Prior estimation of POST accuracy

Manuel Vilares Ferro[*,a], Víctor M. Darriba Bilbao[a], Francisco J. Ribadas Pena[a]

[a]*Department of Computer Science, University of Vigo*
Campus As Lagoas s/n, 32004 – Ourense, Spain

Abstract

We introduce an algorithm to estimate the evolution of accuracy in part-of-speech tagging on the whole of a training corpus, based on the results obtained from a portion of the latter. The technique approximates iteratively the vallue that we seek in the position desired, independently of the statistical model and dataset used. The process proves to be formally correct with respect to our working specifications and includes a stable stopping criterion. This allows the user to fix a reliable convergence threshold with respect to the accuracy finally achievable.

Our aim is to evaluate the training effort, supporting decision making in order to reduce the need for both human and computational resources during tagger generation. The proposal is of interest in at least three operational procedures. The first is the anticipation of accuracy gain, with the purpose of measuring how much work is needed to achieve a certain level of performance. The second relates the comparison between taggers at training time, with the objective of completing this task only for the tool that predictably better suits our requirements. The prediction of accuracy is also a valuable item of information for the customization of the tagger, for example to select the tag-set, since we can estimate in advance its impact on both the performance and the development costs. The experiments corroborate our initial expectations.

Key words: POST accuracy estimation, iterative methods, functional sequences, stopping criterion

Notice:. **This is the authors version of a work that was accepted for publication in Computer Speech and Language. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. In particular, the title has been modified and a deep study on the robustness of the algorithm proposed is not included here. A definitive version was subsequently published in Computer Speech and Language, 41, 2016. DOI 10.1016/j.csl.2016.06.001**

1. Introduction

Within the set of *natural language processing* (NLP) tasks, *part-of-speech* (POS) *tagging* (POST) serves as a first step for other NLP functionalities such as parsing and semantic analysis, so errors at this stage can lower their performance, which justifies the interest in studying its reliability. To evaluate this, we should determine how many of the tags provided are correct, and how many superfluous ones are eliminated (van Halteren, 1999) in the case of ambiguous outputs. The absence of the latter simplifies the question and is the most common situation. We then talk about the *accuracy* of the system (DeRose, 1988), which is also referred to as *tagger*.

As a measure of performance, accuracy provides an effective criterion to select the most suitable POST tool from a set of potential candidates for a concrete task. Tagger configuration also finds a channel of validation,

[*]Corresponding author: tel. +34 988 387280, fax +34 988 387001.
Email addresses: vilares@uvigo.es (Manuel Vilares Ferro), darriba@uvigo.es (Víctor M. Darriba Bilbao), ribadas@uvigo.es (Francisco J. Ribadas Pena)

allowing us to estimate the appropriate balance between the performance and the linguistic refinement of the system, for example with regard to tag-set definition. Finally, the gain of accuracy between two points in time serves as an indicator for the quality of the training process. Given that the generation of taggers is expensive and labor-intensive, to estimate this measure without waiting for the end of the process saves time and resources, giving a practical meaning to our proposal. In order to guarantee stable results, the algorithms proposed to solve the problem should demonstrate the correctness with respect to their specifications, which takes us away from *ad hoc* strategies.

With this aim in mind, we need to identify the factors that influence accuracy during tagger generation, which involves any variation affecting the linguistic resources, the tagset and the method of evaluation considered. Unfortunately, the number and complexity of circumstances that have to be taken into account is such that it often becomes impossible to weight them. Instead, the most practical approach seems to make our own judgments in the context of the task that we are trying to accomplish (van Halteren, 1999). A way to do this is to standarize the evaluation frame in order to analyse the generation process as a single task that compiles, relates and processes all the information provided by the user.

Tagger design is often based on a learning technique that builds a statistical model from labelled and/or unlabelled training data depending on the approach considered. Thus, we then distinguish between supervised (Brants, 2000; Brill, 1995; Daelemans et al., 1996; Giménez and Márquez, 2004; Schmid, 1994; Toutanova et al., 2003), semi-supervised (Søgaard, 2010; Spoustová et al., 2009) and unsupervised (Goldwater and Griffiths, 2007; Merialdo, 1994; Ravi and Knight, 2009) techniques. The latter constitute the only possible choice when no labelled data are available, or an alternative when they are expensive to prepare, even though no useful accuracies have been achieved. On the contrary, supervised strategies have proved to be efficient, but they require a large amount of training input that needs to be hand-annotated by human experts, which is an intensive task in terms of both time and expertise. For its part, semi-supervised methods aim to combine the advantages of the previous ones when only few labelled data are available, although the results have been mostly negative (Søgaard, 2010). In either case, *active learning* (AL) techniques (Cohn et al., 1994; Seung et al., 1992) can be used to reduce the cost of annotating training material. This explains the growing popularity of the concept in POST tasks (Dagan and Engelson, 1995; Haertel et al., 2008; Neubig et al., 2011; Ringger et al., 2007) and closely related ones such as named entity recognition (Laws and Schätze, 2008; Shen et al., 2004; Tomanek et al., 2007) or word sense disambiguation (Chan and Ng, 2007; Chen et al., 2006; Zhu, 2007), which also extends to other information extraction (Culotta and McCallum, 2005; Thompson et al., 1999), parsing (Becker and Osborne, 2005; Tang et al., 2002) or text classification (Lewis and Gale, 1994; Liere and Tadepalli, 1997; McCallum and Nigam, 1998; Tong and Koller, 2002) applications. However, despite the potential and the research effort invested in studying AL, its adoption in practice is today questionable (Attenberg and Provost, 2011). The difficulty to define both the strategy used to select the instances to be annotated at the time and the criterion to stop the learning process once performance ceases to increase, lie at the origin of the users' reluctance. This lead us to the point of departure because the stopping problem is a particular case of that of prior performance evaluation, which reinforces the interest of our approach.

As the intention is for us to provide a methodology independent of the tagger architecture and dataset used, we address accuracy estimation from the results of, and some general hypotheses about, the training process. The structure of the paper is as follows. Firstly, Section 2 examines the methodologies serving as inspiration to solve the question posed. Next, Section 3 reviews the mathematical basis necessary to support our proposal, which we introduce in Section 4. In Section 5, we describe the working frame for the practical results illustrated in Section 6. Finally, Section 7 presents our final conclusions.

2. The state of the art

To the best of our knowledge, no research on prior estimation of POST tagging accuracy has been described, although the use of learning curves for predicting performance bounds has been explored for other *machine learning* (ML) purposes. In particular, the applicability of *a priori* knowledge for restricting the form of such curves has been studied (Gu et al., 2001) and they have proved their potential to compare learning binary classification models (Perlich et al., 2003).

In the domain of NLP, this approach seems confined to the *machine translation* (MT) sphere. So, learning curves were employed to evaluate the impact of a concrete set of distortion factors on the performance of a similarly concrete operational model (Birch et al., 2008), with meagre results. They were also used to predict how many training data are required to achieve a certain level of translation accuracy (Kolachina et al., 2012). This represents a novelty with respect to the previous proposals, where the goal was only the estimation of the maximum performance, and recalls the essence of our problem formulation. Sadly, the technique has no practical sense because no stopping criterion is described in order to identify the end of the process. The degree of veracity achieved is reported as the comparison between a curve fitted from a small number of predictions, and a gold one fitted from a fine grid of observations all over the training corpus. This implies that the latter are computed in advance, which is exactly what we aim to avoid and is also not always possible.

The definition of stopping criteria is a major hurdle in the design of ML algorithms and their reliability is made subject to the convergence of the learning process, that is, its correctness. The research effort in this regard has been significant, particularly in the field of AL, whose main assumption is that instances which are harder to identify are more useful for learning. As these techniques are iterative, the most informative unlabelled examples for human annotation must be selected at each cycle, and determining the end of the process is essential. Most of this research focuses on *pool-based active learning* (Lewis and Gale, 1994), in which the selection is made from a pool and following two main sampling schemes: *uncertainty* (Lewis and Gale, 1994) and *committee-based* (Seung et al., 1992) ones. The former uses only one classifier to select the instances, making the estimation dependent upon the ML model. Thus, probabilistic models commonly use the entropy of its output, while the non-probabilistic ones often consider classification margin, as in the case of *support vector machines* (SVM) (Tong and Koller, 2002). Committee-based sampling considers a set of classifiers working on the principle of maximal disagreement among them. Common ways of estimating this are the vote-entropy (Argamon-Engelson and Dagan, 1999) and the Kullback-Leibler divergence (Becker and Osborne, 2005), always practiced on probabilistic models.

In practice, whatever the AL approach considered, to define a stopping criterion relies on a question of measuring the confidence of the classification either over a separate dataset (Vlachos, 2008) or over the unlabelled data pool (Zhu and Ma, 2012). The assumption in the first case is that we cannot take advantage of the remaining instances in the pool when that confidence drops, while in the second one the starting point is the uncertainty of the classifier. The latter can be assessed against the maximum or the overall one in the current iteration, the accuracy of the top-n selected examples (i.e. n most uncertain instances) or the minimum expected error on all future unlabelled examples (Roy and McCallum, 2001). The point where the pool becomes uninformative can also be determined through the gradient of performance (Laws and Schätze, 2008), whose rise estimation slows to an almost horizontal slope at about the time when the true one reaches its peak. We then stop the process when the gradient approaches zero.

Although these stopping criteria could inspire similar actions in other ML-based applications, we can only see them as *ad hoc* proposals because their correctness has not been proved. In order to alleviate the situation, validation windows to contrast the reliability of the results and error thresholds providing a certain degree of flexibility for variations in learning conditions are both used, but without any guarantee. At best, the dynamic update of thresholds is outlined (Zhu and Ma, 2012), in order to increase soundness across changing datasets. The idea is that if, between recent consecutive learning cycles, an unlabelled example modifies its classification, such a labeling is unstable and could change the decision boundaries. Once there are no such instances in the pool, we think the learning becomes stable, and can end. As an alternative, successive models' predictions on the stop set can be compared to see if they stabilized (Bloodgood and Vijay-Shanker, 2009), which relies on the use of validation windows.

This puts the lack of correctness as the key issue to be assessed. What is at stake is the capacity to adapt to any dataset and learning strategy, filling the gap between aggressive and conservative methods (Bloodgood and Vijay-Shanker, 2009). That means providing users with control over the behaviour of the process, eliminating the risk of stopping it too late and wasting annotation effort, or doing it too early and losing accuracy.

3. The formal framework

The aim now is to describe our abstract model on a mathematical basis that would enable us to prove its correctness. We choose a function, which must be continuous and derivable so that it provides reliability and stability, to estimate in advance the learning curve for accuracy. Since partial approximations of that function can be modelled by fitting a series of accuracy observations, it seems natural to raise its calculation as the convergence of a sequence of such approximations built incrementally while tagger training advances.

3.1. The mathematical support

We first recall some notions (Apostol, 2000) on the theory of sequences in the real metric space $(\mathbb{R}, ||)$, where $||$ denotes the Euclidean distance. For the sake of simplicity, we assume familiarity with the concepts of continuity and derivability of a real function. We denote the set of natural numbers by $\mathbb{N}$, and we assume that $0 \notin \mathbb{N}$.

Definition 1. *Let $\{x_i\}_{i \in \mathbb{N}}$ be a sequence in $(\mathbb{R}, ||)$, we say that it is a* sequence convergent to $x_0 \in \mathbb{R}$ *iff*

$$\forall \varepsilon > 0, \exists n \in \mathbb{N}, \forall i \geq n \Rightarrow |x_i, x_0| < \varepsilon \tag{1}$$

where x_0 is called the limit of $\{x_i\}_{i \in \mathbb{N}}$, *using the notation $\lim\limits_{i \to \infty} x_i = x_0$.*

A sequence converges when we can situate, from a given position, all its elements as close to the limit as we want to. It may be proved that any monotonic increasing (resp. decreasing) and upper (resp. lower) bounded sequence converges to its supremum (resp. infimum). Also, if a sequence converges, then all their sub-sequences do so to the same limit. Since we want to study the convergence of a collection of curves, we need to extend the concept to sequences of real functions.

Definition 2. *Let $\Delta := \{f : E \subseteq \mathbb{R} \to \mathbb{R}\}$ and let $\{f_i\}_{i \in \mathbb{N}}$, $f_i \in \Delta$ be a sequence of functions, we say that it is a* sequence punctually convergent to $f_0 \in \Delta$ *iff*

$$\forall x \in E, \varepsilon > 0, \exists n \in \mathbb{N}, \forall i \geq n \Rightarrow |f_i(x), f_0(x)| < \varepsilon \tag{2}$$

where f_0 is called the punctual limit of $\{f_i\}_{i \in \mathbb{N}}$, *using the notation $\lim\limits_{i \to \infty}{}^{p} f_i = f_0$.*

A sequence of functions is punctually convergent if the sequence of their values on each point converges. This implies that we can calculate the limit point-to-point, although the speed of convergence may be different in each case, which poses a threat when dealing with prediction tasks since the results obtained could vary greatly even over points close to the observations. We then need a criterion guaranteeing a uniform behaviour.

Definition 3. *Let $\Delta := \{f : E \subseteq \mathbb{R} \to \mathbb{R}\}$ and let $\{f_i\}_{i \in \mathbb{N}}$, $f_i \in \Delta$ be a sequence of functions, we say that it is a* sequence uniformly convergent to $f_0 \in \Delta$ *iff*

$$\forall \varepsilon > 0, \exists n \in \mathbb{N}, \forall i \geq n \Rightarrow |f_i(x), f_0(x)| < \varepsilon, \forall x \in E \tag{3}$$

where f_0 is called the uniform limit of $\{f_i\}_{i \in \mathbb{N}}$, *using the notation $\lim\limits_{i \to \infty}{}^{u} f_i = f_0$.*

This identifies functional sequences for which all points converge at the same pace to the limit, which allows its continuity to be inferred when the curves in the sequence are continuous. Such a property is extremely useful to us since continuity is a guarantee of stability, matching small variations in the input to small ones in the output. In order to estimate the rate of these variations, we must turn our attention to the concept of derivability, which is not simply preserved by uniform convergence.

Theorem 1. *Let $\{f_i\}_{i \in \mathbb{N}}$ be a sequence of derivable real functions $f_i : (a, b) \to \mathbb{R}$. Suppose the existence of both $x_0 \in (a, b)$, such that $\{f_i(x_0)\}_{i \in \mathbb{N}}$ converges, and $g := \lim\limits_{i \to \infty}{}^{u} f'_i$ in (a, b). Then $f := \lim\limits_{i \to \infty}{}^{u} f_i$ exists and $f'(x) = g(x)$, in (a, b).*

Proof. See (Apostol, 2000), as for previous results and definitions. ■

3.2. The working hypotheses

As the use of partial fitted approximations to the learning curve for accuracy is at the basis of our proposal, we need to gather as much information as possible about the nature of accuracy, in order to alleviate the risk of producing meaningless results. The point of departure for its calculation is a tagger to be evaluated and a corpus used for both training and testing purposes, without specific hypotheses in the former case. With regard to the second, it must meet certain conditions in order to bring reliability to the process. Basically, we assume the corpus evenly distributes all the styles and language variations present in it, in order to obtain a predictable progression of the estimation trace for accuracy over a virtually infinite interval. This does not imply a loss of generality since we can re-order the data and take as large a corpus as we want.

In these conditions, we accept that the learning curve for accuracy is a positive definite and strictly increasing function on $\mathbb{N}$, upper bounded by 100. We also assume that the speed of increase is higher in its first stretch, where the learning is faster, decreasing as the training process advances and giving the curve a concave nature, giving rise to a horizontal asymptote below its maximum. We can observe all this in the left-most curve of Fig. 1, which shows the observations of the *lookahead part-of-speech* (LAPOS) tagger (Tsuruoka et al., 2011) on the *Freiburg-Brown* (Frown) corpus of American English (Hinrichs et al., 2010) in $[5 * 10^3, 8 * 10^5]$.

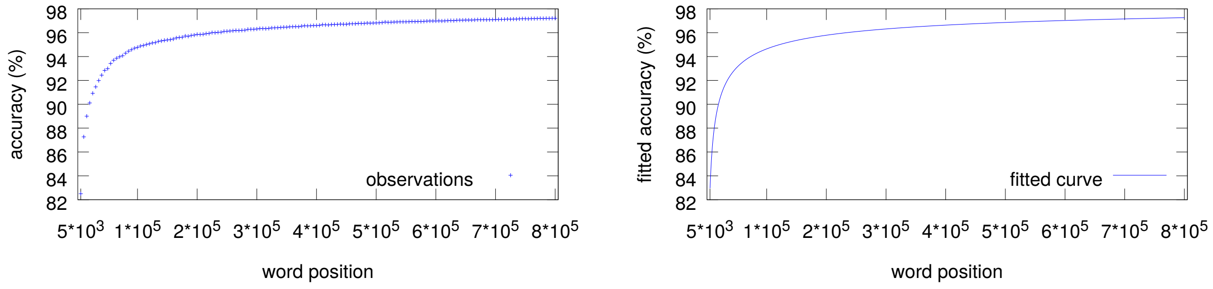

Figure 1: Accuracy for LAPOS on Frown corpus, and a fitted curve

Although we may argue that such requirements cannot be completely guaranteed in practice due to the nature of real corpora, the objective of these working hypotheses is to provide a decidable computational domain for the problem posed. This process of idealization is inherent in the scientific modeling and it is the only way to deal with unpredictable variations in the conditions of calculation. An example of the latter are those depending on human decisions, as in the above-mentioned case of corpora building.

3.3. The notational support

Having identified the context of the problem, we need to formally capture the data structures we are going to work with, such as the collection of data sets on which the accuracy measurements are applied, ideally up to the length of the original corpus. We opt for an incremental sampling.

Definition 4. *Let $\mathscr{C}$ be a corpus, $\mathscr{K} \subsetneq \mathscr{C}$ a proper and not empty subset of initial sentences from $\mathscr{C}$, and $\sigma \in \mathbb{N}$. We define a* learning scheme *for $\mathscr{C}$* with kernel *$\mathscr{K}$* and step *σ, as a triple $\mathscr{C}_\sigma^\mathscr{K} = [\mathscr{K}, \sigma, \{\mathscr{C}_i\}_{i \in \mathbb{N}}]$, such that:*

$$\mathscr{C}_1 = \mathscr{K} \text{ and } \mathscr{C}_i = \mathscr{C}_{i-1} \cup \mathscr{I}_i, \ \mathscr{I}_i \subset \mathscr{C} \setminus \mathscr{C}_{i-1}, \ |\mathscr{I}_i| = \|i * \sigma\| - \|(i-1) * \sigma\|, \ \forall i \geq 2 \tag{4}$$

where $|\mathscr{I}_i|$ is the cardinal (words) of $\mathscr{I}_i$ and $\|n\|$ is the position of the first sentence-ending beyond the n-th word. The latter allows us to adapt the step, avoiding sentence truncation. We refer $\mathscr{C}_i$ as an individual *of $\mathscr{C}_\sigma^\mathscr{K}$.*

The kernel $\mathscr{K}$ delimits a training portion of corpus we believe to be enough to initiate consistent evaluations for accuracy. With regard to the learning scheme, it is to be hoped that the estimation process is better when it is fine-grained, but we need to choose a functional pattern to fit accuracy. In order to situate ourselves in the context of our mathematical support, we consider real C-infinity curves, which guarantees the existence of derivatives of all orders. In other words, we focus on smooth functions.

Definition 5. *Let $C^{\infty}_{(0,\infty)}$ be the real C-infinity functions in the domain $(0,\infty)$ and $n \in \mathbb{N}$, we say that $\pi : \mathbb{R}^{+^n} \to C^{\infty}_{(0,\infty)}$ is an* accuracy pattern *iff $\pi(a_1,\ldots,a_n)$ is positive definite, concave and strictly increasing.*

We can immediately infer from the definition some properties of accuracy patterns. In particular, these functions are upper bounded by a horizontal asymptotic value. To illustrate the concept, we take the power family of curves $\pi(a,b,c)(x) := -a * x^{-b} + c$, hereinafter used as running example. They have $\lim\limits_{x\to\infty} \pi(a,b,c)(x) = c$ as horizontal asymptote and verify:

$$\pi(a,b,c)'(x) = a * b * x^{-(b+1)} > 0 \qquad\qquad \pi(a,b,c)''(x) = -a * b * (b+1) * x^{-(b+2)} < 0 \qquad (5)$$

which guarantees increase and concavity in $(0,\infty)$, respectively. All this is illustrated in the right-most curve of Fig. 1 for values $a = 675.6588$, $b = 0.4376$ and $c = 99.0242$.

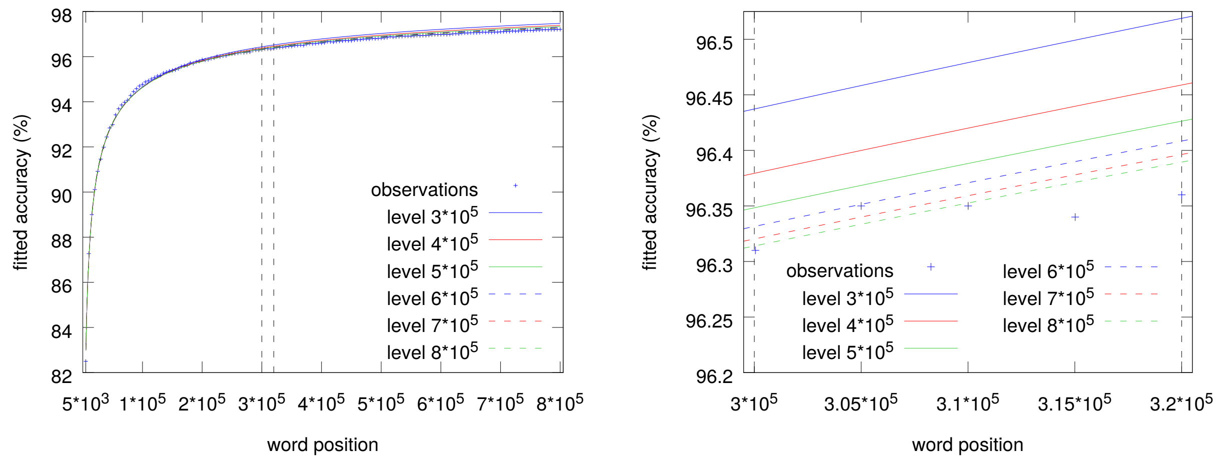

Figure 2: Learning trace and instantaneous configurations for LAPOS on Frown, with details in zoom

Given a tagger $\mathscr{T}$ and a learning scheme $\mathscr{C}^{\mathscr{K}}_{\sigma}$, the goal of an accuracy pattern is to provide functions to fit the sequence of observations $\{[x_i, Ac(\mathscr{T},\mathscr{C})(x_i)],\ x_i = |\mathscr{C}_i|\}_{i=1}^{\lambda}$ together with $(\infty, 100)$, where $Ac(\mathscr{T},\mathscr{C})(x_i)$ is the accuracy of $\mathscr{T}$ achieved on the corpus $\mathscr{C}$ at the word position x_i, and the last pair represents the maximum value expected at the point of infinity. Since parameter assignment is a complex task, we should automate it, for example using a *trust region method* (Branch et al., 1999). This minimizes the summed square of *residuals*, in other words, the differences between the observed values and the fitted ones.

Definition 6. *Let $\mathscr{T}$ be a tagger, $\mathscr{C}^{\mathscr{K}}_{\sigma}$ a learning scheme, π an accuracy pattern and $\lambda \in \mathbb{N}$. We define the* learning trend of level λ for $\mathscr{T}$ on $\mathscr{C}^{\mathscr{K}}_{\sigma}$ using π, *as a curve $\mathscr{T}_{\lambda}(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi) \in \pi$, fitting the sequence $\{[x_i, \mathrm{Ac}(\mathscr{T},\mathscr{C})(x_i)], x_i = |\mathscr{C}_i|\}_{i=1}^{\lambda} \cup (\infty, 100)$.*

A sequence of learning trends $\mathscr{T}(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi) := \{\mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)\}_{i\in\mathbb{N}}$ is called a learning trace*, from which we can extract an* instantaneous configuration *of the form $\mathscr{T}(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)(x) := \{\mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)(x)\}_{i\in\mathbb{N}},\ x \in \{|\mathscr{C}_j|\}_{j\in\mathbb{N}}$.*

We denote by $\rho_i(j) := \mathrm{Ac}(\mathscr{T},\mathscr{C})(x_j) - \mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)(x_j),\ x_j = |\mathscr{C}_j|$ the residual of $\mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)$ at the level j. *The* residual of $\mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)$ at the point of infinity *is the difference of the maximum possible accuracy with the asymptote, $\rho_i^{\infty} := 100 - \lim\limits_{x\to\infty} \mathscr{T}_i(\mathscr{C}^{\mathscr{K}}_{\sigma}, \pi)$.*

Although learning trends fit accuracy beyond its own level, a reliable prediction requires an understanding of the evolution of the learning curve over the corpus. Learning traces solve this question, allowing the extraction of a sequence of fitted values in a word position, representing the evolution of the estimation at that instant. Continuing with the example for the tagger LAPOS and the corpus Frown, Fig. 2 shows a learning trace on the learning scheme with kernel and step $5 * 10^3$, together with two instantaneous configurations at positions $3 * 10^5$ and $3.2 * 10^5$, including a more detailed view.

4. The abstract model

We assume $\mathscr{T}$ a tagger, a learning scheme $\mathscr{C}_\sigma^{\mathscr{K}}$ and an accuracy pattern π. Our aim is to estimate reliably and as far in advance as possible the behaviour of the learning curve for accuracy. We start by computing the learning trace $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi) := \{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ in order to study its evolution over the corpus, through the sequences of instantaneous configurations $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x) := \{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)\}_{i \in \mathbb{N}}$, $x \in \{|\mathscr{C}_j|\}_{j \in \mathbb{N}}$. For that purpose, we first prove some properties of learning traces.

4.1. Some properties of the learning traces

We just work with those results of interest for this work and taking $(0, \infty)$ as domain.

Theorem 2. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace. We have $\lambda \in \mathbb{N}$, the* level of prediction *for $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$, and $x_\lambda < |\mathscr{C}_\lambda|$ such that $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq \lambda}$ is strictly decreasing in (x_λ, ∞) and upper bounded by 100.*

Proof. Having fixed a level $i \in \mathbb{N}$, the fitting algorithm minimizes a weighting function on the set of residuals $\{\rho_i(j)\}_{j \leq i} \cup \{\rho_i^\infty\}$ in order to generate the learning trend $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$. Since the maximum data fitted corresponds to the point of infinity and both observations and learning trends are positive definite, concave and increasing in $(0, \infty)$, the impact of that singularity decreases as the levels ascend. This means that $\{\rho_i^\infty\}_{i \in \mathbb{N}}$ is increasing and, as a consequence, the sequence $\{\alpha_i\}_{i \in \mathbb{N}}$ of horizontal asymptotes associated to $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ is strictly decreasing.

Accordingly, the learning trends $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ either never intersect in $(0, \infty)$, or each curve $\mathscr{T}_j(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ intersects any other one $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$, $i \neq j$ at a single point (x_j^i, y_j^i). In the former case, as $\{\alpha_i\}_{i \in \mathbb{N}}$ is strictly decreasing, so is the sequence $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ and the thesis becomes trivial.

In the case of intersection, the relative position of the curves is reversed once it has taken place and $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ then rises above $\mathscr{T}_j(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ in (x_j^i, ∞), if $i < j$. Since $\{x_j^i\}_{i \in \mathbb{N}}$ is strictly decreasing, if $\mathscr{T}_j(\mathscr{C}_\sigma^{\mathscr{K}}, \pi) \cap \mathscr{T}_{j+1}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ occurs before the level j, that is $x_j^{j+1} < |\mathscr{C}_j|$, the sequence $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq j}$ is also in (x_j^{j+1}, ∞).

To prove the thesis is then sufficient to demonstrate that a $\lambda \in \mathbb{N}$ exists such that $x_\lambda := x_\lambda^{\lambda+1} < |\mathscr{C}_\lambda|$ and $\rho_\lambda^\infty \geq 0$. The former condition is met, since $\lim\limits_{j \to \infty} |\mathscr{C}_j| = \infty$ and $\{x_j^{j+1}\}_{j \in \mathbb{N}}$ is strictly decreasing. The second condition satisfies because $\{\rho_i^\infty\}_{i \in \mathbb{N}}$ is decreasing and if a level ξ verifies the first one, then $\lambda \geq \xi$ so does. ■

Although the condition on the upper bound does not have any impact on the convergence problem we are going to study, it allows us to focus in those learning trends that could be considered as feasible approximations for accuracy, whose maximum value is also upper bounded by 100. As illustration, we again refer to Fig. 2, which suggests the uniform convergence of the learning trace. In order to formally prove it, we first deal with the punctual case.

Theorem 3. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda < |\mathscr{C}_\lambda|$, $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi) := \lim\limits_{i \to \infty}{}^{p}\, \mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exists in (x_λ, ∞).*

Proof. Since, by Theorem 2, an $x_\lambda < |\mathscr{C}_\lambda|$ exists such that $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq \lambda}$ is decreasing in (x_λ, ∞), we conclude that so is $\forall x > x_\lambda$, $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)\}_{i \geq \lambda}$. As curves in $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ are positive definite, $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)\}_{i \geq \lambda}$ is lower bounded and converges $\forall x > x_\lambda$. ■

Unfortunately, the punctual convergence does not permit a common convergence threshold to be fixed with respect to the limit function $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ from a given level in the learning trace $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$. Such a calculation needs to be solved separately for each instantaneous configuration, which makes no practical sense for us. To solve this we need the uniform convergence, whose demonstration from Theorem 1 requires us first to prove that all the hypotheses are verified and, in particular, the uniform convergence of the sequence for the derivatives. We start with the punctual case.

Theorem 4. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda < |\mathscr{C}_\lambda|$, $\lim\limits_{i \to \infty}{}^p \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exists in (x_λ, ∞).*

Proof. As the learning trends $\{\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ are concave and strictly increasing in $(0, \infty)$, their derivatives are positive definite and, therefore, lower bounded. On the other hand, by Theorem 2, an $x_\lambda < |\mathscr{C}_\lambda|$ exists such that $\{\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq \lambda}$ is strictly decreasing in (x_λ, ∞), which means that so is the sequence $\{\mathscr{T}'(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq \lambda}$. The thesis then becomes trivial. ■

We are now ready to prove the uniform convergence for the sequence of derivatives associated to a learning trace.

Theorem 5. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda < |\mathscr{C}_\lambda|$, $\lim\limits_{i \to \infty}{}^u \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exists in (x_λ, ∞).*

Proof. By Theorem 4, $\lim\limits_{i \to \infty}{}^p \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exists in (x'_λ, ∞) for some $x'_\lambda < |\mathscr{C}_\lambda|$. We then have that

$$\forall x > x'_\lambda, \varepsilon > 0, \exists n_x \in \mathbb{N}, \forall i \geq n_x \Rightarrow |\mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x), \lim_{i \to \infty}{}^p \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)| < \varepsilon \tag{6}$$

On the other hand, the learning trends are concave and strictly increasing in $(0, \infty)$, with a horizontal asymptote. This means that their derivatives are convex and strictly decreasing in this interval, with a horizontal asymptote in $y = 0$. Furthermore, as previously established in the proof of Theorem 4, the sequence $\{\mathscr{T}'(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \geq \lambda}$ is strictly decreasing in (x'_λ, ∞). Consequently, the lower $x \in (x'_\lambda, \infty)$ is, the higher the distances between elements in $\{\mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)\}_{i \geq \lambda}$ are.

Since to prove the uniform convergence we only need to demonstrate that for some $x_\lambda < |\mathscr{C}_\lambda|$

$$\forall \varepsilon > 0, \exists n \in \mathbb{N}, \forall i \geq n \Rightarrow |\mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x), \lim_{i \to \infty}{}^p \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)| < \varepsilon,\ \forall x > x_\lambda \tag{7}$$

It is sufficient, for example, to take $x_\lambda := x'_\lambda + \frac{|\mathscr{C}_\lambda| - x'_\lambda}{2}$ and $n := n_{x_\lambda}$ to conclude the thesis. ■

At this point, all the hypotheses of Theorem 1 with respect to the functions in a learning trace have been proved, and we can apply it to prove its uniform convergence.

Theorem 6. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda < |\mathscr{C}_\lambda|$, $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi) := \lim\limits_{i \to \infty}{}^u \mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ and $\mathscr{T}'{}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi) = \lim\limits_{i \to \infty}{}^u \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exist in (x_λ, ∞).*

Proof. Let's take $x_\lambda < |\mathscr{C}_\lambda|$ from Theorem 5, then Theorem 3 proves that $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)\}_{i \geq \lambda}$ converges in (x_λ, ∞). The conclusion is then trivial from Theorem 1 since $\lim\limits_{i \to \infty}{}^u \mathscr{T}'{}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ exists in that interval. ■

Once the existence and derivability of $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ have been proved, we can now demonstrate that it preserves the fundamental properties of the learning trends.

Theorem 7. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda < |\mathscr{C}_\lambda|$, $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ is continuously derivable, concave, strictly increasing and positive definite in (x_λ, ∞).*

Proof. The result is a trivial corollary from Theorem 6 and the fact that the learning trends $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)\}_{i \in \mathbb{N}}$ verify those properties by definition in $(0, \infty)$. ■

4.2. The methodology

Accuracy is computed as the maximum value reached by its learning curve over an incremental sequence of observation levels $\{|\mathscr{C}_i|\}_{i\in\mathbb{I}}$, $\mathbb{I}\subseteq\mathbb{N}$. Because ideally the corpus can be as large as we want, and these observations are increasing and concave, we can identify this maximum with the ordinate of an horizontal asymptote. Thus, accuracy may be seen as the value in the point at infinity of its learning curve. On the other hand, a learning trend $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$ fits the accuracy at point $x=|\mathscr{C}_i|$ from the observations made up to that level. So, in order to fit its value in the point at infinity, we should compute a learning trend throughout the entire sequence of levels $\{|\mathscr{C}_i|\}_{i\in\mathbb{N}}$, which will also enable us to estimate it at each intermediate level of the learning curve.

The methodology we have just outlined implies computing the horizontal asymptote for $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$, whose existence is guaranteed by Theorem 7. A major challenge in achieving this is the absence of an algebraic expression for the learning trace $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$, since its elements are fitted as the observations advance. However, since each learning trend refines the fitting processes corresponding to those of the lower level, one way to address the question is to consider it as an iterative process, in which the step $i\in\mathbb{N}$ corresponds to the calculation of $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$. The consistency of this approach depends on the availability of a criterion that interrupts the process once a convergence threshold set by the user is reached. Given that we are looking for the horizontal asymptote of $\mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$, to define such a criterion means finding a variable capable of measuring the correction to be applied to achieve it from an iteration $i\in\mathbb{N}$. The previous results suggest associating $\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)$, the slope at the last observation fitted in the iteration i, to that variable. Indeed, the learning trace is decreasing from the level of prediction and uniformly convergent, while the learning curves for accuracy and their limit are continuously derivable and increasing. Thus, once a slope has been reached by a learning trend beyond that level, all subsequent calculations provide lower positive values.

Theorem 8. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$ be a learning trace with a level of prediction λ, then for some $x_\lambda<|\mathscr{C}_\lambda|$:*

1. $\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\geq\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(y)\geq 0,\ \forall y>x\geq|\mathscr{C}_\lambda|>x_\lambda$
2. $\forall i\in\mathbb{N},\ \mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\geq\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(y)>0,\ \forall y>x\geq|\mathscr{C}_\lambda|>x_\lambda$
3. $\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\geq\mathscr{T}'_j(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\geq\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x),\forall j\geq i\geq\lambda,x\geq|\mathscr{C}_\lambda|>x_\lambda$

PROOF. Let's take $x_\lambda<|\mathscr{C}_\lambda|$ from Theorem 5, then (2) is trivial and we conclude (1) directly from Theorem 7. As $\{\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)\}_{i\geq\lambda}$ is decreasing and $\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)=\lim\limits_{i\to\infty}{}^{u}\,\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$ in (x_λ,∞) by Theorem 6, (3) is trivial. ■

This result allows us to define a stable stopping criterion for our iterative technique if we can demonstrate that the value of the slope over the asymptote, which is zero, is reached at the same time that the convergence of the learning trace takes place.

Theorem 9. *Let $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$ be a learning trace, then $\lim\limits_{i\to\infty}\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)=\lim\limits_{x\to\infty}\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)$*

PROOF. Let λ be the level of prediction for $\mathscr{T}(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$. By Theorem 8 both sequences $\{\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\}_{x\geq|\mathscr{C}_\lambda|}$ and $\{\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)\}_{i\geq\lambda}$ are decreasing and lower bounded by 0 and, therefore, converge. Following Theorem 4

$$\lim_{i\to\infty}\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)=\lim_{i\to\infty}\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|) \tag{8}$$

and, since $\{\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)\}_{i>\lambda}$ is a sub-sequence of $\{\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x)\}_{x>|\mathscr{C}_\lambda|}$, we deduce that

$$\lim_{i\to\infty}\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(|\mathscr{C}_i|)=\lim_{x\to\infty}\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)(x) \tag{9}$$

which proves the thesis. ■

We now establish the correctness of our proposal, proving that the iterative process brings us as close to the horizontal asymptote of $\mathscr{T}'_\infty(\mathscr{C}_\sigma^{\mathscr{K}},\pi)$ as we like.

Theorem 10. (Correctness) *Let $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)$ be a learning trace with a level of prediction λ, then $\mathscr{T}_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(|\mathscr{C}_i|)$, $i \geq \lambda$ approximates $\lim_{x \to \infty} \mathscr{T}_\infty(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(x)$ with $\mathscr{T}'_i(\mathscr{C}_\sigma^{\mathscr{K}}, \pi)(|\mathscr{C}_i|)$ as convergence threshold on the slope.*

PROOF. Trivial from Theorems 5, 8 and 9. ■

5. A working frame

We need a collection of corpora on a set of target languages, tag-sets and taggers, an accuracy pattern and a methodology to compute the accuracy values we use for both computing and validating predictions. As target languages we consider English and Spanish. The former is the most studied and best-understood one, which contributes to providing reliability and a high number of case studies. For its part, Spanish is characterized by a complex derivational paradigm and is one of the languages with the highest growth and development potential in the domain of NLP. With regard to the corpora, we select them together with their associated tag-sets from the most popular ones in the domain for each target language:

1. The AnCora (Taulé et al., 2008) treebank includes a section for Spanish, previously used as a resource in the shared tasks of CoNNL (Buchholz and Marsi, 2006; Hajič et al., 2009) and SemEval (Recasens et al., 2010). It has served as a training and testing resource for POST (Hulden and Francom, 2012), parsing (Popel et al., 2013) and semantic annotation (Mukund et al., 2010) tasks. Since its tag-set has been developed for languages morphologically richer than English, AnCora has the most detailed annotation of the corpora considered and is the only one to follow the Eagles recommendations (Monachini and Calzolari, 1996). Its 280 tags (Taulé et al., 2008) cover the main POS classes used in Spanish as well as sub-classes and morphological features, accessible on *clic.ub.edu/corpus/webfm_send/18.*
2. The Freiburg-Brown of American English (Mair and Leech, 2007) (Frown) matches the composition and style of the Brown corpus (Francis and Kučera, 1967,71,79). It has been used in linguistic studies with a more theoretical purpose (Leech, 2009; Mair, 2006), which in our opinion entails an interesting counterpoint to the other two corpora considered, which are more oriented to NLP-related applications. The associated tag-set is the UCREL C8. With 169 tags accessible on *ucrel.lancs.ac.uk/claws8tags.pdf*, it was selected as the common tag-set for the Brown family of corpora (Hinrichs et al., 2010).
3. The section with news items from the *Wall Street Journal* (WSJ) included in the Penn treebank (Marcus et al., 1999), is a popular corpus in NLP for both POST (Brants, 2000; Collins, 2002; Giménez and Márquez, 2004; Ratnaparkhi, 1996; Toutanova et al., 2003) and parsing purposes (Charniak, 2000; Petrov et al., 2006). It has also been used in the shared tasks of prestigious events in the domain of natural language learning, such as CoNNL (Hajič et al., 2009; Nivre et al., 2007; Surdeanu et al., 2008), and semantic evaluation, such as SemEval (Yuret et al., 2010). The tag-set associated to this corpus has 45 tags, accessible on *www.comp.leeds.ac.uk/ccalas/tagsets/upenn.html* and covering basic POS classes in English along with some morphological information. This is the simplest of the tag-sets we consider.

Both Penn and AnCora are treebanks annotated with POS tags as well as syntactic structures. By stripping them of the latter, they can be used to train POST systems.

With respect to the taggers, we avoid rule-based taggers, where the absence of a training phase leaves the consideration of accuracy as a predictable magnitude void of content. We then focus on supervised techniques. In contrast to unsupervised ones, those make it possible to work with predefined tag-sets, which facilitates both the evaluation and the comprehension of the results. Furthermore, this kind of strategies provides the best quality results, placing them as a reference for any testing purpose. We have selected a broad range of proposals covering the most representative architectures:

1. In the category of stochastic methods and as representative of the *hidden markov models* (HMMs), we choose TnT (Brants, 2000). We also include here the TreeTagger (Schmid, 1994), which uses decision trees to generate the HMM, and Morfette (Chrupala et al., 2008), an averaged perceptron

approach (Collins, 2002). To illustrate the *maximum entropy models*, we have chosen to work with MXPOST (Ratnaparkhi, 1996) and the MEM associated to Apache OpenNLP (OpenNLP MaxEnt) (see *opennlp.apache.org/*). Finally, the Stanford POS tagger (Toutanova et al., 2003), is based on a *conditional markov model*, which combines features of HMMs and MEMs.

2. Under the heading of other POST methods, the possibilities are many and various. As an example of transformation-based learning, we take fnTBL (Ngai and Florian, 2001), an updated version of the classic Brill (Brill, 1995). In relation to memory-based learning, the representative is the *memory-based tagger* (MBT) (Daelemans et al., 1996), while we chose SVMTool (Giménez and Márquez, 2004) to describe the behaviour with respect to a support vector machine technique. Finally, we use a perceptron-based training method with lookahead, through LAPOS (Tsuruoka et al., 2011).

As regards the computation of each observation for accuracy, we opt for a k-fold cross validation (Clark et al., 2010), due to its good adaptation to small data sets, a key advantage in our particular context.

		Kernel (words)	Step	Threshold (degrees)	CLevel	Control Level (word position)						MAPE	DMR	Control Level (word position)						MAPE	DMR
						$\|3 * 10^5\|$		$\|4 * 10^5\|$		$\|5 * 10^5\|$				$\|6 * 10^5\|$		$\|7 * 10^5\|$		$\|8 * 10^5\|$			
						Ac	EAc	Ac	EAc	Ac	EAc			Ac	EAc	Ac	EAc	Ac	EAc		
fnTBL	AnCora	5000	500	$9.50 * 10^{-4}$	109002	96.67	96.66	96.95	96.98	97.15	97.20	0.04	100.00								
	Frown	10000	500	$5.60 * 10^{-4}$	207519	94.98	95.18	95.41	95.66	95.78	96.00	0.24	100.00	95.98	96.26	96.18	96.47	96.31	96.63	0.27	77.78
	Penn	2000	125	$2.55 * 10^{-4}$	209396	96.10	96.11	96.28	96.33	96.43	96.47	0.05	87.50	96.53	96.58	96.64	96.67	96.72	96.74	0.04	100.00
LAPOS	AnCora	5000	1000	$7.25 * 10^{-4}$	100034	97.55	97.60	97.75	97.81	97.90	97.94	0.05	100.00								
	Frown	5000	500	$5.20 * 10^{-4}$	175535	96.31	96.48	96.60	96.83	96.82	97.07	0.22	100.00	96.98	97.25	97.10	97.39	97.21	97.51	0.25	100.00
	Penn	2000	125	$2.60 * 10^{-4}$	175149	96.81	96.91	96.97	97.08	97.07	97.19	0.12	100.00	97.15	97.28	97.22	97.35	97.28	97.40	0.13	100.00
MaxEnt	AnCora	5000	500	$1.13 * 10^{-3}$	92007	96.23	96.38	96.57	96.66	96.77	96.86	0.12	100.00								
	Frown	10000	125	$1.05 * 10^{-3}$	131254	94.32	94.66	94.76	95.16	95.11	95.51	0.41	77.78	95.33	95.77	95.52	95.98	95.69	96.16	0.44	77.78
	Penn	5000	500	$4.38 * 10^{-4}$	146501	95.95	96.05	96.17	96.27	96.34	96.42	0.11	77.78	96.45	96.54	96.55	96.63	96.63	96.70	0.09	87.50
MBT	AnCora	10000	500	$8.21 * 10^{-4}$	107023	96.10	96.16	96.40	96.47	96.63	96.69	0.06	100.00								
	Frown	10000	1000	$7.30 * 10^{-4}$	181019	93.58	93.71	94.07	94.25	94.52	94.64	0.16	100.00	94.77	94.93	94.97	95.17	95.17	95.36	0.17	100.00
	Penn	2000	125	$6.23 * 10^{-4}$	149896	95.24	95.27	95.56	95.62	95.76	95.86	0.07	100.00	95.89	96.05	96.05	96.20	96.13	96.32	0.12	100.00
Morfette	AnCora	10000	500	$5.80 * 10^{-4}$	121519	97.18	97.19	97.40	97.42	97.52	97.57	0.03	100.00								
	Frown	2000	500	$5.10 * 10^{-4}$	196006	95.65	95.78	95.97	96.19	96.23	96.48	0.22	87.50	96.39	96.70	96.54	96.88	96.69	97.02	0.28	87.50
	Penn	5000	1000	$3.17 * 10^{-4}$	167010	96.47	96.74	96.63	96.93	96.74	97.06	0.31	87.50	96.81	97.16	96.91	97.24	96.96	97.30	0.33	87.50
MXPOST	AnCora	5000	500	$1.14 * 10^{-3}$	106025	96.56	96.65	96.86	97.00	97.08	97.24	0.13	88.89								
	Frown	10000	500	$7.90 * 10^{-4}$	169009	94.75	94.75	95.18	95.24	95.49	95.58	0.06	100.00	95.74	95.84	95.95	96.04	96.09	96.21	0.09	100.00
	Penn	10000	250	$3.68 * 10^{-4}$	170024	96.11	96.17	96.35	96.38	96.52	96.53	0.03	100.00	96.59	96.63	96.69	96.72	96.74	96.78	0.03	100.00
stanford	AnCora	5000	1000	$9.00 * 10^{-4}$	95023	96.86	96.88	97.11	97.13	97.31	97.31	0.02	100.00								
	Frown	5000	500	$6.20 * 10^{-4}$	165001	95.46	95.63	95.79	96.02	96.08	96.30	0.23	77.78	96.27	96.51	96.43	96.68	96.56	96.82	0.24	66.67
	Penn	5000	250	$4.46 * 10^{-4}$	130008	96.41	96.57	96.57	96.77	96.72	96.91	0.18	100.00	96.81	97.01	96.90	97.10	96.95	97.17	0.20	100.00
SVMTool	AnCora	2000	500	$7.85 * 10^{-4}$	105003	97.03	97.05	97.29	97.31	97.47	97.48	0.03	100.00								
	Frown	10000	500	$7.50 * 10^{-4}$	150014	95.77	95.98	96.10	96.38	96.37	96.67	0.29	100.00	96.54	96.88	96.70	97.05	96.79	97.19	0.33	100.00
	Penn	2000	1000	$2.95 * 10^{-4}$	199005	96.30	96.51	96.47	96.76	96.56	96.93	0.30	77.78	96.69	97.06	96.76	97.17	96.81	97.26	0.37	77.78
TnT	AnCora	2000	500	$5.90 * 10^{-4}$	115002	97.11	97.14	97.31	97.34	97.47	97.49	0.03	100.00								
	Frown	10000	500	$6.60 * 10^{-4}$	150014	95.67	95.90	95.97	96.26	96.25	96.52	0.29	87.50	96.42	96.71	96.56	96.86	96.63	96.99	0.30	87.50
	Penn	2000	125	$1.76 * 10^{-4}$	209396	96.13	96.16	96.27	96.30	96.39	96.40	0.03	100.00	96.45	96.47	96.52	96.53	96.57	96.57	0.02	100.00
TreeTagger	AnCora	5000	500	$1.17 * 10^{-3}$	100533	96.09	95.89	96.42	96.26	96.67	96.51	0.19	100.00								
	Frown	5000	1000	$2.20 * 10^{-3}$	84011	94.65	94.20	95.06	94.75	95.47	95.15	0.35	88.89	95.75	95.45	95.92	95.69	96.06	95.89	0.31	87.50
	Penn	2000	125	$6.00 * 10^{-4}$	152778	95.13	95.17	95.62	95.51	95.83	95.75	0.08	100.00	95.94	95.93	96.04	96.08	96.11	96.19	0.06	100.00

Table 1: Accuracy for learning trends whose associated learning curves do not intersect in the control window.

6. The algorithm at work

Given a corpus $\mathscr{C}$, we illustrate how far in advance and well a learning curve for accuracy can be approximated. As evaluation basis we introduce the *run*, a pair $\mathscr{E} = [\mathscr{T}(\mathscr{C}_\sigma^\mathscr{K}, \pi), \tau]$ characterized by one learning trace and a convergence threshold τ. We study the set of runs $\mathscr{F} = \{\mathscr{E}_i\}_{i \in I}$ in Table 1 where, with the aim of avoiding distortions due to particular choices, a common accuracy pattern π is considered. Following previous works on the performance of different models for fitting learning curves (Gu et al., 2001), particularly in the NLP domain (Kolachina et al., 2012), we opt for the power law family.

6.1. The monitoring

All our estimates consider the first learning trend for which the convergence threshold is reached on the learning scheme considered. The latter is characterized by its kernel and step, while the learning trend is indicated using its level. Since this level marks the end of the iterative process, we baptize it as CLevel, after convergence level. As kernels, we chose individuals with 2000, 5000 and 10000 words, deemed as significant sizes for training data (Brants, 2000; Hajič, 2000), while steps of 125, 250, 500 and 1000 words are selected.

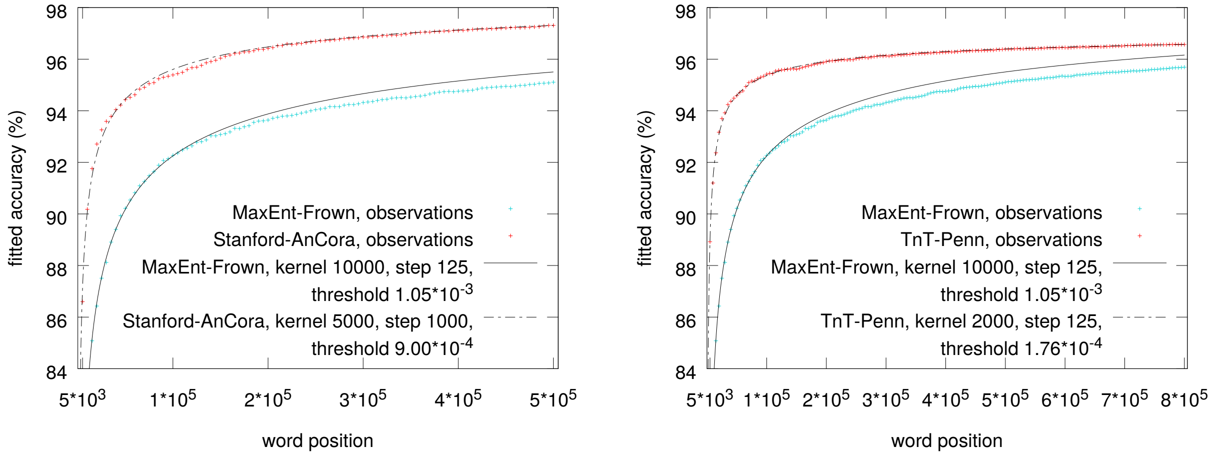

Figure 3: Learning trends for MaxEnt on Frown and Stanford on AnCora, and MaxEnt on Frown and TnT on Penn.

We monitor the results over a set $\mathscr{S}$ of word positions, hereinafter referred as *control levels*, taken from a common finite sub-interval for the prediction windows $\{(\text{CLevel}_i, \infty)\}_{i \in I}$ associated to the set $\mathscr{F} = \{\mathscr{E}_i\}_{i \in I}$ of runs studied. We baptize this sub-interval the *control window*. We consider two control windows: $[3{*}10^5, 5{*}10^5]$ and $[3{*}10^5, 8{*}10^5]$, whose upper control levels approximately coincide with the end of AnCora and the number of observations for the other two corpora, respectively. We then compute, on control levels spaced approximately at 5000 words, the accuracy (Ac) and its estimate (EAc) for each run in Table 1, although for reasons of space only six of them are shown.

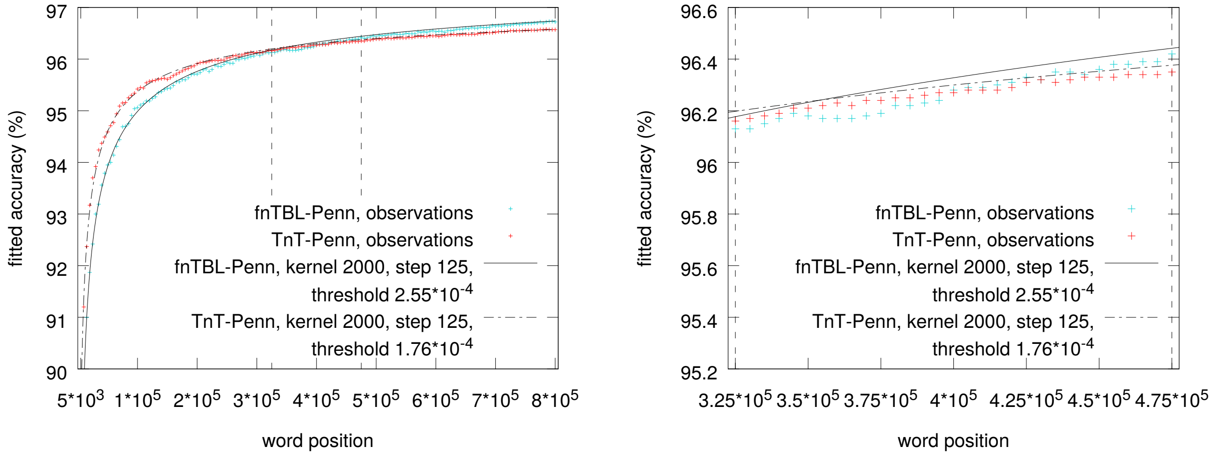

Figure 4: Learning trends for fnTBL and TnT on the Penn Treebank, with details in zoom.

As $\{\text{CLevel}_i\}_{i \in I}$ ranges from 84011 (for TreeTagger on Frown), to 209396 (for fnTBL and TnT on Penn), we can thus conclude that the size of the prediction window multiplies that of the text used to approximate the learning curve by a factor that varies from 1.39 to 4.95 (resp. from 2.82 to 8.52) when the last control level is $\|5 * 10^5\|$ (resp. $\|8 * 10^5\|$). So, the percentage of the corpus used to reach the convergence threshold in the worst and the best case is 41.88% and 16.8% (resp. 26.17% and 10.5%), respectively. These numbers allows us to get a first idea of the scope of our proposal.

6.2. *Quality metrics and experimental results*

For each run, we compute on each control level the *percentage error* (PE) as the difference between the estimated and the observed values, expressed as a percentage of the accuracy at that instant. We then calculate the *mean absolute percent error* (MAPE) as the arithmetic mean of the unsigned PE. Formally, given a run $\mathscr{E}$ and a set of control levels $\mathscr{S}$:

$$\text{PE}(\mathscr{E})(i) := 100 * \frac{\text{EAc}(\mathscr{E})(i) - \text{Ac}(\mathscr{T}, \mathscr{C})(i)}{\text{Ac}(\mathscr{T}, \mathscr{C})(i)}, \; \mathscr{E} = [\mathscr{T}(\mathscr{C}_\sigma^\mathscr{K}, \pi), \tau], \; i \in \mathscr{S}, \tag{10}$$

$$\text{MAPE}(\mathscr{E})(\mathscr{S}) := \frac{100}{|\mathscr{S}|} * \sum_{i \in \mathscr{S}} |\text{PE}(\mathscr{E})(i)| \tag{11}$$

On average, the quality of the estimates done over a control window is inversely proportional to the MAPE, which varies from 0.02 (resp. 0.02) with Stanford on AnCora (resp. TnT on Penn), to 0.41 (resp. 0.44) with MaxEnt on Frown in the case of $[3 * 10^5, 5 * 10^5]$ (resp. $[3 * 10^5, 8 * 10^5]$). To illustrate this, Fig. 3 enables the comparison of learning curves and their estimates for the best and the worst MAPE results on both control windows. On the other hand, 43.33% (resp. 30%) of these values are in the interval $[0, 0.09]$, rising to 66.67% (resp. 50%) in $[0, 0.20]$, and reaching 90% (resp. 75%) in the interval $[0, 0.30]$. However, while at first sight these data seem promising, they are not sufficient to measure the actual performance. We also need, in order to judge the reliability of our approximations, to determine to what extent the estimation errors impact decision-making on accuracy-based criteria. To this end, we are interested in calculating the percentage of runs for which such errors do not cause wrong decisions to be made in comparing accuracy values. So, fixing a set of control levels $\mathscr{S}$ and a set $\mathscr{F}$ of runs on a corpus $\mathscr{C}$, the robustness of one of such runs with respect to the rest depends on its estimates not altering the relative position of the learning curves for accuracy throughout $\mathscr{S}$.

		Kernel (words)	Step	Threshold (degrees)	CLevel	Control Level (word position)						MAPE	RER	Control Level (word position)						MAPE	RER
						$\|3 * 10^5\|$		$\|4 * 10^5\|$		$\|5 * 10^5\|$				$\|6 * 10^5\|$		$\|7 * 10^5\|$		$\|8 * 10^5\|$			
						Ac	EAc	Ac	EAc	Ac	EAc			Ac	EAc	Ac	EAc	Ac	EAc		
AnCora	MBT	10000	500	$8.21 * 10^{-4}$	107023	96.10	96.16	96.40	96.47	96.63	96.69	0.06	26.83								
	TreeTagger	5000	500	$1.17 * 10^{-3}$	100533	96.09	95.89	96.42	96.26	96.67	96.51	0.19									
	SVMTool	2000	500	$7.85 * 10^{-4}$	105003	97.03	97.05	97.29	97.31	97.47	97.48	0.03	100.00								
	TnT	2000	500	$5.90 * 10^{-4}$	115002	97.11	97.14	97.31	97.34	97.47	97.49	0.03									
Frown	Morfette	2000	500	$5.10 * 10^{-4}$	196006	95.65	95.78	95.97	96.19	96.23	96.48	0.22	100.00	96.39	96.70	96.54	96.88	96.69	97.02	0.28	87.13
	TnT	10000	500	$6.60 * 10^{-4}$	150014	95.67	95.90	95.97	96.26	96.25	96.52	0.29		96.42	96.71	96.56	96.86	96.63	96.99	0.30	
	MXPOST	10000	500	$7.90 * 10^{-4}$	169009	94.75	94.75	95.18	95.24	95.49	95.58	0.06		95.74	95.84	95.95	96.04	96.09	96.21	0.09	74.26
	TreeTagger	5000	1000	$2.20 * 10^{-3}$	84011	94.65	94.20	95.06	94.75	95.47	95.15	0.35		95.75	95.45	95.92	95.69	96.06	95.89	0.31	
Penn	fnTBL	2000	125	$2.55 * 10^{-4}$	209396	96.10	96.11	96.28	96.33	96.43	96.47	0.05		96.53	96.58	96.64	96.67	96.72	96.74	0.04	100.00
	MXPOST	10000	250	$3.68 * 10^{-4}$	170024	96.11	96.17	96.35	96.38	96.52	96.53	0.03		96.59	96.63	96.69	96.72	96.74	96.78	0.03	
	fnTBL	2000	125	$2.55 * 10^{-4}$	209396	96.10	96.11	96.28	96.33	96.43	96.47	0.05	85.37	96.53	96.58	96.64	96.67	96.72	96.74	0.04	94.06
	TnT	2000	125	$1.76 * 10^{-4}$	209396	96.13	96.16	96.27	96.30	96.39	96.40	0.03		96.45	96.47	96.52	96.53	96.57	96.57	0.02	
	MaxEnt	5000	500	$4.38 * 10^{-4}$	146501	95.95	96.05	96.17	96.27	96.34	96.42	0.11		96.45	96.54	96.55	96.63	96.63	96.70	0.09	75.25
	TnT	2000	125	$1.76 * 10^{-4}$	209396	96.13	96.16	96.27	96.30	96.39	96.40	0.03		96.45	96.47	96.52	96.53	96.57	96.57	0.02	
	MBT	2000	125	$6.23 * 10^{-4}$	149896	95.24	95.27	95.56	95.62	95.76	95.86	0.07	31.71	95.89	96.05	96.05	96.20	96.13	96.32	0.12	42.57
	TreeTagger	2000	125	$6.00 * 10^{-4}$	152778	95.13	95.17	95.62	95.51	95.83	95.75	0.08		95.94	95.93	96.04	96.08	96.11	96.19	0.06	
	Morfette	5000	1000	$3.17 * 10^{-4}$	167010	96.47	96.74	96.63	96.93	96.74	97.06	0.31	100.00	96.81	97.16	96.91	97.24	96.96	97.30	0.33	84.16
	Stanford	5000	250	$4.46 * 10^{-4}$	130008	96.41	96.57	96.57	96.77	96.72	96.91	0.18		96.81	97.01	96.90	97.10	96.95	97.17	0.20	
	MXPOST	10000	250	$3.68 * 10^{-4}$	170024	96.11	96.17	96.35	96.38	96.52	96.53	0.03	97.56	96.59	96.63	96.69	96.72	96.74	96.78	0.03	99.01
	TnT	2000	125	$1.76 * 10^{-4}$	209396	96.13	96.16	96.27	96.30	96.39	96.40	0.03		96.45	96.47	96.52	96.53	96.57	96.57	0.02	

Table 2: Relative positions for learning trends whose associated learning curves intersect in the control window.

From this perspective, a set of similar MAPE values for the runs studied can provide good results regardless of whether they are large or small. We thereby justify the need for the complementary evaluation view just described, independent of the MAPE concept, even though it may be sometimes unrealistic due to its high exigency level. This follows from the fact that the robustness condition applies on all the control levels, which is excessive when we compare runs whose learning curves intersect in the control window. More specifically, as we matched observation and control levels, the error in the estimates of the intersection point should be lower than the distance between its neighbouring control levels. Since we distribute these levels approximately each 5000 words, such a degree of performance is highly unlikely. This can be seen from Fig. 4, where the runs compared have similar MAPE values in both control windows $[3*10^5, 5*10^5]$ (0.05 and 0.03) and $[3*10^5, 8*10^5]$ (0.04 and 0.02), as shown in Table 2. Furthermore, these values are low and close to the minimum ones observed (0.02), which provides learning trends also close to the associated learning curves. In spite of everything, the robustness condition is not verified, although the first impression is that the estimate is good. Therefore, we are here more interested in assessing the rate of distortion introduced in the comparison of both runs, understood as the percentage of estimated values preserving the relative positions of the corresponding observed ones.

We then distinguish two testing scenarios from the starting set of runs $\mathscr{F}$. The first scenario, reflected in Table 1, refers to the compliance for the robustness condition when the corresponding learning curves are disjoint. The second, detailed in Table 2, analyzes the impact of prediction errors in the comparison of runs whose learning curves intersect. More formally, given a set of control levels $\mathscr{S}$ and a set $\mathscr{F}$ of runs on a corpus $\mathscr{C}$, our primary reference for the first (resp. the second) scenario is the *reliability estimation* (RE) *of a pair of runs $\mathscr{E}, \tilde{\mathscr{E}} \in \mathscr{F}$ with respect to $i \in \mathscr{S}$*, defined as follows:

$$\text{RE}(\mathscr{E}, \tilde{\mathscr{E}})(i) := \begin{cases} 1 \text{ if } [\text{Ac}(\mathscr{T}, \mathscr{C})(i) \ @ \ \text{Ac}(\tilde{\mathscr{T}}, \mathscr{C})(i)] \text{ and } [\text{EAc}(\mathscr{T}\mathscr{C})(i) \ @ \ \text{Ac}(\tilde{\mathscr{T}}, \mathscr{C})(i)] \\ 0 \text{ otherwise} \end{cases} \tag{12}$$

$$(\textit{resp. } \text{RE}(\mathscr{E}, \tilde{\mathscr{E}})(i) := \begin{cases} 1 \text{ if } [\text{Ac}(\mathscr{T}, \mathscr{C})(i) - \text{Ac}(\tilde{\mathscr{T}}, \mathscr{C})(i)] * [\text{EAc}(\mathscr{T}\mathscr{C})(i) - \text{EAc}(\tilde{\mathscr{T}}, \mathscr{C})(i)] \geq 0 \\ 0 \text{ otherwise} \end{cases}) \tag{13}$$

where $\mathscr{E} = [\mathscr{T}(\mathscr{C}_{\sigma}^{\mathscr{K}}, \pi), \tau]$, $\tilde{\mathscr{E}} = [\tilde{\mathscr{T}}(\mathscr{C}_{\tilde{\sigma}}^{\tilde{\mathscr{K}}}, \pi), \tilde{\tau}]$, $@ \in \{<, >\}$ and $\mathscr{E} \neq \tilde{\mathscr{E}}$. For the first scenario and fixed a control level i, this Boolean function determines when an estimation error made on $\mathscr{E}$ does not conflict with the observation on $\tilde{\mathscr{E}}$. In the case of the second scenario, RE verifies when the estimates for $\mathscr{E}$ and $\tilde{\mathscr{E}}$ on the control level i preserve the relative positions of the corresponding observations. In any case, we can naturally extends the notion of RE to the entire control window $\mathscr{S}$.

Definition 7. *Let, on a corpus $\mathscr{C}$, $\mathscr{S}$ be a control window and $\mathscr{E}$ and $\tilde{\mathscr{E}}$ two different runs. We define the* reliability estimation ratio (RER) for $\mathscr{E}$ and $\tilde{\mathscr{E}}$ with respect to $\mathscr{S}$ *as*

$$\text{RER}(\mathscr{E}, \tilde{\mathscr{E}})(\mathscr{S}) := 100 * \frac{\sum_{i \in \mathscr{S}} \text{RE}(\mathscr{E}, \tilde{\mathscr{E}})(i)}{|\mathscr{S}|} \tag{14}$$

Although the RER covers our requirements to measure the performance in the second scenario, this is not the case for the first one, where we need to calculate the number of runs in a set $\mathscr{F}$ with regard to which the estimates for a given run $\mathscr{E}$ are reliable at all control levels.

Definition 8. *Let, on a corpus $\mathscr{C}$, $\mathscr{E}$ be a run, $\mathscr{S}$ a control window and $\mathscr{F} = \{\mathscr{E}_i\}_{i \in I}$ a set of runs such that $\mathscr{E} \notin \mathscr{F}$. We define the* decision-making robustness of $\mathscr{E}$ with regard to $\mathscr{F}$ on $\mathscr{S}$ *as the value*

$$\text{DMR}(\mathscr{E}, \mathscr{F})(\mathscr{S}) := 100 * \frac{|\mathscr{E}_i \in \mathscr{F}, \ \text{RER}(\mathscr{E}, \mathscr{E}_i)(\mathscr{S}) = 100|}{|\mathscr{F}|} \tag{15}$$

We are now ready to measure the performance of our proposal in the terms previously defined for each testing scenario. So, we include in Table 1 the DMR values for the first one, i.e. the prediction of accuracy when no intersecting learning curves are present in the control window, immediately adjacent to the corresponding MAPE entry. They range from 77.78% (resp. 66.67%) to 100% (resp. 100%) in the control window $[3*10^5, 5*10^5]$ (resp. $[3*10^5, 8*10^5]$). Moreover, 86.67% (resp. 80%) of these values appear in the interval $[87.50, 100]$. This rate reached 100.00% (resp. 95%) in the interval $[77.78, 100]$, as shown in Fig. 5, where we can also appreciate some of the characteristics that define the stability of the estimates. Firstly, low MAPE values are in line with full DMR performance (100%). This implies that the proximity between learning trends and learning curves is associated with a reduced impact for accuracy-based decisions, as might have been expected. Second, even in the case of the runs with the highest MAPE values, most DMR values remain close to the maximum. As a consequence, a practical level of robustness for the estimates seems to be guaranteed in every case. Finally, when the length of the corpora allows us to overlay the two control windows used ($[3*10^5, 5*10^5]$ and $[3*10^5, 8*10^5]$), our measures show that both MAPE and DMR values hardly changed significantly, reaffirming the stability of calculations.

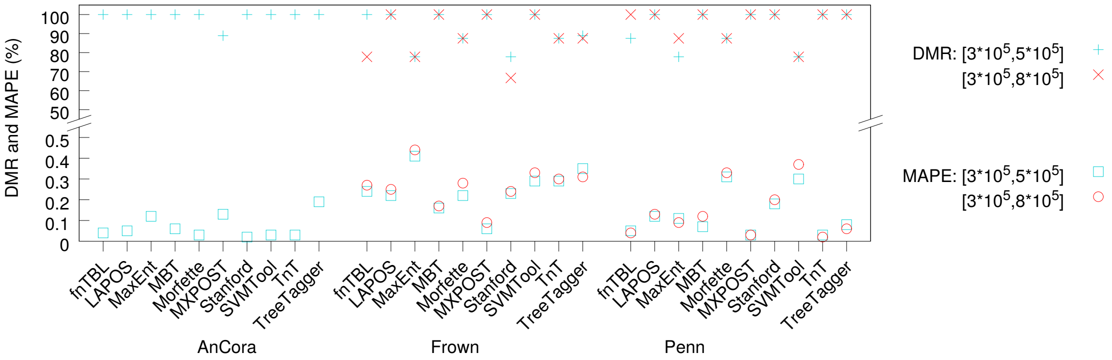

Figure 5: MAPE values for all runs, and DMR ones excluding intersections in the control windows.

The RER values for the second scenario, i.e. the comparison between accuracy estimates when the learning curves intersect in the control window, are reflected in Table 2 with regard to the pairs of associated runs. These range from 26.83% (resp. 42.57%) to 100% (resp. 100%) in the control window $[3*10^5, 5*10^5]$ (resp. $[3*10^5, 8*10^5]$). While 57.14% (resp. 50%) of these RERs are inside the interval $[87.13, 100]$, 71.43% (resp. 87.5%) of them can be found in $[74.26, 100]$, as can be seen in Fig. 6. As for the first scenario, these results support the stability of the estimates. So, when the MAPE values in the pair studied are comparable, the RER values are usually close to the maximum. In other words, the more similar the residuals, the higher the level of confidence in locating the relative position of the learning curves. This performance seems also to extend to most pairs with dissimilar MAPE, reflecting good behaviour of the technique under the worst of circumstances. Finally, the already mentioned stability of MAPE through the extension from the control window $[3*10^5, 5*10^5]$ to $[3*10^5, 8*10^5]$, applies to a lesser extent to RER values. Since intersection necessarily has to take place in $[3*10^5, 5*10^5]$ to provide meaning to this extension, the natural outcome is therefore the slight increase in RER. However, the extreme proximity between the learning curves, together with the existence of small irregularities in the observations can have the opposite effect. This occurs in two of the pairs considered (Morfette-TnT and Morfette-Stanford), where the real underlying factor in these cases is the impossibility of distinguishing the curves involved as from a given level. In any case, the reliability of the last estimate in the control window $[3*10^5, 5*10^5]$ (resp. $[3*10^5, 8*10^5]$) with regard to the relative position for the learning curves of each pair is 71.43% (resp. 100%)

7. Conclusions

Based on a functional analysis approach, we extend the classic discrete calculation of accuracy in POST systems to a continuous domain. We model the process as the uniform convergence of a sequence of learning trends which iteratively approximates the learning curve for accuracy. Since the limit learning trend computed is C-infinity, the strategy described also allows us to estimate in advance the learning curve for

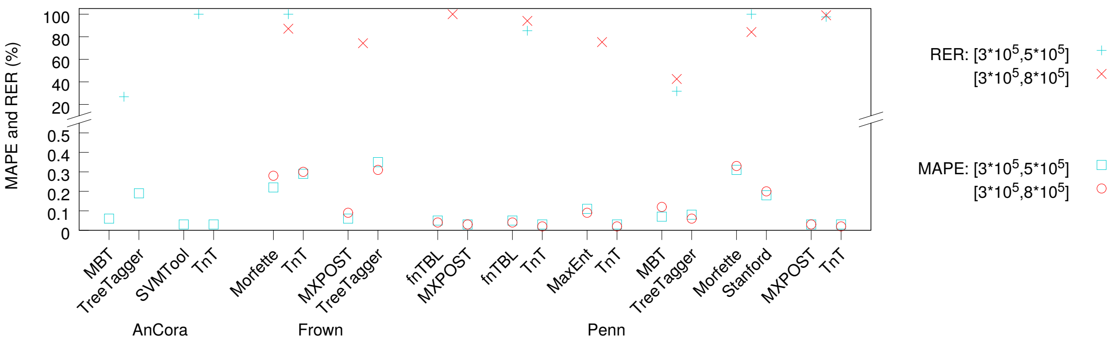

Figure 6: MAPE and RER values for all pairs of runs intersecting in the control windows.

accuracy over the entire corpus. The correctness of the algorithm has been formally proved, including a stopping criterion, with respect to our working hypotheses. The stability of these calculations is guaranteed once a point baptized as level of prediction has been surpassed.

Unlike previous approaches to the approximation of learning curves for a given particular linguistic task, we can exploit these properties to deal with a variety of practical uses beyond the mere estimation for the final value of accuracy. Focusing on the most representative ones, it is possible to estimate the extra accuracy increase between any two levels in the corpus, which is helpful to evaluate the training effort needed to attain a certain measurement performance. Comparing POST systems also becomes realistic at all training levels, providing us with a useful instrument for choosing the most appropriate tagger in each case. Finally, accuracy prediction below a certain degree of convergence fixed by the user can be guaranteed, which gives us the possibility of evaluating the adequacy of tagger configuration on the basis of a fraction of its generation process. Altogether, these facilities involve both quantitative and qualitative aspects, forming a powerful tool for reducing the training effort in tagger construction. In this context, the experimental results illustrate the goodness of the method, corroborating the initial expectations on a wide range of particular cases for a representative sample of POST systems and corpora.

Given that our only primary hypothesis is the use of an accuracy pattern as parametric family for modeling learning trends, we believe that it is possible to directly apply the same technique on other prediction scenarios, as long as they verify such a condition. This should be the case of a range of well known questions in ML, particularly in the domain of AL, in order to solve the stopping learning question. We can here mention various NLP tasks, such as machine translation, text classification, named entity recognition or any other kind of linguistic notation. All these are new fields of application we plan to explore in a future work.

Acknowledgments

We thank Prof. Christian Mair, of the University of Freiburg, who kindly provided us with a copy of the tagged versions of the Brown family of corpora, including the Freiburg-Brown corpus of American English. This research has been partially funded by the Ministry of Science and Innovation, and ERDF through project TIN2010-18552-C03-01, and by the Autonomous Government of Galicia through projects R2014/029 and R2014/034.

References

Apostol, T. M., 2000. Mathematical analysis. Narosa Book Distributors Pvt Ltd, New Delhi.

Argamon-Engelson, S., Dagan, I., 1999. Committee-based sample selection for probabilistic classifiers. Journal of Artificial Intelligence Research 11, 335–360.

Attenberg, J., Provost, F., 2011. Inactive learning?: Difficulties employing active learning in practice. ACM SIGKDD Explorations Newsletter 12 (2), 36–41.

Becker, M., Osborne, M., 2005. A two-stage method for active learning of statistical grammars. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence. Edinburgh, pp. 991–996.

Birch, A., Osborne, M., Koehn, P., 2008. Predicting success in machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. Honolulu, pp. 745–754.

Bloodgood, M., Vijay-Shanker, K., 2009. A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping. In: Proceedings of the 13th Conference on Computational Natural Language Learning. Boulder, pp. 39–47.

Branch, M. A., Coleman, T. F., Li, Y., 1999. A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems. SIAM Journal on Scientific Computing 21 (1), 1–23.

Brants, T., 2000. TnT: a statistical part-of-speech tagger. In: Proceedings of the 6th Conference on Applied Natural Language Processing. Seattle, pp. 224–231.

Brill, E., 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. Computational Linguistics 21 (4), 543–565.

Buchholz, S., Marsi, E., 2006. CoNLL-X shared task on multilingual dependency parsing. In: Proceedings of the 10th Conference on Computational Natural Language Learning. New York, pp. 149–164.

Chan, Y. S., Ng, H. T., 2007. Domain adaptation with active learning for word sense disambiguation. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. Prague, pp. 49–56.

Charniak, E., 2000. A maximum-entropy-inspired parser. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. Seattle, pp. 132–139.

Chen, J., Schein, A., Ungar, L., Palmer, M., 2006. An empirical study of the behavior of active learning for word sense disambiguation. In: Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics. New York, pp. 120–127.

Chrupala, G., Dinu, G., van Genabith, J., 2008. Learning morphology with Morfette. In: Proceedings of the 6th International Conference on Language Resources and Evaluation. Marrakech, pp. 2362–2367.

Clark, A., Fox, C., Lappin, S., 2010. The Handbook of Computational Linguistics and Natural Language Processing. John Wiley & Sons, Hoboken.

Cohn, D., Atlas, L., Ladner, R., 1994. Improving generalization with active learning. Machine Learning 15 (2), 201–221.

Collins, M., 2002. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (Vol 10). Philadelphia, pp. 1–8.

Culotta, A., McCallum, A., 2005. Reducing labeling effort for structured prediction tasks. In: Proceedings of the 20th National Conference on Artificial Intelligence (Vol 2). Pittsburgh, pp. 746–751.

Daelemans, W., Zavrel, J., Berck, P., Gillis, S., 1996. MBT: A memory–based part-of-speech tagger generator. In: Proceedings of the 4th Workshop on Very Large Corpora. Copenhagen, pp. 14–27.

Dagan, I., Engelson, S. P., 1995. Committee-based sampling for training probabilistic classifiers. In: Proceedings of the 12th International Conference on Machine Learning. Tahoe City, pp. 150–157.

DeRose, S. J., 1988. Grammatical category disambiguation by statistical optimization. Computational Linguistics 14 (1), 31–39.

Francis, W. N., Kučera, H., 1967,71,79. A Standard Corpus of Present-Day Edited American English, for use with Digital Computers. Brown University, Providence.

Giménez, J., Márquez, L., 2004. SVMTool: A general POS tagger generator based on support vector machines. In: Proceedings of the 4th International Conference on Language Resources and Evaluation. Lisbon, pp. 43–46.

Goldwater, S., Griffiths, T., 2007. A fully bayesian approach to unsupervised part-of-speech tagging. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. Prague, pp. 744–751.

Gu, B., Hu, F., Liu, H., 2001. Modelling classification performance for large data sets. In: Proceedings of the Second International Conference on Advances in Web-Age Information Management. Xi'an, pp. 317–328.

Haertel, R., Ringger, E., Seppi, K., Carroll, J., McClanahan, P., 2008. Assessing the costs of sampling methods in active learning for annotation. In: Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers. Columbus, pp. 65–68.

Hajič, J., 2000. Morphological tagging: data vs. dictionaries. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. Seattle, pp. 94–101.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., Zhang, Y., 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In: Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. Boulder, pp. 1–18.

Hinrichs, L., Smith, N., Waibel, B., 2010. Manual of information for the part-of-speech-tagged, post-edited 'Brown' corpora. ICAME Journal 34, 189–233.

Hulden, M., Francom, J., 2012. Boosting statistical tagger accuracy with simple rule-based grammars. In: Proceedings of the 8th International Conference on Language Resources and Evaluation. Istanbul, pp. 2114–2117.

Kolachina, P., Cancedda, N., Dymetman, M., Venkatapathy, S., 2012. Prediction of learning curves in machine translation. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers (Vol 1). Jeju Island, pp. 22–30.

Laws, F., Schätze, H., 2008. Stopping criteria for active learning of named entity recognition. In: Proceedings of the 22nd International Conference on Computational Linguistics (Vol 1). Manchester, pp. 465–472.

Leech, G., 2009. Change in Contemporary English: A Grammatical Study. Studies in English Language. Cambridge University Press, Cambridge.

Lewis, D. D., Gale, W. A., 1994. A sequential algorithm for training text classifiers. In: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Dublin, pp. 3–12.

Liere, R., Tadepalli, P., 1997. Active learning with committees for text categorization. In: Proceedings of the 14th National Conference on Artificial Intelligence. Providence, pp. 591–596.

Mair, C., 2006. Twentieth-Century English: History, Variation and Standardization. Studies in English Language. Cambridge University Press, Cambridge.

Mair, C., Leech, G., 2007. The Freiburg-Brown Corpus ('Frown') (POS-tagged version).

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., Taylor, A., 1999. Treebank-3 LDC99T42. Web download file. Linguistic Data Consortium, Philadelphia.

McCallum, A., Nigam, K., 1998. Employing EM and Pool-Based Active Learning for Text Classification. In: Proceedings of the 15th International Conference on Machine Learning. Madison, pp. 350–358.

Merialdo, B., Jun. 1994. Tagging english text with a probabilistic model. Computational Linguistics 20 (2), 155–171.

Monachini, M., Calzolari, N., 1996. EAGLES Synopsis and Comparison of Morphosyntactic Phenomena Encoded in Lexicons and Corpora. A Common Proposal and Applications to European Languages. EAGLES document EAG-CLWG-MORPHSYN/R, CNRS.

Mukund, S., Ghosh, D., Srihari, R. K., 2010. Using cross-lingual projections to generate semantic role labeled corpus for Urdu: A resource poor language. In: Proceedings of the 23rd International Conference on Computational Linguistics. Beijing, pp. 797–805.

Neubig, G., Nakata, Y., Mori, S., 2011. Pointwise prediction for robust, adaptable japanese morphological analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers (Vol 2). Portland, pp. 529–533.

Ngai, G., Florian, R., 2001. Transformation-Based Learning in the Fast Lane. In: Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies. Pittsburgh, pp. 1–8.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., Yuret, D., 2007. The CoNLL 2007 shared task on dependency parsing. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Prague, pp. 915–932.

Perlich, C., Provost, F., Simonoff, J. S., 2003. Tree induction vs. logistic regression: A learning-curve analysis. The Journal of Machine Learning Research 4, 211–255.

Petrov, S., Barrett, L., Thibaux, R., Klein, D., 2006. Learning accurate, compact, and interpretable tree annotation. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics. Sydney, pp. 433–440.

Popel, M., Mareček, D., Štěpánek, J., Zeman, D., Žabokrtský, Z., 2013. Coordination structures in dependency treebanks. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Vol 1). Sofia, pp. 517–527.

Ratnaparkhi, A., 1996. A maximum entropy model for part-of-speech tagging. In: Proceedings of the 1996 Conference on Empirical Methods in Natural Language Processing. Philadelphia, pp. 133–142.

Ravi, S., Knight, K., 2009. Minimized models for unsupervised part-of-speech tagging. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Conference on Natural Language Processing of the AFNLP (Vol 1). Suntec, pp. 504–512.

Recasens, M., Màrquez, L., Sapena, E., Martí, M. A., Taulé, M., Hoste, V., Poesio, M., Versley, Y., 2010. SemEval-2010 task 1: Coreference resolution in multiple languages. In: Proceedings of the 5th International Workshop on Semantic Evaluation. Uppsala, pp. 1–8.

Ringger, E., McClanahan, P., Haertel, R., Busby, G., Carmen, M., Carroll, J., Seppi, K., Lonsdale, D., 2007. Active learning for part-of-speech tagging: Accelerating corpus annotation. In: Proceedings of the Linguistic Annotation Workshop. Prague, pp. 101–108.

Roy, N., McCallum, A., 2001. Toward optimal active learning through sampling estimation of error reduction. In: Proceedings of the 18th International Conference on Machine Learning. Williamstown, pp. 441–448.

Schmid, H., 1994. Probabilistic part-of-speech tagging using decision trees. In: Proceedings of the International Conference on New Methods in Language Processing. Manchester, pp. 44–49.

Seung, H. S., Opper, M., Sompolinsky, H., 1992. Query by committee. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory. Pittsburgh, pp. 287–294.

Shen, D., Zhang, J., Su, J., Zhou, G., Tan, C.-L., 2004. Multi-criteria-based active learning for named entity recognition. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics. Barcelona, pp. 589–596.

Søgaard, A., 2010. Simple semi-supervised training of part-of-speech taggers. In: Proceedings of the ACL 2010 Conference Short Papers. Uppsala, pp. 205–208.

Spoustová, D., Hajič, J., Raab, J., Spousta, M., 2009. Semi-supervised training for the averaged perceptron POS tagger. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Athens, pp. 763–771.

Surdeanu, M., Johansson, R., Meyers, A., Màrquez, L., Nivre, J., 2008. The CoNLL 2008 shared task on joint parsing of syntactic and semantic dependencies. In: Proceedings of the 12th Conference on Computational Natural Language Learning. Manchester, pp. 159–177.

Tang, M., Luo, X., Roukos, S., 2002. Active learning for statistical natural language parsing. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Philadelphia, pp. 120–127.

Taulé, M., Martí, M. A., Recasens, M., 2008. AnCora: Multilevel annotated corpora for Catalan and Spanish. In: Proceedings of the 6th International Conference on Language Resources and Evaluation. Marrakech, pp. 96–101.

Thompson, C. A., Califf, M. E., Mooney, R. J., 1999. Active learning for natural language parsing and information extraction.

In: Proceedings of the 16th International Conference on Machine Learning. Bled, pp. 406–414.

Tomanek, K., Wermter, J., Hahn, U., 2007. An approach to text corpus construction which cuts annotation costs and maintains reusability of annotated data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Prague, pp. 486–495.

Tong, S., Koller, D., 2002. Support vector machine active learning with applications to text classification. The Journal of Machine Learning Research 2, 45–66.

Toutanova, K., Klein, D., Manning, C. D., Singer, Y., 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (Vol 1). Edmonton, pp. 173–180.

Tsuruoka, Y., Miyao, Y., Kazama, J., 2011. Learning with Lookahead: Can History-Based Models Rival Globally Optimized Models? In: Proceedings of the 15th Conference on Computational Natural Language Learning. Portland, pp. 238–246.

van Halteren, H., 1999. Performance of taggers. In: van Halteren, H. (Ed.), Syntactic Wordclass Tagging. Kluwer Academic Pub., Hingham, pp. 81–94.

Vlachos, A., 2008. A stopping criterion for active learning. Computer Speech and Language 22 (3), 295–312.

Yuret, D., Han, A., Turgut, Z., 2010. SemEval-2010 task 12: Parser evaluation using textual entailments. In: Proceedings of the 5th International Workshop on Semantic Evaluation. Los Angeles, pp. 51–56.

Zhu, J., 2007. Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Prague, pp. 783–790.

Zhu, J., Ma, M., 2012. Uncertainty-based active learning with instability estimation for text classification. ACM Transactions on Speech and Language Processing 8 (4), 5:1–5:21.