Lang. Resources & Evaluation
DOI 10.1007/s10579-012-9193-0

Coupling an annotated corpus and a lexicon for state-of-the-art POS tagging

Pascal Denis, Benoît Sagot

Abstract This paper investigates how to best couple hand-annotated data with information extracted from an external lexical resource to improve POS tagging performance. Focusing mostly on French tagging, we introduce a Maximum Entropy Markov Model-based tagging system that is enriched with information extracted from a morphological resource. This system gives a 97.75% accuracy on the French Treebank, an error reduction of 25% (38% on unknown words) over the same tagger without lexical information. We perform a series of experiments that help understanding how this lexical information helps improving tagging accuracy. We also conduct experiments on datasets and lexicons of varying sizes in order to assess the best trade-off between annotating data vs. developing a lexicon. We find that the use of a lexicon improves the quality of the tagger at any stage of development of either resource, and that for fixed performance levels the availability of the full lexicon consistently reduces the need for supervised data by at least one half.

Keywords Part-of-speech tagging, maximum entropy models, morphosyntactic lexicon, French, language resource development

1 Introduction

Over recent years, numerous systems for automatic part-of-speech (POS) tagging have been proposed for a large variety of languages. Among the best performing systems are those based on supervised machine learning techniques (see [12] for an overview). For some languages like English and other European languages, these systems have reached performance that comes close to human levels. Interestingly, the majority of these systems have been built without resorting to any external lexical information sources; they instead rely on a dictionary that is based on the training corpus (see however [9]). This raises the question of whether we can still improve tagging performance by exploiting this type of resource. Arguably, a potential advantage of using an external dictionary is in a better handling of unknown words (i.e., words that are not present in the training corpus, but that may be present

Alpage, INRIA Paris-Rocquencourt & Université Paris 7
Domaine de Voluceau — Rocquencourt, BP 105
78153 Le Chesnay Cedex, France
{pascal.denis,benoit.sagot}@inria.fr

in the external dictionary). A subsequent question is how to best integrate the information from a lexical resource into a probabilistic POS tagger. In this paper, we consider two distinct scenarios: (i) using the external dictionary as *constraints* that restrict the set of possible tags that the tagger can choose from, and (ii) incorporating the dictionary tags as *features* in a probabilistic POS tagging model. Another interesting question is that of the relative impact of training corpora and of lexicons of various sizes. This issue is crucial to the development of POS taggers for resource-scarce languages for which it is important to determine the best trade-off between annotating data and constructing dictionaries.

This paper addresses these questions through various tagging experiments carried out on French, based on our new tagging system called MElt (Maximum-Entropy Lexicon-enriched Tagger). An obvious motivation for working on this language is the availability of a training corpus (namely, the French Treebank [1]) and a large-scale lexical resource (namely, Le*fff* [21]). Additional motivation comes from the fact that there has been comparatively little work in probabilistic POS tagging in this language. An important side contribution of our paper is the development of a state-of-the-art, freely distributed POS tagger for French.[1] Specifically, we here adopt Maximum Entropy Markov Models (MEMMs), an extension of MaxEnt models for sequence labeling. MEMMs remain among the best performing tagging systems for English and they are particularly easy to build and fast to train.

This paper is organized as follows. Section 2 describes the datasets and the lexical resources that were used. Section 3 presents a baseline MEMM tagger for French that is inspired by previous work, in particular [17] and [27], that already outperforms TreeTagger [23] retrained on the same data. In Section 4, we show that the performance of our MEMM tagger can be further improved by incorporating features extracted from a large-scale lexicon, reaching a 97.75% accuracy, which compares favorably with the best results obtained for English with a similar tagset. In order to assess the robustness of our approach, we apply it to various languages and of different sizes. Finally, Section 6 evaluates the relative impact on accuracy of the training data and the lexicon during tagger development by varying their respective sizes. These last two sections build on two previous conference publications, [8] and [7], respectively.

2 Resources and tagset

2.1 Corpus

The morphosyntactically annotated corpus we used is a variant of the French TreeBank or FTB, [1]. It differs from the original FTB in so far that all compounds that do not correspond to a syntactically regular sequence of categories have been merged into unique tokens and assigned a category corresponding to their spanning node; other compounds have been left as sequences of several tokens (Candito, p.c.). The resulting corpus has $350,931$ tokens in $12,351$ sentences.

In the original FTB, words are split into 13 main categories, themselves divided into 34 subcategories. The version of the treebank we used was obtained by converting subcategories into a tagset consisting of 28 tags, with a granularity that is intermediate between categories and subcategories. Basically, these tags enhance main categories with information on the mood of verbs and a few other lexical features. This expanded tagset has been

[1] The MElt tagger is freely available from `http://lingwb.gforge.inria.fr/`. Results reported in this paper correspond to release MElt 1.0.

Cette/DET mesure/NC ,/PONCT qui/PROREL pourrait/V être/VINF appliquée/VPP dans/P les/DET prochaines/ADJ semaines/NC ,/PONCT permettrait/V d'/P économiser/VINF quelque/DET 4/ADJ milliards/NC de/P francs/NC ./PONCT

Fig. 1 Sample data from FTB in Brown format

shown to give the best statistical parsing results for French [6].[2] A sample tagged sentence from the FTB is given in Figure 1.

As in [4], the FTB is divided into 3 sections: training (80%), development (10%) and test (10%). The dataset sizes are presented in Table 1 together with the number of unknown words.

Data Set	# of sent.	# of tokens	# of unk. tokens
FTB-TRAIN	9,881	278,083	
FTB-DEV	1,235	36,508	1,892 (5.2%)
FTB-TEST	1,235	36,340	1,774 (4.9%)

Table 1 Data sets

2.2 Lexicon

One of the goals of this work is to study the impact of using an external dictionary for training a tagger, in addition to the training corpus itself. We used the morphosyntactic information included in the large-coverage morphological and syntactic lexicon Lefff, developed in the Alexina framework [20].[3]

Although Lefff contains both morphological and syntactic information for each entry (including sub-categorization frames, in particular for verbs), we extracted only the morphosyntactic information. We converted categories and morphological tags into the same tagset used in the training corpus, hence building a large-coverage morphosyntactic lexicon containing 507,362 distinct entries of the form *(form, tag, lemma)*, corresponding to 502,223 distinct entries of the form *(form, tag)*. If grouping all verbal tags into a single "category" while considering all tags as "categories", these entries correspond to 117,397 *(lemma, category)* pairs (the relevance of these pairs will appear in Section 6).

3 Baseline MEMM tagger

This section presents our baseline MaxEnt-based French POS tagger, $\text{MElt}^0_{\text{fr}}$. This tagger is largely inspired by [17] and [27], both in terms of the model and the features being used. To date, MEMM taggers are still among the best performing taggers developed for English.[4] An important appeal of MaxEnt models is that they allow for the combination of very diverse,

[2] This tagset is known as TREEBANK+ in [6], and since then as CC [4].

[3] The Lefff is freely distributed under the LGPL-LR license at `http://alexina.gforge.inria.fr/`

[4] [17] and [27] report accuracy scores of 96.43% and 96.86% on section 23-24 of the Penn Treebank, respectively.

potentially overlapping features without assuming independence between the predictors. These models have also the advantage of being very fast to train.[5]

3.1 Description of the task

Given a tagset T and a string of words w_1^n, we define the task of tagging as the process of assigning the maximum likelihood tag sequence $\hat{t}_1^n \in T^n$ to w_1^n. Following [17], we can approximate the conditional probability $P(t_1^n|w_1^n)$ so that:

$$\hat{t}_1^n = \arg\max_{t_1^n \in T^n} P(t_1^n|w_1^n) \approx \arg\max_{t_1^n \in T^n} \prod_{i=1}^{n} P(t_i|h_i) \tag{1}$$

where t_i is the tag for word w_i, and h_i is the "history" (or context) for (w_i, t_i), which comprises the preceding tags t_i^{i-1} and the word sequence w_i^n.

3.2 Model and features

In a MaxEnt model, the parameters of an exponential model of the following form are estimated:

$$P(t_i|h_i) = \frac{1}{Z(h)} \cdot \exp\left(\sum_{i=j}^{m} \lambda_j f_j(h_i, t_i)\right) \tag{2}$$

f_1^m are feature functions defined over tag t_i and history h_i (with $f(h_i, t_i) \in \{0, 1\}$), λ_1^m are the parameters associated with f_1^m, and $Z(h)$ is a normalization term over the different tags. In this type of model, the choice of the parameters is subject to constraints that force the model expectations of the features to be equal to their empirical expectations over the training data [3]. In our experiments, the parameters were estimated using the Limited Memory Variable Metric Algorithm [11] implemented in the Megam package.[6]

The feature templates we used for designing our French tagging model is a superset of the features used by [17] and [27] for English (these were largely language independent). These features fall into two main categories. A first set of features try to capture the *lexical form* of the word being tagged: these include the actual word string for the current word w_i, prefixes and suffixes (of character length 4 and less), as well as binary features testing whether w_i contains special characters like numbers, hyphens, and uppercase letters. A second set of features directly model the *context* of the current word and tag: these include the previous tag, the concatenation of the two previous tags, as well as the surrounding word forms in a window of 5 tokens.

The detailed list of feature templates we used in this baseline tagger is shown in Table 2.[7]

[5] Arguably better suited for sequential problems, Conditional Random Fields (CRF) [10] are considerably slower to train.

[6] Available from `http://www.cs.utah.edu/~hal/megam/`.

[7] Recall that features in MaxEnt are functions ranging on both contexts and classes. A concrete example of one of our features is given below:

$$f_{100}(h, t) = \begin{cases} 1 \text{ if } w_i = \text{"le" \& } t = \texttt{DET} \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

Internal lexical features	
$t_i =_{uni} X$, if $\mathsf{lefff}(w_i) = \{X\}$	& $t_i = T$
$t_i = X, \forall X \in \mathsf{lefff}(w_i)$ if $\lvert\mathsf{lefff}(w_i)\rvert > 1$	& $t_i = T$
$t_i = \bigvee \mathsf{lefff}(w_i)$ if $\lvert\mathsf{lefff}(w_i)\rvert > 1$	& $t_i = T$
$t_i = unk$, if $\mathsf{lefff}(w_i) = \emptyset$	& $t_i = T$
External lexical features	
$t_{i+j} = \bigvee \mathsf{lefff}(w_{i+1}), j \in \{-2, -1, 1, 2\}$	& $t_i = T$
$t_{i+j}t_{i+k} = \bigvee \mathsf{lefff}(w_{i+j}) \bigvee \mathsf{lefff}(w_{i+k}), (j,k) \in \{(-2,-1),(1,2),(-1,1)\}$	& $t_i = T$

Table 2 Baseline model features

An important difference with [17] in terms of feature design is that we did not restrict the application of the prefix/suffix features to words that are rare in the training data. In our model, these features always get triggered, even for frequent words. We found that the permanent inclusion of these features led to better performance during development, which can probably be explained by the fact that these features get better statistics and are extremely useful for unknown words. These features are also probably more discriminative in French than in English, since it is morphologically richer. Another difference to previous work regards smoothing. [17] and [27] use a feature count cutoff of 10 to avoid unreliable statistics for rare features. We did not use cutoffs but instead use a regularization Gaussian prior on the weights[8], which is arguably a more principled smoothing technique.[9]

3.3 Testing and Performance

The test procedure relies on a *beam search* to find the most probable tag sequence for a given sentence. That is, each sentence is decoded from left to right and we maintain for each word w_i the n highest probability tag sequence candidates up to w_i. For our experiments, we used a beam size of 3.[10] In addition, the test procedure utilizes a *tag dictionary* which lists for a given word the tags associated with this word in the training data. This drastically restricts the allowable labels that the tagger can choose from for a given word, in principle leading to fewer tagging errors and reduced tagging time.

The maximum entropy tagger described above, $\text{MElt}^0_{\text{fr}}$, was compared against two other baseline taggers, namely: UNIGRAM and TreeTagger. UNIGRAM works as follows: for a word seen in the training corpus, this tagger uses the most frequent tag associated with this word in the corpus; for unknown words, it uses the most frequent tag in the corpus (in this case, NC). TreeTagger is a statistical, decision tree-based POS tagger [23].[11] The version used for this comparison was retrained on the FTB training corpus. The performance results of the three taggers are given in Table 3; scores are reported in terms of accuracy over both the entire test set and the words that were not seen during training.

As shown in Table 3, $\text{MElt}^0_{\text{fr}}$ achieves accuracy scores of 97% overall and 86.1% on unknown words.[12] Our baseline tagger significantly outperforms the retrained version of

[8] Specifically, we used a prior with precision (i.e., inverse variance) of 1 (which is the default in Megam); other values were tested during development but did not yield improvements.

[9] Informally, the effect of this kind of regularization is to penalize artificially large weights by forcing the weights to be distributed according to a Gaussian distribution with mean zero.

[10] We tried larger values (i.e., 5, 10, 15, 20) during development, but none of these led to significant improvements.

[11] Available at `http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/`.

[12] The accuracy results of $\text{MElt}^0_{\text{fr}}$ on FTB-DEV are: 96.7% overall and 86.2% on unknown words.

Tagger	Overall Accuracy	Unknown Word Accuracy
UNIGRAM	91.90	24.50
TreeTagger	96.12	75.77
$\text{MElt}^0_{\text{fr}}$	**97.00**	**86.10**

Table 3 Baseline tagger performance

TreeTagger, with an improvement of over 10% on unknown words.[13] There are several possible explanations for such a discrepancy in handling unknown words. The first one is that MaxEnt parameter estimation is less prone to data fragmentation for sparse features than Decision Tree parameter estimation due to the fact that it does not partition the training sample. A second related explanation is that TreeTagger simply misses some of the generalizations regarding lexical features due to the fact that it only includes suffixes and this only for unknown words.

4 Lexicon-enriched MEMM tagger

For trying to further improve $\text{MElt}^0_{\text{fr}}$, we investigate in this Section the impact of coupling it with an external lexical resource, and compare two ways of integrating this new information: as constraints vs. as features.

4.1 Integrating lexical information in the tagger

The most natural way to make use of the extra knowledge supplied by a lexicon is to represent it as "filtering" constraints: that is, the lexicon is used as an additional tag dictionary guiding the POS tagger, in addition to the lexicon extracted from the training corpus. Under this scenario, the tagger is forced for a given word w to assign one of the tags associated with w in the full tag dictionary: the set of allowed tags for w is the union of the sets of its tags in the corpus and in Lefff. This approach is similar to that of [9], who applied it to highly inflected languages, and in particular to Czech.

In a learning-based tagging approach, there is another possibility to accommodate the extra information provided by Lefff: we can directly incorporate the tags associated by Lefff to each word in the form of features. Specifically, for each word, we posit a new lexical feature for each of its possible tags according to the Lefff, as well as a feature that represents the disjunction of all Lefff tags (provided there is more than one). Similarly, we can also use the Lefff to provide additional contextual features: that is, we can include Lefff tags for all the words in a window of 5 tokens centered on the current token. Table 4 summarizes these new feature templates.

Integrating the lexical information in this way has a number of potential advantages. First, features are by definition more robust to noise (in this case, to potential errors in the lexicon or simply mismatches between the corpus annotations and the lexicon categories). Furthermore, some of the above features directly model the context, while the filtering constraints are entirely non contextual.

[13] Chi-square statistical significance tests were applied to changes in accuracy, with p set to 0.01 unless otherwise stated.

Lexical features	
Lefff tag for $w_i = X$	& $t_i = T$
Lefff tags for $w_i = X_0 \dots X_n$	& $t_i = T$
Contextual features	
Lefff tag for $w_{i+j} = X, j \in \{-2, -1, 1, 2\}$	& $t_i = T$
Lefff tags for $w_{i+j} = X_0 \dots X_n, j \in \{-2, -1, 1, 2\}$	& $t_i = T$

Table 4 Lexicon-based features

4.2 Comparative evaluation

We compared the performance of the Lefff-constraints based tagger $\text{MElt}^c_{\text{fr}}$ and Lefff-features based tagger $\text{MElt}^f_{\text{fr}}$ to other lexicon-enriched taggers. The first of these taggers, $\text{UNIGRAM}_{\text{Lefff}}$, like UNIGRAM, is a unigram model based on the training corpus, but it uses Lefff for labeling unknown words: among the possible Lefff tag for a word, this model chooses the tag that is most frequent in the training corpus (all words taken into account). Words that are unknown to both the corpus and Lefff are assigned NC. The second tagger, $\text{TreeTagger}_{\text{Lefff}}$ is a retrained version of TreeTagger to which we provide Lefff as an external dictionary. Finally, we also compare our tagger to F-BKY, an instantiation of the Berkeley lexicalized parser adapted for French by [6] and used as a POS tagger. The performance results for these taggers are given in Table 5.

Tagger	**Overall accuracy (%)**	**Unknown words accuracy (%)**
$\text{UNIGRAM}_{\text{Lefff}}$	93.40	55.00
$\text{TreeTagger}_{\text{Lefff}}$	96.55	82.14
F-BKY	97.30	82.90
$\text{MElt}^0_{\text{fr}}$	97.00	86.10
$\text{MElt}^c_{\text{fr}}$	97.25	86.47
$\text{MElt}^f_{\text{fr}}$	**97.75**	**91.36**

Table 5 Lexicon-based taggers performance

The best tagger is $\text{MElt}^f_{\text{fr}}$, with accuracy scores of 97.75% overall and 91.36% for unknown words. This represents significant improvements of $.75\%$ and 5.26% over $\text{MElt}^0_{\text{fr}}$, respectively.[14] By contrast, $\text{MElt}^c_{\text{fr}}$ achieves a rather limited (and statistically insignificant) performance gain of $.1\%$ overall but a 2.9% improvement on unknown words. Our explanation for these improvements is that the Lefff-based features reduce data sparseness and provide useful information on the right context: first, fewer errors on unknown words (a direct result of the use of a morphosyntactic lexicon) necessarily leads to fewer erroneous contexts for other words, and therefore to better tagging; second, the possible categories of tokens that are on the right of the current tokens are valuable pieces of information, and they are available only from the lexicon. The lower result of $\text{MElt}^c_{\text{fr}}$ can probably be explained by two differences: it does not benefit from this additional information about the right context, and it uses Lefff information as hard constraints, not as (soft) features.

Accuracy scores put $\text{MElt}^f_{\text{fr}}$ above all the other taggers we have tested, including the parser-based F-BKY, by a significant margin. To our knowledge, these scores are the best

[14] The accuracy results of $\text{MElt}^f_{\text{fr}}$ on FTB-DEV are: 97.23% overall and 90.01% on unknown words.

Error type		Frequency
Standard errors	Adjective vs. past participle	5.5%
	Errors on *de*, *du*, *des*	4.0%
	Other errors	34.0%
Errors on numbers		15.5%
Errors related to named entities		27.5%
$\text{MElt}^{f}_{\text{fr}}$'s result seems correct	Error in FTB-DEV	8.5%
	Unclear cases (both tags seem valid)	4.5%
	Truncated text in FTB-DEV	0.5%

Table 6 Manual error analysis of the 200 first errors of $\text{MElt}^{f}_{\text{fr}}$ on the development corpus

scores reported for French POS tagging.[15] Other taggers have been proposed for French, some of which have been evaluated during the GRACE evaluation campaign.[16] Although a direct comparison is difficult, given the differences in terms of reference corpus and tagsets, it is worth mentioning that the best scores during this campaign, approximately 96%, have been obtained by parser-based taggers [2]. Finally, [16] report a 97.82% accuracy on the FTB, but their tagger/chunker does not take unknown words into account.

4.3 Error analysis

In order to understand whether the 97.75% accuracy of $\text{MElt}^{f}_{\text{fr}}$ could still be improved, we decided to examine manually its first 200 errors on FTB-DEV, and classify them according to an adequate typology of errors. The resulting typology and the corresponding figures are given in Table 6.

These results show that the 97.75% score can still be improved. Indeed, standard named entity recognition techniques could help solve most errors related to named entities, i.e., more than one out of four errors. Moreover, simple regular patterns could allow for replacing automatically all numbers by one or several placeholder(s) both in the training and evaluation data. Indeed, preserving numbers as such inevitably leads to a sparse data problem, which prevents the training algorithm from modeling the complex task of tagging numbers — they can be determiners, nouns, adjectives or pronouns. Appropriate placeholders should significantly help the training algorithm and improve the results. Finally, no less than 13.5% of $\text{MElt}^{f}_{\text{fr}}$'s apparent errors are in fact related to FTB-DEV's annotation, because of errors (9%) or unclear situations, for which both the gold tag and $\text{MElt}^{f}_{\text{fr}}$'s tag seem valid.

Given these facts, we consider it feasible to improve $\text{MElt}^{f}_{\text{fr}}$ from 97.75% to 98.5% in the future.

[15] An adaptation to French of the Morfette POS-tagger [5] using the FTB and the Lefff has been realized by G. Chrupała and D. Seddah (p.c.). Their accuracy results are similar to ours, although slightly lower (on the same data sets, Henestroza and Candito have obtained a 97.68% accuracy). On other variants of the FTB, Chrupała and Seddah report 97.9% (p.c.). However, these figures do not correspond exactly to the same experimental setup, as Morfette extracts and uses in its models an additional source of information, namely lemmas.

[16] `http://www.limsi.fr/TLP/grace/`

4.4 Impact of various sets of Le*fff*-based lexical features

In order to understand better the relative impact on MElt$^{f}_{\text{fr}}$'s model of various types of information extracted from the Le*fff*, we have run a series of ablation experiments on the set of features described in 4. Specifically, we have evaluated the 8 possible configurations consisting in including (or not) internal lexical features (INT), external lexical features defined on the left context (LEFT), and external lexical features defined on the right context (RIGHT). The results of these experiments performed on the development corpus FTB-DEV are given in Table 7. Note that the experiments named here $\emptyset$ and INT+LEFT+RIGHT, correspond respectively to the variants MElt$^{0}_{\text{fr}}$ and MElt$^{f}_{\text{fr}}$ of our system.

Le*fff* features	**Overall accuracy (%)**	**Unknown words accuracy (%)**
$\emptyset$ (MElt$^{0}_{\text{fr}}$)	96.54	83.95
INT	97.04	91.4
LEFT	96.38	85.36
RIGHT	96.39	86.48
INT+LEFT	96.92	91.28
INT+RIGHT	97.30	92.01
LEFT+RIGHT	96.57	86.93
INT+LEFT+RIGHT (MElt$^{f}_{\text{fr}}$)	97.41	92.35

Table 7 Comparative accuracy of MElt$^{f}_{\text{fr}}$ using various subsets of lexical features on FTB-DEV

These results indicate that it is the combination of internal and right external lexical features that brings the most information to the tagger. Indeed, the subset INT+RIGHT yields the best results after MElt$^{f}_{\text{fr}}$ itself, both on all words and on unknown words only. These two subsets of lexical features are complementary: INT features improve the lexical coverage of the tagger (some unknown words, i.e., unseen in the training corpus, are covered by the lexicon), whereas RIGHT features provide important information about the right context that MElt$^{0}_{\text{fr}}$'s features only model in a rough way.

5 Varying tagsets and languages

In order to validate the robustness of our approach, we have trained two series of taggers with the same architecture as MElt$^{f}_{\text{fr}}$:

- several other taggers trained on the same corpus and the same lexicon, but with tagsets of different granularities; indeed, different NLP tasks may require tagging with a different level of detail, including only major categories (tagset `small`, 15 tags), standard categories (standard taget, 28 tags) and detailed tags that include morphological features such as gender, number, person, tense, and others (tagset `large`, 239 tags);
- two other taggers trained on corpora and lexicon for other languages, namely English and Spanish; for English, we used the Penn TreeBank [13] as a corpus (sections 2 to 21 for training and section 23 for tagging, as usual in the parsing community; 46 tags) and the lexicon EnLex developed in the same framework as the Le*fff*, Alexina; for Spanish, we used the Ancora corpus [26] (the first 60,000 sentences as a training corpus, the last 3,328 sentences as a test corpus; 16 tags) and the Alexina lexicon for Spanish, the Le*ff*e [15].

Language	Training Corpus	Tagset Size	Overall Acc. (%)
French	FTB-TRAIN (`small` tagset)	15	97.69%
French ($\text{MElt}^{f}_{\text{fr}}$)	FTB-TRAIN (standard tagset)	28	97.25%
French	FTB-TRAIN (`large` tagset)	239	93.73%
English	Penn TreeBank (sec. 2-23)	46	97.03%
Spanish	Ancora (60,000 first sentences)	16	97.73%

Table 8 Applying the MElt system on other tagsets and languages (see text for information about test corpora)

Results are shown in Table 8. Note that the last three experiments (French `large`, English, Spanish) rely on lexicons that do not use the same tagset as the corpus.[17] Indeed, since lexical information extracted from the lexicon is used in the form of features, there is no particular need for having the same tags in the lexicon as in the corpus.

These results are satisfying not only for $\text{MElt}^{f}_{\text{fr}}$ itself. For example, the state-of-the-art for English on the same corpus (not necessarily split in the same way, though) is approximately 97.4% [25], a figure that is reached by combining several taggers.

6 Varying training corpus and lexicon sizes

6.1 Motivations and experimental setup

The results achieved by $\text{MElt}^{f}_{\text{fr}}$ have been made possible by the (relatively) large size of the corpus and the broad coverage of the lexicon. However, such resources are not always available for a given language, in particular for so-called under-resourced languages. Moreover, the significant improvement observed by using Le*fff* shows that the information contained in a morphosyntactic lexicon is worth using. The question arises whether this lexical information is able to compensate for the lack of a large training corpus. Symmetrically, it is unclear how various lexicon sizes impact the quality of the results.

Therefore, we performed a series of experiments by training $\text{MElt}^{f}_{\text{fr}}$ on various sub-corpora and sub-lexicons. Extracting sub-corpora from FTB-TRAIN is simple: the first s sentences constitute a reasonable corpus of size s. However, extracting sub-lexicons from the Le*fff* is less trivial. We decided to extract increasingly large sub-lexicons in a way that approximately simulates the development of a morphosyntactic lexicon. To achieve this goal, we used the $\text{MElt}^{f}_{\text{fr}}$ tagger described in the previous section to tag a large raw corpus.[18] We then lemmatized the corpus by assigning to each token the list of all of its possible lemmas that exhibit a category consistent with the annotation. Finally, we ranked all resulting *(lemma, category)* pairs w.r.t. frequency in the corpus. Extracting a sub-lexicon of size n then consists in extracting all *(form,tag,lemma)* entries whose corresponding *(lemma, category)* pair is among the l best ranked ones.

We reproduced the same experiments as those described in Section 4, but training $\text{MElt}^{f}_{\text{fr}}$ on various sub-corpora and various sub-lexicons. We used 9 different lexicon sizes and 8 different corpus sizes, summed up in Table 9. For each resulting tagger, we evaluated on FTB-TEST the overall accuracy and the accuracy on unknown words.

[17] MElt has also been used for training POS taggers for Persian [22] and Kurmanji Kurdish [28] (see below) based on noisy corpora and medium-size lexicons, with promising results.

[18] We used a corpus of 20 million words extracted from the *L'Est Républicain* journalistic corpus, freely available at the web site of the CNRTL (`http://www.cnrtl.fr/corpus/estrepublicain/`).

Lexicon size (lemmas)	0; 500; 1,000; 2,000; 5,000; 10,000; 20,000; 50,000; 110,000
Corpus size (sentences)	50; 100; 200; 500; 1,000; 2,000; 5,000; 9,881

Table 9 Varying training corpus and lexicon sizes: experimental setups

6.2 Results and discussion

Before comparing the respective relevance of lexicon and corpus manual development for optimizing the tagger performance, we need to be able to quantitatively compare their development costs, i.e., times.

In [13], the authors report a POS annotation speed that "exceeds 3,000 words per hour" during the development of the Penn TreeBank. This speed is reached after a 1 month period (with 15 annotation hours per week, i.e., approximately 60 hours) during which the POS tagger used for pre-annotation was still improving. The authors also report on a manual tagging experiment (without automatic pre-annotation); they observed an annotation speed that is around 1,300 words per hour. Therefore, it is probably safe to assume that, on average, the creation of a manually validated training corpus starts at a speed that is around 1,000 words (30 sentences) per hour, and increases up to 3,000 words (100 sentences) per hour once the corpus has reached, say, 5,000 sentences.

For lexicon development, techniques such as those described in [19] allow for a fast validation of automatically proposed hypothetical lemmas. Manual intervention is then limited to validation steps that take around 2 to 3 seconds per lemma, i.e., about 1,500 lemmas per hour.

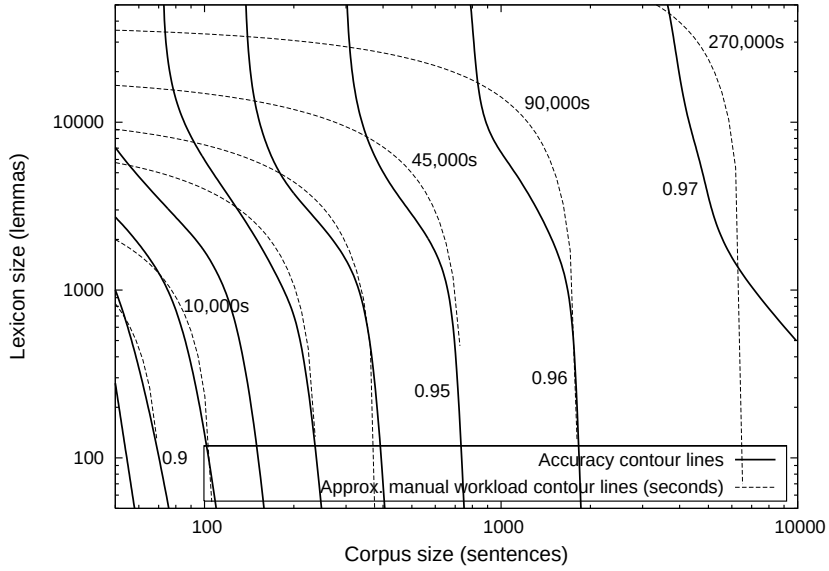
Figure 2 compares contour lines[19] for two functions of corpus and lexicon sizes: tagger accuracy and development time.[20] These graphs show different things:

- during the first steps of development (less than 3 hours of manual work), the distribution of the manual work between lexicon and corpus development has no significant impact on overall tagging accuracy, but accuracy on unknown words is better when focusing more or equally on the lexicon than on the corpus;
- in later stages of development, the optimal approach is to develop *both* the lexicon and the corpus, and this is true for both overall and unknown words tagging accuracy; however, it is by far better to concentrate only on corpus than only on lexicon development;
- using a morphological lexicon drastically improves the tagging accuracy on unknown words, whatever the development stage;[21]
- for fixed performance levels, the availability of the full lexicon consistently reduces the need for training data by at least one half (and up to two thirds).

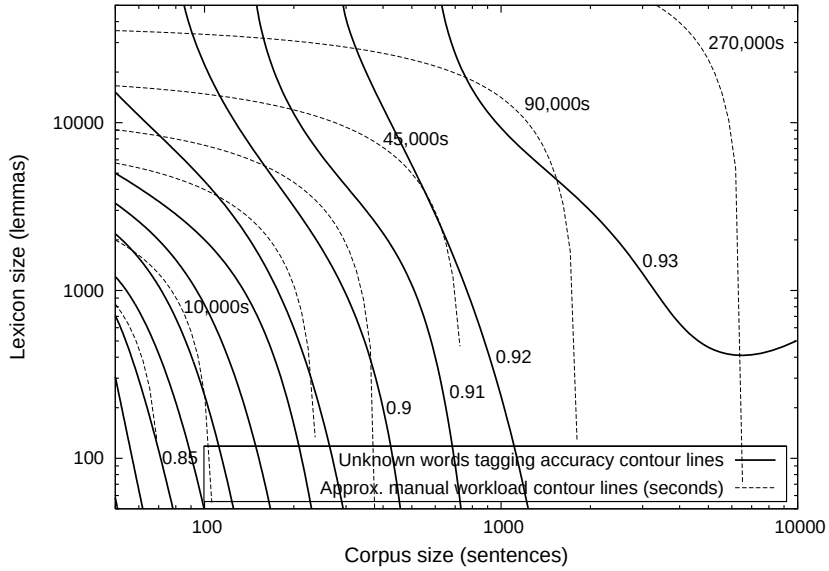
[19] As computed by the `bspline` mode of `gnuplot`'s contour lines generation algorithm.

[20] The development times per sentence and per lexical entry mentioned in the previous paragraphs lead to the following formula for the total development time $t(s,l)$ (expressed in seconds), in which s is the number of sentences, l the number of lexical entries: $t(s,l) = 36s + 8400 \cdot log(s/100 + 1) + 2.4 \cdot l$.

[21] Performing POS tagging with a morphological lexicon but without any training corpus is a significantly different task, addressed by an increasing literature [14, 24, 18]. In that regard, MElt has been used in a simple experiment on Kurmanji Kurdish, a resource-scarse Iranian language [28]: in that paper, the authors project the morphological lexicon they have built for that language, disambiguate the resulting ambiguous annotation in three different ways, merge these annotations for producing a (noisy) training corpus, and train a MElt tagger based on this corpus and their lexicon. Despite the simplicity of the three disambiguation techniques, the authors report a 85.7% accuracy on a tiny gold standard using a 36-tag tagset.



(a)



(b)

Fig. 2 Contour lines for two functions of corpus and lexicon sizes: tagger accuracy and development time. In (a), the tagger accuracy is measured overall, whereas in (b) it is restricted to unknown words

These results demonstrate the relevance of developing and using a morphosyntactic lexicon for improving tagging accuracy both in the early stages of development and for long-term optimization.

7 Conclusions and perspectives

We have introduced a new MaxEnt-based tagger, MElt, that we trained on the FTB for building a tagger for French. We show that this baseline, named $\text{MElt}^{0}_{\text{fr}}$, can be significantly improved by coupling it with the French morphosyntactic lexicon Le*fff*. The resulting tagger, $\text{MElt}^{f}_{\text{fr}}$, reaches a 97.75% accuracy that are, to our knowledge, the best figures reported for French tagging, including parsing-based taggers. More precisely, the addition of lexicon-based features yield error reductions of 25% overall and of 38% for unknown words (corresponding to accuracy improvements of $.75\%$ and 5.26%, respectively) compared to the baseline tagger.

We also showed that the use of a lexicon improves the quality of the tagger at any stage of lexicon and training corpus development. Moreover, we approximately estimated development times for both resources, and show that the best way to optimize human work for tagger development is to work on the development of both an annotated corpus and a morphosyntactic lexicon.

In future work, we plan on trying and demonstrating this result in practice, by developing such resources and the corresponding $\text{MElt}^{f}_{\text{fr}}$ tagger for an under-resourced language. We also intend to study the influence of the tagset, in particular by training taggers based on larger tagsets. This work should try and understand how to benefit as much as possible from the internal structure of tags in such tagsets (gender, number, etc.).

References

1. Abeillé, A., Clément, L., Toussenel, F.: Building a treebank for French. In: A. Abeillé (ed.) Treebanks. Kluwer, Dordrecht (2003)
2. Adda, G., Mariani, J., Paroubek, P., Rajman, M., Lecomte, J.: Métrique et premiers résultats de l'évaluation grace des étiqueteurs morphosyntaxiques pour le français. In: TALN (1999)
3. Berger, A., Pietra, S.D., Pietra, V.D.: A maximum entropy approach to natural language processing. Computational Linguistics **22**(1), 39–71 (1996)
4. Candito, M., Crabbé, B., Seddah, D.: On statistical parsing of French with supervised and semi-supervised strategies. In: Proceedings of the EACL'09 workshop on Grammatical Inference for Computational linguistics. Athens, Greece (2009)
5. Chrupała, G., Dinu, G., van Genabith, J.: Learning morphology with morfette. In: Proceedings of the 6th Language Resource and Evaluation Conference. Marrakesh, Morocco (2008)
6. Crabbé, B., Candito, M.: Expériences d'analyses syntaxique statistique du français. In: Proceedings of TALN'08. Avignon, France (2008)
7. Denis, P., Sagot, B.: Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art pos tagging with less human effort. In: Proceedings of PACLIC 2009. Hong Kong, China (2009)
8. Denis, P., Sagot, B.: Exploitation d'une ressource lexicale pour la construction d'un étiqueteur morphosyntaxique état-de-l'art du français. In: Traitement Automatique des Langues Naturelles : TALN 2010. Montréal, Canada (2010). URL `http://hal.inria.fr/inria-00521231/en/`
9. Hajič, J.: Morphological Tagging: Data vs. Dictionaries. In: Proceedings of ANLP'00, pp. 94–101. Seattle, WA, USA (2000)
10. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ICML, pp. 282–289 (2001)
11. Malouf, R.: A comparison of algorithms for maximum entropy parameter estimation. In: Proceedings of the Sixth Workshop on Natural Language Learning, pp. 49–55. Taipei, Taiwan (2002)

12. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA (1999)
13. Marcus, M.P., Marcinkiewicz, M.A., Santorini, B.: Building a large annotated corpus of English: The Penn Treebank. Computational Linguistics **19**(2), 313–330 (1993)
14. Merialdo, B.: Tagging English text with a probabilistic model. Computational Linguistics **20**(2), 155–72 (1994)
15. Molinero, M.A., Sagot, B., Nicolas, L.: A morphological and syntactic wide-coverage lexicon for Spanish: The Le*ff*e. In: Proceedings of the 7th conference on Recent Advances in Natural Language Processing (RANLP 2009). Borovets, Bulgaria (2009)
16. Nasr, A., Volanschi, A.: Couplage d'un étiqueteur morpho-syntaxique et d'un analyseur partiel représentés sous la forme d'automates finis pondérés. In: TALN (2004)
17. Ratnaparkhi, A.: A maximum entropy model for part-of-speech tagging. In: Proceedings of International Conference on Empirical Methods in Natural Language Processing, pp. 133–142 (1996)
18. Ravi, S., Knight, K.: Minimized models for unsupervised part-of-speech tagging. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP'09), pp. 504–512. Singapore (2009)
19. Sagot, B.: Automatic acquisition of a Slovak lexicon from a raw corpus. In: Lecture Notes in Artificial Intelligence 3658, Proceedings of TSD'05, pp. 156–163. Springer-Verlag, Karlovy Vary, Czech Republic (2005)
20. Sagot, B.: The Le*fff*, a freely available, accurate and large-coverage lexicon for French. In: Proceedings of the 7th Language Resources and Evaluation Conference (LREC 2010). Valletta, Malta (2010)
21. Sagot, B., Clément, L., de la Clergerie, É., Boullier, P.: The Le*ffff* 2 syntactic lexicon for French: architecture, acquisition, use. In: Proceedings of the 5th Language Resource and Evaluation Conference (LREC 2006). Lisbon, Portugal (2006). URL `http://atoll.inria.fr/~sagot/pub/LREC06b.pdf`
22. Sagot, B., Walther, G., Faghiri, P., Samvelian, P.: A new morphological lexicon and a POS tagger for the Persian Language. In: International Conference in Iranian Linguistics. Uppsala, Sweden (2011). URL `http://hal.inria.fr/inria-00614711/en/`
23. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of International Conference on New Methods in Language Processing. Manchester, UK (1994)
24. Smith, N., Eisner, J.: Contrastive estimation: Training log-linear models on unlabeled data. In: Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics (ACL'05), pp. 354–362. nn Arbor, Michigan, USA (2005)
25. Spoustová, D.J., Hajič, J., Raab, J., Spousta, M.: Semi-supervised training for the averaged perceptron POS tagger. In: EACL '09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, pp. 763–771. Morristown, NJ, USA (2009)
26. Taulé, M., Martí, M., Recasens, M.: Ancora: Multilevel Annotated Corpora for Catalan and Spanish. In: Proceedings of 6th International Conference on Language Resources and Evaluation. Marrakesh, Morocco (2008)
27. Toutanova, K., Manning, C.D.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of International Conference on New Methods in Language Processing, pp. 63–70. Hong Kong (2000)
28. Walther, G., Sagot, B., Fort, K.: Fast Development of Basic NLP Tools: Towards a Lexicon and a POS Tagger for Kurmanji Kurdish. In: International Conference on Lexis and Grammar. Belgrade, Serbia (2010)