

Influence of Pre-annotation on POS-tagged Corpus Development

Karen Fort, Benoît Sagot

▶ **To cite this version:**

Karen Fort, Benoît Sagot. Influence of Pre-annotation on POS-tagged Corpus Development. The Fourth ACL Linguistic Annotation Workshop, Jul 2010, Uppsala, Sweden. pp.56–63. hal-00484294

HAL Id: hal-00484294

https://hal.archives-ouvertes.fr/hal-00484294

Submitted on 18 May 2010

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.

Influence of Pre-annotation on POS-tagged Corpus Development

Karën Fort
INIST CNRS / LIPN
Nancy / Paris, France.
karen.fort@inist.fr

Benoît Sagot
INRIA Paris-Rocquencourt / Paris 7
Paris, France.
benoit.sagot@inria.fr

Abstract

This article details a series of carefully designed experiments aiming at evaluating the influence of automatic pre-annotation on the manual part-of-speech annotation of a corpus, both from the quality and the time points of view, with a specific attention drawn to biases. For this purpose, we manually annotated parts of the Penn Treebank corpus (Marcus et al., 1993) under various experimental setups, either from scratch or using various pre-annotations. These experiments confirm and detail the gain in quality observed before (Marcus et al., 1993; Dandapat et al., 2009; Rehbein et al., 2009), while showing that biases do appear and should be taken into account. They finally demonstrate that even a not so accurate tagger can help improving annotation speed.

1 Introduction

Training a machine-learning based part-of-speech (POS) tagger implies manually tagging a significant amount of text. The cost of this, in terms of human effort, slows down the development of taggers for under-resourced languages.

One usual way to improve this situation is to automatically pre-annotate the corpus, so that the work of the annotators is limited to the validation of this pre-annotation. This method proved quite efficient in a number of POS-annotated corpus development projects (Marcus et al., 1993; Dandapat et al., 2009), allowing for a significant gain not only in annotation time but also in consistency. However, the influence of the pre-tagging quality on the error rate in the resulting annotated corpus and the bias introduced by the pre-annotation has been little examined. This is what we propose to do here, using different parts of the Penn Treebank to train various instances of a POS tagger and experiment on pre-annotation. Our goal is to assess the impact of the quality (i.e., accuracy) of the POS tagger used for pre-annotating and to compare the use of pre-annotation with purely manual tagging, while minimizing all kinds of biases. We quantify the results in terms of error rate in the resulting annotated corpus, manual annotation time and inter-annotator agreement.

This article is organized as follows. In Section 2, we mention some related work, while Section 3 describes the experimental setup, followed by a discussion on the obtained results (Section 4) and a conclusion.

2 Related Work

2.1 Pre-annotation for POS Tagging

Very few manual annotation projects give details about the campaign itself. One major exception is the Penn Treebank project (Marcus et al., 1993), that provided detailed information about the manual annotation methodology, evaluation and cost. Marcus et al. (1993) thus showed that manual tagging took twice as long as correcting pre-tagged text and resulted in twice the inter-annotator disagreement rate, as well as an error rate (using a gold-standard annotation) about 50% higher. The pre-annotation was done using a tagger trained on the Brown Corpus, which, due to errors introduced by an automatic mapping of tags from the Brown tagset to the Penn Treebank tagset, had an error rate of 7–9%. However, they report neither the influence of the training of the annotators on the potential biases in correction, nor that of the quality of the tagger on the correction time and the obtained quality.

Dandapat et al. (2009) went further and showed that, for complex POS-tagging (for Hindi and Bangla), pre-annotation of the corpus allows for a gain in time, but not necessarily in consis-

tency, which depends largely on the pre-tagging quality. They also noticed that untrained annotators were more influenced by pre-annotation than the trained ones, who showed "consistent performance". However, this very complete and interesting experiment lacked a reference allowing for an evaluation of the quality of the annotations. Besides, it only took into account two types of pre-tagging quality, high accuracy and low accuracy.

2.2 Pre-annotation in Other Annotation Tasks

Alex et al. (2008) led some experiments in the biomedical domain, within the framework of a "curation" task of protein-protein interaction. Curation consists in reading through electronic version of papers and entering retrieved information into a template. They showed that perfectly pre-annotating the corpus leads to a reduction of more than 1/3 in curation time, as well as a better recall from the annotators. Less perfect pre-annotation still leads to a gain in time, but less so (a little less than 1/4th). They also tested the effect of higher recall or precision of pre-annotation on one annotator (curator), who rated recall more positively than precision. However, as they notice, this result can be explained by the curation style and should be tested on more annotators.

Rehbein et al. (2009) led quite thorough experiments on the subject, in the field of semantic frame assignment annotation. They asked 6 annotators to annotate or correct frame assignment using a task-specific annotation tool. Here again, pre-annotation was done using only two types of pre-tagging quality, state-of-the-art and enhanced. The results of the experiments are a bit disappointing as they could not find a direct improvement of annotation time using pre-annotation. The authors reckon this might be at least partly due to "an interaction between time savings from pre-annotation and time savings due to a training effect." For the same reason, they had to exclude some of the annotation results for quality evaluation in order to show that, in line with (Marcus et al., 1993), quality pre-annotation helps increasing annotation quality. They also found that noisy and low quality pre-annotation does not overall corrupt human judgment.

On the other hand, Fort et al. (2009) claim that pre-annotation introduces a bias in named entity annotation, due to the preference given by annotators to what is already annotated, thus preventing them from noticing entities that were not pre-annotated. This particular type of bias should not appear in POS-tagging, as all the elements are to be annotated, but a pre-tagging could influence the annotators, preventing them from asking themselves questions about a specific pre-annotation.

In a completely different field, Barque et al. (2010) used a series of NLP tools, called MACAON, to automatically identify the central component and optional peripheral components of dictionary definitions. This pre-processing gave disappointing results as compared to entirely manual annotation, as it did not allow for a significant gain in time. The authors consider that the bad results are due to the quality of the tool that they wish to improve as they believe that "an automatic segmentation of better quality would surely yield some gains."

Yet, the question remains: is there a quality threshold for pre-annotation to be useful? and if so, how can we evaluate it? We tried to answer at least part of these questions for a quite simple task for which data is available: POS-tagging in English.

3 Experimental Setup

The idea underlying our experiments is the following. We split the Penn Treebank corpus (Marcus et al., 1993) in a usual manner, namely we use Sections 2 to 21 to train various instances of a POS tagger, and Section 23 to perform the actual experiments. In order to measure the impact of the POS tagger's quality, we trained it on subcorpora of increasing sizes, and pre-annotated Section 23 with these various POS taggers. Then, we manually annotated parts of Section 23 under various experimental setups, either from scratch or using various pre-annotations, as explained below.

3.1 Creating the Taggers

We used the MElt POS tagger (Denis and Sagot, 2009), a maximum-entropy based system that is able to take into account both information extracted from a training corpus and information extracted from an external morphological lexicon.[1] It has been shown to lead to a state-of-the-art POS tagger for French. Trained on Sections 2 to 21

[1] MElt is freely available under LGPL license, on the web page of its hosting project (`http://gforge.inria.fr/projects/lingwb/`).

of the Penn Treebank ($\text{MElt}_{\text{en}}^{\text{ALL}}$), and evaluated on Section 23, MElt exhibits a 96.4% accuracy, which is reasonably close to the state-of-the-art (Spoustová et al. (2009) report 97.4%). Since it is trained without any external lexicon, $\text{MElt}_{\text{en}}^{\text{ALL}}$ is very close to the original maximum-entropy based tagger (Ratnaparkhi, 1996), which has indeed a similar 96.6% accuracy.

We trained MElt on increasingly larger parts of the POS-tagged Penn Treebank,[2] thus creating different taggers with growing degrees of accuracy (see table 1). We then POS-tagged the Section 23 with each of these taggers, thus obtaining for each sentence in Section 23 a set of pre-annotations, one from each tagger.

Tagger	Nb train. sent.	Nb tokens	Acc. (%)
$\text{MElt}_{\text{en}}^{10}$	10	189	66.5
$\text{MElt}_{\text{en}}^{50}$	50	1,254	81.6
$\text{MElt}_{\text{en}}^{100}$	100	2,774	86.7
$\text{MElt}_{\text{en}}^{500}$	500	12,630	92.1
$\text{MElt}_{\text{en}}^{1000}$	1,000	25,994	93.6
$\text{MElt}_{\text{en}}^{5000}$	5,000	126,376	95.8
$\text{MElt}_{\text{en}}^{10000}$	10,000	252,416	96.2
$\text{MElt}_{\text{en}}^{\text{ALL}}$	37,990	944,859	96.4

Table 1: Accuracy of the created taggers evaluated on Section 23 of the Penn Treebank

3.2 Experiments

We designed different experimental setups to evaluate the impact of pre-annotation and pre-annotation accuracy on the quality of the resulting corpus. The subparts of Section 23 that we used for these experiments are identified by sentence ids (e.g., 1–100 denotes the 100 first sentences in Section 23).

Two annotators were involved in the experiments. They both have a good knowledge of linguistics, without being linguists themselves and had only little prior knowledge of the Penn Treebank POS tagset. One of them had previous expertise in POS tagging (Annotator1). It should also be noticed that, though they speak fluent English, they are not native speakers of the language. They were asked to keep track of their annotation time, noting the time it took them to annotate or correct each series of 10 sentences. They were also asked to use only a basic text editor, with no macro or specific feature that could help them, apart from the usual ones, like `Find`, `Replace`, etc. The set of 36 tags used in the Penn Treebank and quite a number of particular cases is a lot to keep in mind. This implies a heavy cognitive load in short-term memory, especially as no specific interface was used to help annotating or correcting the pre-annotations.

It was demonstrated that training improves the quality of manual annotation in a significant way as well as allows for a significant gain in time (Marcus et al., 1993; Dandapat et al., 2009; Mikulová and Štěpánek, 2009). In particular, Marcus et al. (1993) observed that it took the Penn Treebank annotators 1 month to get fully efficient on the POS-tagging correction task, reaching a speed of 20 minutes per 1,000 words. The speed of annotation in our experiments cannot be compared to this, as our annotators only annotated and corrected small samples of the Penn Treebank. However, the annotators' speed and correctness did improve with practice. As explained below, we took this learning curve into account, as previous work (Rehbein et al., 2009) showed it has an significant impact on the results.

Also, during each experiment, sentences were annotated sequentially. Moreover, the experiments were conducted in the order we describe them below. For example, both annotators started their first annotation task (sentences 1–100) with sentence 1.

We conducted the following experiments:

1. **Impact of the pre-annotation accuracy on precision and inter-annotator agreement:** In this experiment, we used sentences 1–400 with random pre-annotation: for each sentence, one pre-annotation is randomly selected among its possible pre-annotations (one for each tagger instance). The aim of this is to eliminate the bias caused by the annotators' learning curve. Annotation time for each series of 10 consecutive sentences was gathered, as well as precision w.r.t. the reference and inter-annotator agreement (both annotators annotated sentences 1–100 and 301–400, while only one annotated 101–200 and the other 201–300).

2. **Impact of the pre-annotation accuracy on annotation time:** This experiment is based on sentences 601–760, with pre-annotation. We divided them in series of 10 sentences.

[2] More precisely, $\text{MElt}_{\text{en}}^{i}$ is trained on the i first sentences of the overall training corpus, i.e. Sections 2 to 21.

For each series, one pre-annotation is selected (i.e., the pre-annotation produced by one of the 8 taggers), in such a way that each pre-annotation is used for 2 series. We measured the manual annotation time for each series and each annotator.

3. **Bias induced by pre-annotation:** In this experiment, both annotators annotated sentences 451–500 fully manually.[3] Later, they annotated sentences 451–475 with the pre-annotation from $\text{MElt}_{\text{en}}^{\text{ALL}}$ (the best tagger) and sentences 476–500 with the pre-annotation from $\text{MElt}_{\text{en}}^{50}$ (the second-worst tagger). We then compared the fully manual annotations with those based on pre-annotations to check if and how they diverge from the Penn Treebank "gold-standard"; we also compared annotation times, in order to get a confirmation of the gain in time observed in previous experiments.

4 Results and Discussion

4.1 Impact of the Pre-annotation Accuracy on Precision and Inter-annotator Agreement

The quality of the annotations created during experiment 1 was evaluated using two methods. First, we considered the original Penn Treebank annotations as reference and calculated a simple precision as compared to this reference. Figure 1 gives an overview of the obtained results (note that the scale is not regular).

However, this is not sufficient to evaluate the quality of the annotation as, actually, the reference annotation is not perfect (see below). We therefore evaluated the reliability of the annotation, calculating the inter-annotator agreement between Annotator1 and Annotator2 on the 100-sentence series they both annotated. We calculated this agreement on some of the subcorpora using π, aka Carletta's Kappa (Carletta, 1996)[4]. The results of this are shown in table 2.

[3]During this manual annotation step (with no pre-annotation), we noticed that the annotators used the `Find/Replace all` feature of the text editor to fasten the tagging of some obvious tokens like *the* or *Corp.*, which partly explains that the first groups of 10 sentences took longer to annotate. Also, as no specific interface was use to help annotating, a (very) few typographic errors were made, such as *DET* instead of *DT*.

[4]For more information on the terminology issue, refer to the introduction of (Artstein and Poesio, 2008).

Subcorpus	π
1-100	0.955
301-400	0.963

Table 2: Inter-annotator agreement on subcorpora

The results show a very good agreement according to all scales (Krippendorff, 1980; Neuendorf, 2002; Krippendorff, 2004) as π is always superior to 0.9. Besides, it improves with training (from 0.955 at the beginning to 0.963 at the end).

We also calculated π on the corpus we used to evaluate the pre-annotation bias (Experiment 3). The results of this are shown in table 3.

Subcorpus	**Nb sent.**	π
No pre-annotation	50	0.947
$\text{MElt}_{\text{en}}^{50}$	25	0.944
$\text{MElt}_{\text{en}}^{\text{ALL}}$	25	0.983

Table 3: Inter-annotator agreement on subcorpora used to evaluate bias

Here again, the results are very good, though a little bit less so than at the beginning of the mixed annotation session. They are almost perfect with $\text{MElt}_{\text{en}}^{\text{ALL}}$.

Finally, we calculated π throughout Experiment 2. The results are given in Figure 2 and, apart from a bizarre peak at $\text{MElt}_{\text{en}}^{50}$, they show a steady progression of the accuracy and the inter-annotator agreement, which are correlated. As for the $\text{MElt}_{\text{en}}^{50}$ peak, it does not appear in Figure 1, we therefore interpret it as an artifact.

4.2 Impact of the Pre-annotation Accuracy on Annotation Time

Before discussing the results of Experiment 2, annotation time measurements during Experiment 3 confirm that using a good quality pre-annotation (say, $\text{MElt}_{\text{en}}^{\text{ALL}}$) strongly reduces the annotation time as compared with fully manual annotation. For example, Annotator1 needed an average time of approximately 7.5 minutes to annotate 10 sentences without pre-annotation (Experiment 3), whereas Experiment 2 shows that it goes down to approximately 2.5 minutes when using $\text{MElt}_{\text{en}}^{\text{ALL}}$ pre-annotation. For Annotator2, the corresponding figures are respectively 11.5 and 2.5 minutes.

Figure 3 shows the impact on the pre-annotation type on annotation times. Surprisingly, only the worst tagger ($\text{MElt}_{\text{en}}^{10}$) produces pre-annotations that lead to a significantly slower annotation. In

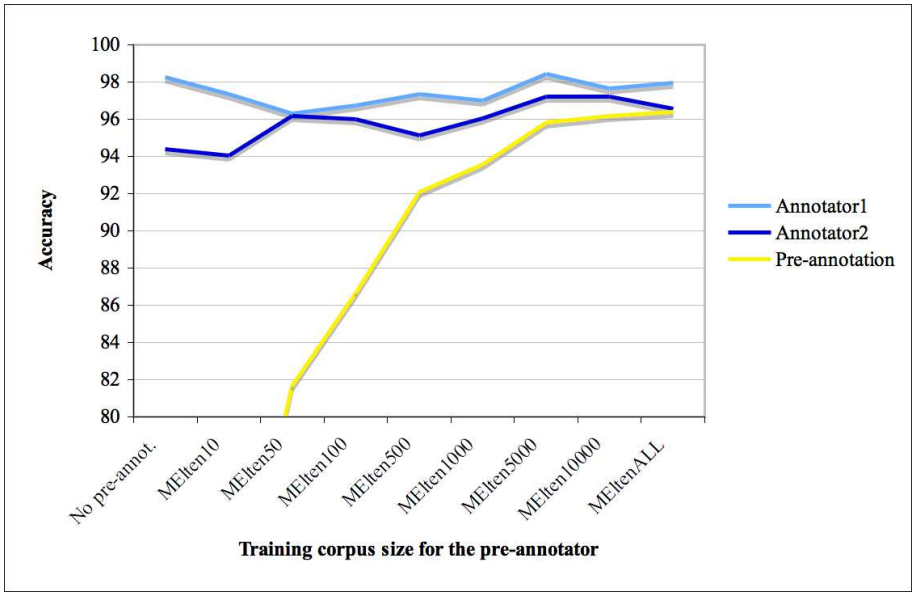


Figure 1: Accuracy of annotation

other words, a 96.4% accurate pre-annotation does not significantly speed up the annotation process with respect to a 81.6% accurate pre-annotation. This is very interesting, since it could mean that the development of a POS-annotated corpus for a new language with no POS tagger could be drastically sped up. Annotating approximately 50 sentences could be sufficient to train a POS tagger such as MElt and use it as a pre-annotator, even though its quality is not yet satisfying.

One interpretation of this could be the following. Annotation based on pre-annotations involves two different tasks: reading the pre-annotated sentence and replacing incorrect tags. The reading task takes a time that does not really depends on the pre-annotation quality. But the correction task takes a time that is, say, linear w.r.t. the number of pre-annotation errors. Therefore, when the number of pre-annotation errors is below a certain level, the correction task takes significantly less time than the reading task. Therefore, below this level, variations in the pre-annotation error rate do not lead to significant overall annotation time. Apparently, this threshold is between 66.5% and 81.6% pre-annotation accuracy, which can be reached with a surprisingly small training corpus.

4.3 Bias Induced by Pre-annotation

We evaluated both the bias induced by a pre-annotation with the best tagger, $\text{MElt}_{\text{en}}^{\text{ALL}}$, and the one induced by one of the least accurate taggers, $\text{MElt}_{\text{en}}^{50}$. The results are given in table 4 and 5, respectively.

They show a very different bias according to the annotator. Annotator2's accuracy raises from 94.6% to 95.2% with a 81.6% accuracy tagger ($\text{MElt}_{\text{en}}^{50}$) and from 94.1% to 97.1% with a 96.4% accuracy tagger ($\text{MElt}_{\text{en}}^{\text{ALL}}$). Therefore, Annotator2, whose accuracy is less than that of Annotator1 under all circumstances (see figure 1), seems to be positively influenced by pre-annotation, whether it be good or bad. The gain is however much more salient with the best pre-annotation (plus 3 points).

As for Annotator1, who is the most accurate annotator (see figure 1), the results are more surprising as they show a significant degradation of accuracy, from 98.1 without pre-annotation to 95.8 with pre-annotation using $\text{MElt}_{\text{en}}^{50}$, the less accurate tagger. Examining the actual results allowed us to see that, first, Annotator1 non pre-annotated version is better than the reference, and second, the errors made in the pre-annotated version with $\text{MElt}_{\text{en}}^{50}$ are so obvious that they can only be due to a lapse in concentration.

The results, however, remain stable with pre-annotation using the best tagger (from 98.4 to 98.2), which is consistent with the results obtained by Dandapat et al. (2009), who showed that better trained annotators are less influenced by pre-annotation and show stable performance.

When asked about it, both annotators say they felt they concentrated more without pre-

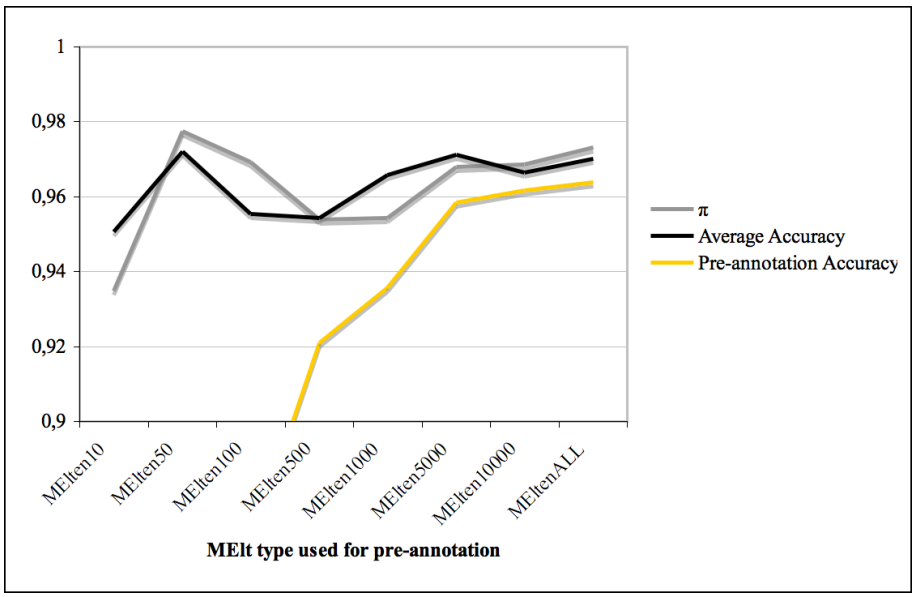


Figure 2: Annotation accuracy and π depending on the type of pre-annotation

Annotator	**No pre-annotation**	**with MElt$^{\mathrm{ALL}}_{\mathrm{en}}$**
Annotator1	98.4	98.2
Annotator2	94.1	97.1

Table 4: Accuracy with or without pre-annotation with MElt$^{\mathrm{ALL}}_{\mathrm{en}}$ (sentences 451-475)

Annotator	**No pre-annotation**	**with MElt$^{50}_{\mathrm{en}}$**
Annotator1	98.1	95.8
Annotator2	94.6	95.2

Table 5: Accuracy with or without pre-annotation with MElt$^{50}_{\mathrm{en}}$ (sentences 476-500)

annotation. It seems that the rather good results of the taggers cause the attention of the annotators to be reduced, even more so as the task is repetitive and tedious. However, annotators also had the feeling that fully manual annotation could be more subject to oversights.

These impressions are confirmed by the comparison of the contingency tables, as can be seen from Tables 6, 7 and 8 (in these tables, lines correspond to tags from the annotation and columns to reference tags; only lines containing at least one cell with 2 errors or more are shown, with all corresponding columns). For example, Annotator1 makes more random errors when no pre-annotation is available and more systematic errors when MElt$^{\mathrm{ALL}}_{\mathrm{en}}$ pre-annotations are used (typically, *JJ* instead of *VBN*, i.e., adjective instead of past participle, which corresponds to a systematic trend in MElt$^{\mathrm{ALL}}_{\mathrm{en}}$'s results).

	JJ	VBN
JJ	36	4

(Annotator 1)

	JJ	NN	NNP	NNPS	VB	VBN
JJ	36					4
NN	1	68			2	
NNP			24	2		

(Annotator 2)

Table 6: Excerpts of the contingency tables for sentences 451–457 (512 tokens) with MElt$^{\mathrm{ALL}}_{\mathrm{en}}$ pre-annotation

	IN	JJ	NN	NNP	NNS	RB	VBD	VBN
JJ		30	2					2
NNS			1	2	40			
RB	2					16		
VBD	1						17	2
WDT	2							

(Annotator 1)

	JJ	NN	RB	VBN
JJ	28	3		
NN	2	75	1	
RB	2		16	
VBN	2			10

(Annotator 2)

Table 7: Excerpts of the contingency tables for sentences 476–500 (523 tokens) with MElt$^{50}_{\mathrm{en}}$ pre-annotation

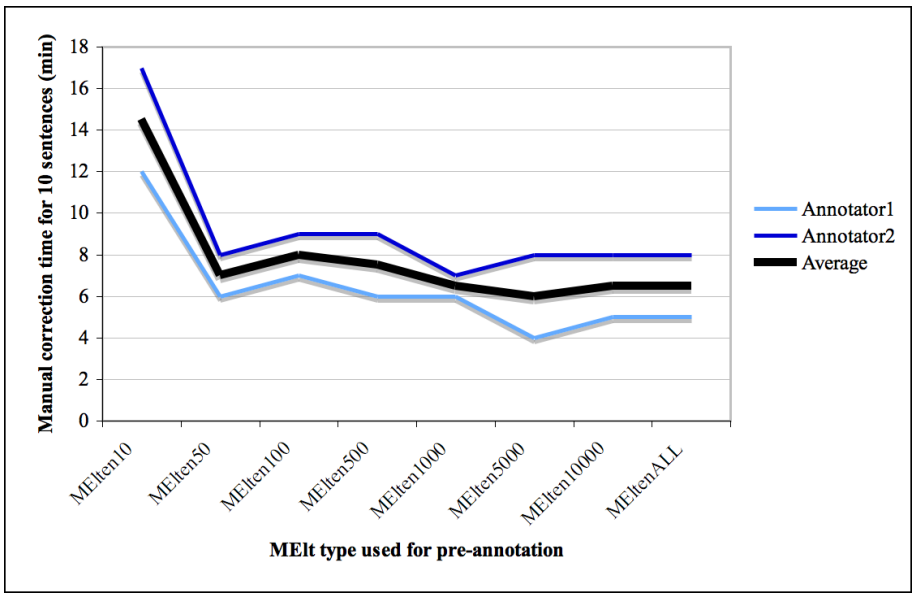


Figure 3: Annotation time depending on the type of pre-annotation

	CD	DT	JJ	NN	NNP	NNS
CD	30			2		
JJ		2	72			
NN			2	148		
NNS					3	68

(Annotator 1)

	CD	DT	IN	JJ	JJR	NN	NNP	NNS	RB	VBN
IN			104						2	
JJ		2		61		2			1	9
NN	1			4		145				
NNPS							2			
NNS						1	2	68		
RBR					2					

(Annotator 2)

Table 8: Excerpts of the contingency tables for sentences 450–500 (1,035 tokens) without pre-annotation

5 Conclusion and Further Work

The series of experiments we detailed in this article confirms that pre-annotation allows for a gain in quality, both in terms of accuracy w.r.t. a reference and in terms of inter-annotator agreement, i.e., reliability. We also demonstrated that this comes with biases that should be identified and notified to the annotators, so that they can be extra careful during correction. Finally, we discovered that a surprisingly small training corpus could be sufficient to build a pre-annotation tool that would help drastically speeding up the annotation.

This should help developing taggers for under-resourced languages. In order to check that, we intend to use this method in a near future to develop a POS tagger for Sorani Kurdish.

We also want to experiment on other, more precision-driven, annotation tasks, like complex relations annotation or definition segmentation, that are more intrinsically complex and for which there exist no automatic tool as accurate as for POS tagging.

Acknowledgments

This work was partly realized as part of the Quaero Programme[5], funded by OSEO, French State agency for innovation.

References

Beatrice Alex, Claire Grover, Barry Haddow, Mijail Kabadjov, Ewan Klein, Michael Matthews, Stuart Roebuck, Richard Tobin, and Xinglong Wang. 2008. Assisted Curation: Does Text Mining Really Help? In *Pacific Symposium on Biocomputing*.

Ron Artstein and Massimo Poesio. 2008. Inter-coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

Lucie Barque, Alexis Nasr, and Alain Polguère. 2010. From the Definitions of the Trésor de la Langue Française to a Semantic Database of the French Language. In *Proceedings of the 14th EURALEX International Congress*, Leeuwarden.

Jean Carletta. 1996. Assessing Agreement on Classification Tasks: the Kappa Statistic. *Computational Linguistics*, 22:249–254.

[5] http://quaero.org/

Sandipan Dandapat, Priyanka Biswas, Monojit Choudhury, and Kalika Bali. 2009. Complex Linguistic Annotation - No Easy Way Out! a Case from Bangla and Hindi POS Labeling Tasks. In *Proceedings of the third ACL Linguistic Annotation Workshop*.

Pascal Denis and Benoît Sagot. 2009. 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.

Karën Fort, Maud Ehrmann, and Adeline Nazarenko. 2009. Vers une méthodologie d'annotation des entités nommées en corpus ? In *Actes de la 16ème Conférence sur le Traitement Automatique des Langues Naturelles 2009 Traitement Automatique des Langues Naturelles 2009*, Senlis, France.

Klaus Krippendorff, 1980. *Content Analysis: An Introduction to Its Methodology*, chapter 12. Sage, Beverly Hills, CA.

Klaus Krippendorff, 2004. *Content Analysis: An Introduction to Its Methodology, second edition*, chapter 11. Sage, Thousand Oaks, CA.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Marie Mikulová and Jan Štěpánek. 2009. Annotation Quality Checking and its Implications for Design of Treebank (in Building the Prague Czech-English Dependency Treebank). In *Proceedings of the Eight International Workshop on Treebanks and Linguistic Theories*, volume 4-5, Milan, Italy, December.

Kimberly Neuendorf. 2002. *The Content Analysis Guidebook*. Sage, Thousand Oaks CA.

Adwait Ratnaparkhi. 1996. A Maximum Entropy Model for Part-Of-Speech Tagging. In *Proceedings of International Conference on Empirical Methods in Natural Language Processing*, pages 133–142.

Ines Rehbein, Josef Ruppenhofer, and Caroline Sporleder. 2009. Assessing the Benefits of Partial Automatic Pre-labeling for Frame-semantic Annotation. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 19–26, Suntec, Singapore, August. Association for Computational Linguistics.

Drahomíra "Johanka" Spoustová, Jan Hajič, Jan Raab, and Miroslav Spousta. 2009. 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*, pages 763–771, Morristown, NJ, USA.