 = P$ with $\|P\| < 5$	$\&l_0 = L$
Suffix of $w_0 = S$ with $\|S\| < 5$	$\&l_0 = L$
w_0 contains a hyphen	$\&l_0 = L$
w_0 contains a digit	$\&l_0 = L$
w_0 is capitalized	$\&l_0 = L$
w_0 is all in capital	$\&l_0 = L$
w_0 is capitalized and BOS [8]	$\&l_0 = L$
w_0 is part of a multiword	$\&l_0 = L$
$w_i w_j = XY, (j, k) \in \{(-1, 0), (0, 1), (-1, 1)\}$	$\&l_0 = L$
$l_{-1} = L'$	$\&l_0 = L$

TABLE 1 – Feature templates without lexicon-based ones

We used all the resources described in subsection 2.3. and we tested two ways of integrating lexicon-based features : *Learn-concat* and *Learn-bool* (cf. subsection 3.3.). Each model was trained on 80% of the FTB with an Intel(R)

Core(TM)2 Quad CPU Q9400 @ 2.66GHz including 3.6 Gb memory. Each model combined about 40 million (resp. 25 million) features and its training duration was 2h-2h20 (resp. 1h-1h10) for version (i) of the FTB (resp. version ii). Three different approaches were evaluated : (a) baseline approach ; (b) 1-model approach, (c) 2-model approach. The baseline approach (a) consists in first performing a naive lexical segmentation (including MW recognition) and then in tagging the segmented text with a standard POS-tagger. The segmentation phase is based on a lexical analysis that generates a finite automaton of all possible analyses for each sentence. The segmentation is found by selecting the shortest path for each of them, i.e. giving priority to the longest analyses (i.e. MWUs). The lexical analysis is performed by a simple look-up in the MWU lexicon of the training corpus. The POS-standard tagger is CRF-based and uses the template features detailed in (Constant and Sigogne, 2011) and the lexical resources described in subsection 2.3.. It reaches 97.7-97.8% accuracy (94.3-94.4% for MWUs) when the segmentation is perfect. The 1-model approach (b) consists in applying a single CRF model that allows for jointly performing the lexical segmentation and the POS-tagging (cf. 2.1.). The 2-model approach (c) is composed of two phases : a lexical segmentation by applying the same model as in (b) (the POS labels are ignored) and then a POS-tagging of the segmented text with the same standard tagger as in (a).

We evaluated the different approaches by cross-validation, by using a standard f-score on the lexical unit segments (uniformely combining recall and precision). We computed the global tagging score and the tagging score of the MWUs (in parenthesis). Results are provided in table 2. Note that the *Filtering* and *Learn-ex* methods described in subsection 3.3. were not tested here, because (Constant et al., 2011) have shown that they are less efficient than the other two.

We can first observe that the segmentation has a cost of around 1.8 and 3.8 points (97.8% for standard tagging vs. 96.0% and 94.0% for joint segmentation and tagging). The integration of large-coverage lexicons in the learning step makes the results improve by 0.5 points in the best case. The best lexicon-based method is the concatenation one (*Learn-concat*). The experiments also show that a 2-model approach has a similar score as a 1-model approach at the cost of training two models instead of one. Both approaches outperform the baseline by 1.2-1.3 points.

We also evaluated the tagging performances when the training corpus size varies. They are significantly different whether the model integrates or not lexicon-based features. The tagging score evolutions are given in Figures 2 and 3. They show that the use of lexicon-based features might compensate a small training corpus. For instance, with lexicon-based features, only 30% of the version (ii) of the FTB is needed to obtain a model with equivalent performances as the model trained with 90% of the FTB that does not integrate lexicon-based features.

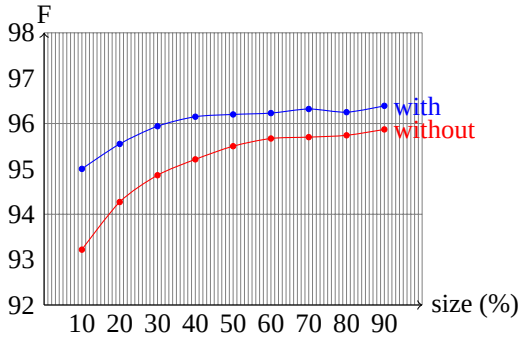

FIGURE 2 – Overall tagging score evolution according to the size of the training corpus. The size is indicated in terms of percentage of the version (ii) of the French Treebank. We applied 5-fold cross-validation with (resp. without) corresponds to the experiment with lexical resources (resp. without).

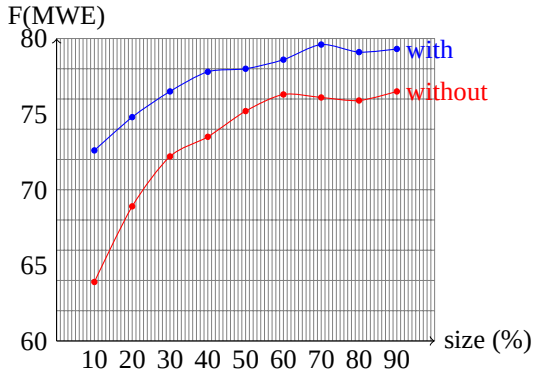

FIGURE 3 – Multiword expression tagging score evolution according to the size of the training corpus. The size is indicated in terms of percentage of the version (ii) of the French Treebank. We applied a 5-fold cross-validation with (resp. without) corresponds to the experiment with lexical resources (resp. without).

5. Summary and Future Work

This paper evaluates the impact of external lexical resources into a CRF-based joint Multiword Segmenter and Part-of-Speech Tagger. We especially show different ways of integrating lexicon-based features in the model. We display an absolute gain of 0.5% in terms of f-measure. Moreover, we show that the integration of lexicon-based features significantly compensates the use of a small training corpus.

Future work would consist in adapting such approach to chunking. We are also willing to integrate new types of features computed from a symbolic rule-based tagger.

6. References

A. Abeillé, L. Clément, and F. Toussenel. 2003. Building a treebank for french. In A. Abeillé, editor, *Treebanks*. Kluwer, Dordrecht.

M. Candito and B. Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technology (IWPT'09)*, pages 138–141.

M. Constant and A. Sigogne. 2011. MWU-aware part-of-speech tagging with a CRF model and lexical resources.

		lexicon-based features			
		FTB (i)		FTB (ii)	
Approach	Resource	Learn-concat	Learn-bool	Learn-concat	Learn-bool
Baseline	no	92.7 (71.0)		93.8 (64.9)	
1-model	no	93.5 (73.2)		95.5 (76.0)	
1-model	yes	94.0 (75.1)	93.8 (74.0)	**96.0** (**78.2**)	95.7 (76.7)
2-model	yes	94.0 (75.1)	93.9 (74.2)	96.0 (78.1)	95.9 (76.8)

TABLE 2 – Evaluation (f-score in percentage) : general tagging score with MWU tagging score in parenthesis.

In *Proceedings of ACL workshop on Multiword Expressions*, Portland, Oregon.

M. Constant, I. Tellier, D. Duchier, Y. Dupont, A. Sigogne, and S. Billot. 2011. Intégrer des connaissances linguistiques dans un CRF : application à l'apprentissage d'un segmenteur-étiqueteur du français. In *Actes de la Conférence sur le traitement automatique des langues naturelles (TALN'11)*, Montpellier, France.

B. Courtois, M. Garrigues, G. Gross, M. Gross, R. Jung, M. Mathieu-Colas, A. Monceaux, A. Poncet-Montange, M. Silberztein, and R. Vivés. 1997. Dictionnaire électronique DELAC : les mots composés binaires. Technical Report 56, LADL, University Paris 7.

Blandine Courtois. 1990. Un système de dictionnaires électroniques pour les mots simples du français. *Langue Française*, 87 :11–22.

P. Denis and B. Sagot. 2009. Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art pos tagging with less human effort. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC 2009)*.

Maurice Gross. 1997. The construction of local grammars. In E. Roche and Y. Schabes, editors, *Finite-State Language Processing*, pages 329–352. The MIT Press, Cambridge, Mass.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001)*, pages 282–289.

Thomas Lavergne, Olivier Cappé, and François Yvon. 2010. Practical very large scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics, July.

Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*.

O. Piton, D. Maurel, and C. Belleil. 1999. The prolex data base : Toponyms and gentiles for nlp. In *Proceedings of the Third International Workshop on Applications of Natural Language to Data Bases (NLDB'99)*, pages 233–237.

L. A. Ramshaw and M. P. Marcus. 1995. Text chunking using transformation-based learning. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 88–94.

B. Sagot. 2010. The lefff, a freely available, accurate and large-coverage lexicon for french. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC'10)*.