Part–of–speech Tagging: A Machine Learning Approach based on Decision Trees

Memòria presentada al Departament de Llenguatges i Sistemes Informàtics de la Universitat Politècnica de Catalunya per a optar al grau de Doctor en Informàtica

Lluís Màrquez i Villodre

sota la direcció del doctor
Horacio Rodríguez Hontoria

Barcelona, Maig de 1999

CHAPTER 7

Related Work: A Comparative Analysis

In this chapter we will present both a qualitative and quantitative analysis of our work, and a comparison with a number of the most representative related approaches.

The chapter is divided into two parts. The first is introductory and includes some discussion about the difficulties in comparing taggers from a quantitative perspective. The second is devoted to the comparative study proper. In order to do this, some important works were selected, from those previously presented in the state–of–the–art chapter, taking into account several features: currency, good results, good capabilities, use of decision trees, classical paradigms, similarity to our approach, availability, etc. The comparisons are done from several perspectives, e.g., accuracy efficiency, flexibility, expressiveness, etc., pointing out the most important pros and cons.

1. About the Evaluation and Comparison of Results

In the NLP literature, comparisons of alternative taggers are very rarely performed under identical experimental conditions. Despite this fact, in most current papers it is argued that the performance of a particular tagger is better than others, as a result of some kind of indirect comparison between them. We think that in many cases the conclusions presented are stated on the basis of insufficiently strong evidence and, therefore, are not reliable enough. As a conclusion, the significance of such experiments has to be carefully interpreted from a conservative perspective. In particular, all the results from indirect comparisons of taggers reported in the previous chapters and in the following section, should not be understood as strong statements, but only as advisory figures.

Unfortunately, there have been very few direct comparisons of alternative taggers[1] on identical test data. Even if so is done, a distorting factor, which has proven to be very significant, is the presence of noise (mistagged words) in the training and test corpora [**PM98**]. The core of this section is devoted to a particular study on this topic, which illustrates, with a real example, that a conventional statistical test for the difference of proportions is insufficient in the presence of a certain amount of noise.

It is important to note that another source of uncertainty on the significance of the statistical tests for comparing supervised learning algorithms, is the proper experimental methodology. Several authors indicated the risks of such methods in detecting a difference between alternative algorithms when no difference exists, and, conversely, to decide that two alternatives perform equally, when these differences do exist [**EMS97, Kay97, Die98b**]. All these problems come from the violation

[1] Some of the exceptions are the works [**SV97, VS98**], in which a very strict comparison between taggers is performed.

of the underlying independence assumptions of such tests (e.g. training and test sets obtained by resampling are not independent). Whenever it was possible, we used cross–validation in order to avoid dependence in the testing sets[2]. In this way, the probability of error is substantially reduced.

1.1. Identifying Problems. We think that there are a number of not enough controlled factors when an experimental evaluation or comparison of taggers is designed. To illustrate this point, we report below a list of some of the factors which are common practice.

Regarding the evaluation process:

- Training and test experiments are usually performed over noisy corpora which distorts the results obtained.
- Performance figures are too often calculated from only a single or very small number of trials. Average results from multiple trials are crucial to obtain reliable estimations of accuracy[3].
- Testing experiments are usually done on corpora of the same characteristics of the training data —usually a small fresh portion of the training corpus— However, no serious attempts have been done in order to determine the reliability of the results when moving from one domain to another[4].
- No figures about computational effort —space/time complexity— are usually reported, neither from a theoretical nor an empirical perspective.

Regarding the comparison process:

- Comparisons between taggers are often indirect. They should be compared under the same conditions, including: identical training and test sets, the same tagset, etc., in a multiple–trial experiment with statistical tests of significance.

In the next section we will concentrate on one of these problems, namely the noise present in the test corpus, and we will study to what extent this situation affects the estimation of the real performance, and the result of a comparison between different taggers.

1.2. The Effect of Noise in Testing Corpora. From a machine–learning perspective, the relevant noise in the corpus is that of non systematically mistagged words (i.e. different annotations for words appearing in the same syntactic/semantic contexts). Commonly used annotated corpora have noise. See, for instance, the following examples collected from the Wall Street Journal corpus.

Verb participle forms are sometimes tagged as such (VBN) and also as adjectives (JJ) in other sentences with no structural differences:

1a) `...failing_VBG to_TO voluntarily_RB submit_VB the_DT` *`requested_VBN`* `information_NN...`

1b) `...a_DT large_JJ sample_NN of_IN` *`married_JJ`* `women_NNS with_IN at_IN least_JJS one_CD child_NN...`

---

[2]Dietterich [**Die98b**] proposes a particular variant of the classical n–fold cross–validation, namely 5 x 2cv, which is based on 5 iterations of 2–fold cross validation and which has proven to be very effective to prevent both *false negative* and *false positive* predictions.

[3]See [**Moo96**] for a severe criticism coming from the machine–learning community.

[4]See [**Kro97**] reporting experiments using well established taggers in the information extraction field.

Another structure not coherently tagged are noun chains when the nouns (NN) are ambiguous and can be also adjectives (JJ). In this case we find a number of different, and apparently arbitrary, combinations:

2a) ...Mr._NNP Hahn_NNP ,_, the_DT 62-year-old_JJ chairman_NN and_CC *chief_NN executive_JJ officer_NN* of_IN Georgia-Pacific_NNP Corp._NNP...

2b) ...Burger_NNP King_NNP 's_POS *chief_JJ executive_NN officer_NN* ,_, Barry_NNP Gibbons_NNP ,_, stars_VBZ in_IN ads_NNS saying_VBG...

2c) ...and_CC Barrett_NNP B._NNP Weekes_NNP ,_, chairman_NN ,_, president_NN and_CC *chief_JJ executive_JJ officer_NN* ._.

2d) ...the_DT company_NN includes_VBZ Neil_NNP Davenport_NNP ,_, 47_CD ,_, president_NN and_CC *chief_NN executive_NN officer_NN* ;_:

The noise in the test set produces a wrong estimation of accuracy, since correct answers are computed as wrong and vice-versa. In following sections we will show how this uncertainty in the evaluation (varying up to 5% under extreme situations) may be, in some cases, larger than the improvements reported from one system to another, so invalidating the conclusions of the comparison.

1.2.1. *Model Setting.* To study the appropriateness of the choices made by a POS tagger, a reference tagging must be selected and assumed to be correct in order to compare it with the tagger output. This is usually done by assuming that the disambiguated test corpora being used contains the correct POS disambiguation. This approach is quite right when the tagger error rate is larger enough than the test corpus error rate, nevertheless, the current POS taggers have reached a performance level that invalidates this choice, since the tagger error rate is getting too close to the error rate of the test corpus.

Since we want to study the relationship between the tagger error rate and the test corpus error rate, we have to establish an absolute reference point. Although Church [**Chu92**] questioned the concept of *correct analysis*, Samuelsson and Voutilainen [**SV97**] established with a 95% confidence level that two human annotators would disagree only in 0.1% of the cases in the tagging of a 55,000 word corpus, when both the tagset and the annotation criteria are well–defined. This leads us to assume that there exists an *absolute* correct disambiguation, respect to which the error rates of either the tagger or the test corpus can be computed. What we will focus on is how distorted is the tagger error rate by the use of a noisy test corpus as a reference.

The cases we can find when evaluating the performance of a certain tagger are presented in table 1. OK/¬OK stand for a right/wrong tag (respect to the absolute correct disambiguation). When both the tagger and the test corpus have the correct tag, the tag is correctly evaluated as *right.* When the test corpus has the correct tag and the tagger gets it wrong, the occurrence is correctly evaluated as *wrong.* But problems appear when the test corpus has a wrong tag: If the tagger gets it correctly, it is evaluated as *wrong* when it should be *right* (false negative). If the tagger gets it wrong, it will be rightly evaluated as *wrong* if the error committed by the tagger is other than the error in the test corpus, but wrongly evaluated as *right* (false positive) if the error is the same.

Table 1 shows the computation of the percentages of each case. The meaning of the variables appearing in the table and used in the following sections is reported below:

C: Test corpus error rate. Usually an estimation is supplied with the corpus.

t: Tagger performance rate on words rightly tagged in the test corpus. It can be seen as $P(\text{OK}_t|\text{OK}_c)$.

u: Tagger performance rate on words wrongly tagged in the test corpus. It can be seen as $P(\text{OK}_t|\neg\text{OK}_c)$.

p: Probability that the tagger makes the same error than the test corpus, given that both get a wrong tag.

x: *Real* performance of the tagger, i.e. what would be obtained on an error–free test set.

K: Observed performance of the tagger, computed on the noisy test corpus.

Corpus	Tagger	Eval: *right*	Eval: *wrong*
OK_c	OK_t	$(1-C)\,t$	—
OK_c	$\neg\text{OK}_t$	—	$(1-C)\,(1-t)$
$\neg\text{OK}_c$	OK_t	—	$C\,u$
$\neg\text{OK}_c$	$\neg\text{OK}_t$	$C\,(1-u)\,p$	$C\,(1-u)\,(1-p)$

TABLE 1. Possible cases when evaluating a tagger

For simplicity, we consider only performance on ambiguous words. Considering unambiguous words will make the analysis more complex, since it should be taken into account that neither the behaviour of the tagger (given by u, t, p) nor the errors in the test corpus (given by C) are the same on ambiguous and unambiguous words. Nevertheless this is an issue that must be further addressed.

If we knew each one of the above proportions, we would be able to compute the *real* performance of our tagger. It would be as easy as adding up the rows from table 1 with a OK_t label —that is, the parts in which the tagger got the right disambiguation independently from the tagging of the test set— This gives us the following equation:

$$x = (1-C)\,t + C\,u\ . \tag{3}$$

The equation of the observed performance can also be extracted from table 1, adding up what is evaluated as *right*, obtaining:

$$K = (1-C)\,t + C\,(1-u)\,p\ . \tag{4}$$

It is trivial to extract from equations 3 and 4 the relationship between the real and the observed performance values:

$$x = K - C\,(1-u)\,p + C\,u\ .$$

Since only K and C are known (or approximately estimated) we can not compute the real performance of the tagger. All we can do is to establish some reasonable bounds for t, u and p, and see in which range is x.

Since all variables are probabilities, they are bounded in $[0, 1]$. We also can assume[5] that $K > C$. We can use this constraints and the above equations to bound the values of all variables. From 4, we obtain:

[5]In the cases we are interested in, the tagger observed performance, K, is over 90%, while the corpus error rate, C, is below 10%.

$$u = 1 - \frac{K - t(1-C)}{Cp}, \quad p = \frac{K - t(1-C)}{C(1-u)}, \quad \text{and} \quad t = \frac{K - C(1-u)p}{1-C}.$$

Thus, u will be maximum when p and t are maximum (i.e. 1). This gives an upper bound for u of $(1-K)/C$. When $t=0$, u will range in $[-\infty, 1-K/C]$ depending on the value of p. Since we are assuming $K>C$, the most informative lower bound for u keeps being zero. Similarly, p is minimum when $t=1$ and $u=0$. When $t=0$ the value for p will range in $[K/C, +\infty]$ depending on u. Since $K>C$, the most informative upper bound for p is still 1. Finally, t will be maximum when $u=1$ and $p=0$, and minimum when $u=0$ and $p=1$. Summarizing:

$$0 \leq u \leq min\left\{1, \frac{1-K}{C}\right\} \tag{5}$$

$$max\left\{0, \frac{K+C-1}{C}\right\} \leq p \leq 1 \tag{6}$$

$$\frac{K-C}{1-C} \leq t \leq min\left\{1, \frac{K}{1-C}\right\} \tag{7}$$

Since the values of the variables are mutually constrained, it is not possible that, for instance, u and t have simultaneously they upper bound value (if $(1-K)/C<1$ then $K/(1-C)>1$ and vice versa). Any bound which is out of $[0, 1]$ is not informative and the appropriate boundary, 0 or 1, is then used. Note that the lower bound for t will never be negative under the assumption $K>C$.

Once we have established these bounds, we can use equation 3 to compute the range for the real performance value of our tagger: x will be minimum when u and t are minimum (which requires $p=1$). This produces a lower bound $x_{min}=K-C$. Similarly, The upper bound for the real performance is:

$$x_{max} = \begin{cases} K+C & \text{if } K \leq 1-C \\ 2-K-C & \text{if } K \geq 1-C \end{cases}$$

As it will be used below, we can also state that if we allow p to vary, the bounds obtained are:

$$x_{min} = k - Cp \tag{8}$$

$$x_{max} = \begin{cases} K+C & \text{if } K \leq 1-C \\ 1-\frac{K+C-1}{p} & \text{if } K \geq 1-C \end{cases} \tag{9}$$

As an example, let's suppose we evaluate a tagger on a test corpus which is known to contain about 3% of errors ($C=0.03$), and obtain a reported performance of 93%[6] ($K = 0.93$). In this case, equations 8 and 9 yield a range for the real performance x that varies from $[0.93, 0.96]$ when $p=0$ to $[0.90, 0.96]$ when $p=1$.

These results suggest that although we observe a performance of K, we can not be sure of how well is our tagger performing without taking into account the values of t, u and p.

It is also obvious that the intervals in the above example are too wide, since they consider all the possible parameter values, even when they are very unlikely to

[6]This is the case of the RELAX tagger presented in section 4.1 of chapter 4. Note that 93% is the accuracy on ambiguous words (the equivalent overall accuracy was over 97%).

happen[7]. This enables us to try to narrow those intervals, limiting the possibilities to *reasonable* cases. This issue will be discussed in the following section.

1.2.2. *Reasonable Bounds for the Basic Parameters.* In real cases, not all parameter combinations will be equally likely. In addition, the bounds for the values of t, u and p are closely related to the tagging similarities between the training and test corpora. That is, if the training and test sets are extracted from the same corpus, they will probably contain the same kind of errors in the same kind of situations. This may cause the training procedure to *learn* the errors —especially if they are systematic— and thus the resulting tagger will tend to make the same errors that appear in the test set. On the contrary, if the training and test sets come from different sources —sharing only the tagset— the behaviour of the resulting tagger will not depend on the right or wrong tagging of the test set.

We can try to establish narrower bounds for the parameters than those obtained in section 1.2.1.

First of all, the value of t is already constrained enough, due to its high contribution $(1-C)$ to the value of K, which forces t to take a value close to K. For instance, applying the boundaries in equation 7 to the case $C=0.03$ and $K=0.93$, we obtain that t belongs to $[0.928, 0.959]$. For a more extreme (and unlikely) case such as $C=0.10$ and $K=0.90$, t would be in $[0.889, 1]$.

The range for u can be slightly narrowed considering the following: In the case of independent test and training corpora, u will tend to be equal to t. Otherwise, the more biased towards the corpus errors is the language model, the lower u will be. Note than allowing $u>t$ would mean that the tagger disambiguates *better* the noisy cases than the correct ones. Concerning to the lower bound, only in the case that all the errors in the training and test corpus were systematic (and thus can be learned) could u reach zero. However, not only this is not a likely situation, but also requires a perfect-learning tagger. It seems more reasonable that, in normal cases, errors will be random, and the tagger will behave randomly on the noisy occurrences. This yields a lower bound for u of $1/a$, being a the average ambiguity ratio for ambiguous words in the corpus.

The *reasonable* bounds for u are thus:

$$\frac{1}{a} \leq u \leq min\left\{t, \frac{1-K}{C}\right\} .$$

Finally, the value of p has similar constraints to those of u. If the test and training corpora are independent, the probability of making the same error, given that both are wrong, will be the random $1/(a-1)$. If the corpora are not independent, the errors that can be learned by the tagger will cause p to rise up to (potentially) 1. Again, only in the case that all errors where systematic, could p reach 1.

Then, the *reasonable* bounds for p are:

$$max\left\{\frac{1}{a-1}, \frac{K+C-1}{C}\right\} \leq p \leq 1 .$$

[7]For instance, it is not reasonable that $u = 0$, which would mean that the tagger never disambiguates correctly a wrong word in the corpus, or $p=1$, which would mean that it always makes the same error when both are wrong.

1.2.3. *On Comparing Tagger Performances.* As stated above, knowing which are the *reasonable* limits for the u, p and t parameters enables us to compute the range in which the real performance of the tagger could vary.

So, given two different taggers T_1 and T_2, and provided we know the values for the test corpus error rate and the observed performance of both cases (C_1, C_2, K_1, K_2), we can compare them by matching the *reasonable* intervals for the respective real performances x_1 and x_2.

From a conservative position, we cannot strongly state than one of the taggers performs better than the other when the two intervals overlap, since this implies a chance that the real performances of both taggers are the same.

The following real example has been extracted again from section 4.1 of chapter 4. The tagger T_1 uses only bigram information and has an observed performance on ambiguous words $K_1 = 0.9135$ (96.86% overall). The tagger T_2 uses trigrams and automatically acquired context constraints and has an accuracy of $K_2 = 0.9282$ (97.39% overall). Both taggers have been evaluated on a corpus (WSJ) with an estimated error rate[8] $C_1 = C_2 = 0.03$. The average ambiguity ratio of the ambiguous words in the corpus is $a = 2.5$ tags/word.

These data yield the following range of *reasonable* intervals for the real performance of the taggers.

for $p_i = 1/a = 0.4$	for $p_i = 1$
$x_1 \in [91.35, 94.05]$	$x_1 \in [90.75, 93.99]$
$x_2 \in [92.82, 95.60]$	$x_2 \in [92.22, 95.55]$

The same information is included in figure 1 which presents the reasonable accuracy intervals for both taggers, for p ranging from $1/a = 0.4$ to 1 (the shadowed part corresponds to the overlapping region between intervals).

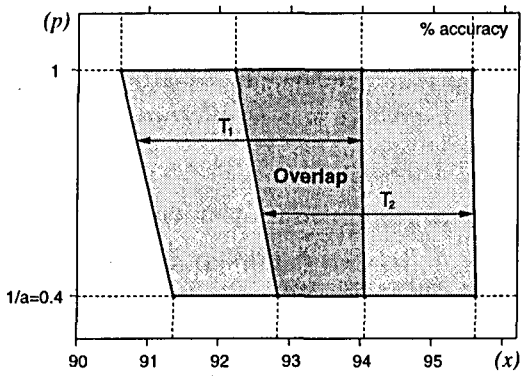

FIGURE 1. Reasonable intervals for both taggers

The intervals obtained have a large overlap region which implies that there are *reasonable* parameter combinations that could cause the taggers to produce different observed performances though their real accuracies were very similar. From this

[8]The (WSJ) corpus error rate is estimated over all words. We are assuming that the errors distribute uniformly among all words, although probably ambiguous words have a higher rate. Nevertheless, a higher value for C would cause the intervals to be wider and to overlap even more.

conservative approach, we would not be able to conclude that the tagger T_2 is better than T_1, even though the 95% confidence intervals for the observed performances did allow us to do so.

1.2.4. *Discussion.* The analysis presented of the effects of noise in the test corpus on the evaluation of POS taggers leads us to conclude that when a tagger is evaluated as better than another using noisy test corpus, there are *reasonable* chances that they are in fact very similar but one of them is just adapting better than the other to the noise in the corpus.

We believe that the widespread practice of evaluating taggers against a noisy test corpus has reached its limit, since the performance of current taggers is getting too close to the error rate usually found in test corpora.

An obvious solution —and maybe not as costly as one might think, since small test sets properly used may yield enough statistical evidence— is using only error-free test corpora. Another possibility is to study further the influence of noise in order to establish a criterion —which could be statistical, depending on the amount of overlapping between intervals— to decide whether a given tagger can be considered better than another.

Some of the issues that should be further considered are: The effect of noise on unambiguous words; the reasonable intervals for *overall* real performance; the (probably) different values of C, p, u and t for ambiguous/unambiguous words; how to estimate the parameter values of the tagger being evaluated in order to constrain as much as possible the intervals; the statistical significance of the interval overlappings; a more informed (and less conservative) criterion to reject/accept the hypothesis that both taggers are different, etc.

There is still much to be done in this direction. This work does not intend to establish a new evaluation method for POS tagging, but it aims to point out that there are some issues —such as the noise in test corpus— which have been paid little attention and which are more important than what they seem to be. In the same direction, we think that it would be important to start a discussion on POS tagger evaluation, with the objective of establishing a more rigorous test experimentation setting, which is needed to extract reliable conclusions.

2. A Comparison with Other Approaches

2.1. From a Qualitative Perspective. Recall that the main characteristic of our approach to tagging is the separation between language model acquisition and the disambiguation algorithm. The acquisition of the language model is performed by decision-tree induction and it is oriented to obtain a model which is both general and as accurate as possible, and which may be used by different taggers.

Decision trees have proven to be an appropriate formalism for acquiring and representing probabilistic information about the relevant features of POS ambiguities, and to meet the following requirements: independence, manageable size, flexibility to model and extend rich contextual information, efficiency, etc. The resulting trees can be also interpreted as rules or constraints from a linguistic perspective. However, this intuitive interpretation of the model is dubious in some cases, and falls far short of the manually derived linguistic models of the constraint grammar framework, and falls short even of some automatic rule-induction systems, such as Brill's TBL tagger.

The possibility of choosing the type of tagger in which to incorporate the tree–based models —from between different disambiguating algorithms— allows users to take advantage of any of these algorithms, given their own needs, available information, etc. In particular, we have applied them to three different taggers, namely, a reductionistic tagger (RTT), a statistically–based tagger (STT), and a constraint–based tagger (RELAX). These taggers have different properties in terms of accuracy, efficiency, flexibility, adaptability, etc., and none of them is preferable to the rest in absolute terms: the choice depends on the particular use.

In the folowing subsections, we will survey the main approaches to POS tagging and we will compare them to our proposal, by mainly discussing features in three directions: (1) Type of information considered; (2) How is this information acquired? and (3) How is this information used for disambiguating?

Given our previous description of POS tagging, the two first points refer to the language model, while the third refers to the tagging algorithm proper.

2.1.1. *Linguistic Taggers.* The Constraint Grammar framework (CG), developed at the Helsinki University by Karlsson and colleagues[**KVHA95**], is the most important representative of the linguistic–based family. It supplies a very rich and expressive constraint based language, in which linguistic knowledge can be properly expressed. Additionally, it reports the best accuracy figures for state–of–the–art tagging, which are far better than the best accuracies reported by taggers that uses automatically acquired knowledge.

The main drawback of such an approach is the very high acquisition cost (e.g., the linguistic models range from a few hundreds to several thousand rules, and they usually require years of labour), which makes it practically untransportable. The aim of automatic acquisition algorithms for tagging is, specifically, to try to surpass this limitation.

In our case, the acquisition cost is very low, since the construction of the tagger, given a relatively small annotated training set, is fully automatic. Additionally, the decision–tree–based model could be extended to incorporate more complex features, similar to those appearing in CG rules. However, the addition of such information would probably require some manual effort in designing the proper patterns and in tuning its influence.

The reductionistic tagger RTT described in chapter 4 presents a disambiguation algorithm that works in a similar way to that of CGs. Such a tagger can be properly tuned to adjust the output ambiguity, for instance to avoid deciding in the most difficult cases. In this way, the precision decreases slightly in favour of a higher recall.

Conversely, decision tree models can be transformed into a set of rules and can be used in combination with more elaborated linguistic knowledge in a flexible rule–based environment. This is the case with RELAX [**Pad98, MP97**], a relaxation labelling based tagger which is able to combine information obtained from several sources within a common constraint language capable of encompassing the most common CG rules. The combined approach with decision trees is also explained in chapter 4.

The results obtained in our approach are lower than those of CG (e.g. a recall about 2 points lower, for similar values of precision). It is a fact that years of work on machine learning for tagging has resulted in taggers that still significantly underperform manually constructed systems. It is our belief that automatic taggers

will never be able to compete with manual linguistic taggers. Apart from the fact that the type of information used in linguistic models is usually richer, we think that the key is in the treatment of infrequent events. It is well–known that, after growing a relatively small body of important rules that report a tagging accuracy of the same order as automatic taggers, much of the linguistic effort in the developing of a rich and comprehensive constraint grammar is devoted to covering exceptions and infrequent linguistic phenomena. The infrequent events have, in isolation, a very low statistical significance, and so they are very difficult to acquire with automatic machine learning algorithms (which are statistical in nature), especially in the presence of a certain amount of noise.

2.1.2. *Statistical and HMM–based Taggers.* Classic statistical or HMM–based taggers, such as the works in [**DeR88, Chu88, DeM90, CKPS92, Mer94**], work by maximizing the probability of a tag sequence, given a concrete sequence of input words. The probability is calculated under some simplifying assumptions, in particular, considering that the probability of the tag t for a certain word w in a concrete context depends only on the proper word and on a finite amount of preceding tags (n–gram probabilities). In our approach, decision trees can be used as a way of estimating the parameters of a similar probabilistic model, which will be used in a statistical tagger (this is the case of the STT tagger presented in chapter 4). From this perspective, our approach is equivalent to the statistical approach. The advantage is that the contextual model can easily be enriched in the DT approach, obtaining taggers that surpass the limited pure bigram and trigram models. In addition, this extension of the model can be performed avoiding the exponential growing of the number of parameters, since the proper tree induction algorithm automatically reduces the model to a tractable subset of relevant contexts, mixing different levels of generality (a fact that can be seen as a kind of smoothing process, [**ZD97**]).

One possible advantage of statistical taggers is that they are able to properly train the probabilistic model from an unannotated corpus plus a lexicon, providing the ambiguity class for each word (semi–supervised learning), by using the iterative Forward–Backward algorithm. In our case the learning is fully supervised, however, we have proven that a relatively small annotated training corpus is enough to achieve very good tagging accuracy, which can be further improved by applying a kind of retraining technique based on bootstrapping and tagger combination.

From the new family of statistical taggers extending the n–gram approach, we should comment on the maximum entropy (ME) approach, and, in particular, the tagger by Ratnaparkhi [**Rat96**], which is one of the most relevant works.

The tagger based on the ME approach has several features in common with our DT approach, e.g., it uses a rich feature representation and generates a tag probability distribution for each word. However, the ME model has the advantage of being able to combine diverse and non–local information sources without making the common independence assumptions that are necessary in the statistical approach. Additionally, Ratnaparkhi argues that the ME approach is superior to decision–tree approaches (as Magerman's system [**Mag95a**]) because it does not need to consider word classes to help prevent data fragmentation, and a trivial *count cut off* suffices as a smoothing procedure in order to achieve roughly the same level of accuracy.

We agree with the simplicity of the ME approach, which is able to deal with a huge amount of parameters with very little data preprocessing and smoothing.

However, in our basic approach, also very simple smoothing techniques (discount and redistribution, and pruning of DTs) were used to deal with sparseness, and the reported accuracy on the WSJ corpus is, at least, as good as his. In contrast, the decision–tree approach compares favourably with ME in the acquisition and tagging cost. In ME, both depend linearly on the input word sequence length, however they include a significant multiplicative constant, i.e., the number of active context features for a given event (word plus context). From the figures reported in [**Rat96**], our tagger is, experimentally, about ten times faster in both steps.

Another interesting issue to be investigated is the possibility of including the decision–tree–acquired relevant contexts in an ME approach, as a way to automatically acquire specialized features for the probabilistic model (see the discussion about specialized features in [**Rat96**]).

2.1.3. *Taggers that use Decision Trees.* DTs have been used in other approaches to tagging. For instance, in the case of Schmid [**Sch94b**] decision trees are used to estimate the probabilistic parameters of a statistical trigram–based tagger, in a way that is similar to our STT. However, in his approach the contextual information was not extended as in ours. Instead, trees are only involved in acquiring trigram probabilities.

In [**Mag95a**], Magerman and colleagues apply decision trees in a parsing oriented environment (SPATTER). In that approach, a decision–tree based POS tagger is one of the first components of the whole system, which is used in the probability calculation of syntactic structures. SPATTER achieves a tagging accuracy of 96.5% on the WSJ corpus. This tagger shares several features with ours, namely, the main attributes describing the surrounding contexts, the statistical usage of the decision trees, the division in word classes, etc. However, it is oriented to be included as a partial module in a broader system and it has a lot of particularities that make it more complicated to use. For instance, context features need to be initially converted into binary–valued features, and as a result its number is greatly increased; and, instead of pruning, the probability distributions of the obtained trees are smoothed by applying a sophisticated and costly adaptation of the Forward–Bakward algorithm, to maximize the probability of a held–out corpus. In contrast, our model is much simpler to acquire, and combined in a proper disambiguation algorithm, achieves a slightly better accuracy.

The MBT approach of Daelemans, et al. [**DZBG96**] can also be placed in this category. Despite MBT being formulated within the instance–based learning paradigm, it uses a particular tree–based representation (IGTREEs) of the base of examples (allowing a notable saving in space needs), which is then used for direct disambiguation. As it is noted in chapter 3, those trees are equivalent to the decision trees that would be inductively acquired with a simple feature selection criterion. The accuracy reported by this memory–based tagger (96.4%), applied to the same corpus and using the same tagset, is slightly lower than ours. Additionally, the amount of examples they used is almost double that used in our case. We also performed a comparison, under the same experimental conditions, of both algorithms when dealing with unknown words. Our algorithm performed better in terms of accuracy and space saving (see section 4.1 of chapter 3). We give experimental evidence that this is due to the feature selection function and to the branch merging strategy used in our algorithm. Conversely, it has to be said that MBT is slightly superior in tagging speed and design simplicity, which has permitted to

adapt and reuse the memory–based model to tackle several NLP problems regarding disambiguation.

2.1.4. *Other Machine Learning Based Taggers.* We discuss in this section the Transformation–based error–driven Learning (TBL) approach by Brill [**Bri95a**]. In this case, no direct comparison of taggers was performed, however the word–accuracy results (96.5%) reported by Brill on the Wall Street Journal using the same tagset and similar type of information are slightly worse than ours.

Regarding the contextual information it uses, it is very similar to the information used in our approach, and, in fact, it has been almost completely incorporated in our tree–modelling. In addition, some of the rule patterns for capturing collocational information used in Brill's approach have also been recast as composite attributes and included as new features in the learning algorithm. Experiments explaining these extensions are reported in chapter 3.

The acquisition step is the major bottleneck of TBL, since it performs a brute force search[9] within the space of all possible pattern instantiations to find the best transformation at each step[10]. From this point of view, our approach is clearly preferable. The tagging efficiency of its original version is also lower since, despite being linear on the input word–sequence length, the cost is multiplied by a constant equal to the number of acquired rules. Our basic RTT tagger also performs several passes through the input sequence but they are about 20 times lower. The implementation of Brill's rule–based tagger using FSTs [**RS95**] is much faster than our tree–based approach. However, a better implementation of our tagger (using *C*, and parallelizing the tree application for each word at a sentence level) would speed it up notably. We think that it would also be worthwhile to consider an implementation of the decision-tree rules using FST technology.

Another limitation of TBL is that, since it is non-statistical, it does not provide probability distributions and, unlike other approaches such as statistical taggers, DTs, or ME it cannot be used as a probabilistic component in a larger model.

Conversely, a clear advantage of Brill's approach is that the final language model is represented with a relatively small set of transformation rules, which require very little storage space, and which have an easy and direct interpretation. However, whether this information is useful from a linguistic perspective or not, is a matter that has not been thoroughly discussed by the author.

In a paper by Brill [**Bri95a**] it is shown that any binary decision tree can be converted into a transformation list (but not conversely), and a serious criticism is made against the decision tree approach by explaining some of the practical differences between TBL and the induction of decision trees[11]. We generally agree with that discussion, however most of the drawbacks that Brill indicates do not hold in our approach, which is more complex than the one that he considers. More precisely, there are two main differences: First, we use the decision trees as a *statistical component* instead of a classifier, and second, this component is integrated inside

[9]This is not exactly true since the search is data–driven and not all possible transformations are really tested. See [**RM94**] for an additional speed–up of Brill's learning algorithm by means of indexing the training corpus.

[10]Recently, Samuel [**Sam98**] has presented an efficient approximation called Lazy TBL which restrict the search to a small subset of all possible instantiations, by applying Monte Carlo sampling techniques. This approach allows the application of TBL to more complex tasks.

[11]In the same paper it is shown that TBL is equivalent to Decision List modelling.

a disambiguation algorithm which in turn can use other kinds of information and require the use of the tree–model several times during the classification process[12].

2.1.5. *Taggers by Combination.* There are two main approaches in tagger combination. On the one hand, it is possible to combine different types of knowledge, probably acquired from different sources and using different algorithms, inside the language model of a single tagger (*internal combination*). On the other hand, the combination can be done externally, by constructing an ensemble of taggers (preferably based on different principles and information), in which all taggers are constructed individually, and which is used to label new words by combining, in a sort of voting, the outcomes of each single tagger (*external combination*).

Examples of internal combination are the already cited papers [**OT96, Pad96, VP97, EAA+98, TO98, TRG97**]. Our contribution to this methodology is presented in chapter 4, in which we describe the way of including decision trees into the RELAX tagger. Direct comparison with the other approaches is not possible, since the corpora and languages of application are different. In addition, the resulting performance depends highly on the type and amount of knowledge used, something that greatly varies from one work to another. Compared to our other approaches to tagging, RELAX is more flexible than RTT and STT, and more accurate when it combines all available information. However, the efficiency is clearly lower, due to the intrinsic complexity of the convergence procedure of relaxation labelling.

Regarding the external combination, we should refer to the two recent works [**HZD98, BW98**], in which very good results were obtained. Such papers study the application of some algorithms for combining several preexisting taggers, however, we cannot say that they introduce *new* tagging paradigms. One problem of such systems is the efficiency degradation, since each individual tagger must vote for each input word to assign the combined result. In this way, the efficiency is forced to be lower than that of the least efficient tagger in the ensemble.

In our case, we have internally applied ensembles of decision trees only to those ambiguity classes which we consider worthy of attention. In section 2.5 of chapter 6 we have shown that this is a feasible way for improving our tree–based taggers. The accuracy reported is comparable to that of [**BW98**], obtained by combination of taggers. Finally, it must be said that our approach is not exclusive to the external combination, since any of our taggers could be individually included in an ensemble of taggers.

2.1.6. *Final Comments.* All throughout chapters 3, 4, 5, and 6, we have applied a number of machine–learning techniques, usually based on the combination of classifiers, in order to address some of the weak points in our approach, namely, the sparseness of some ambiguity classes, difficult subproblems, small training sets, etc. However, there still remains an important open problem, which is to efficiently adapt and tune taggers that have been acquired from a specific corpus onto another one —perhaps containing texts from other domains, where different lexical probabilities hold— Up to the present, not much interest has been devoted to this question, however, the very recent contribution of Roth and Zelenko [**RZ98**], represents a good effort in this direction. It is fair to say that this is the first tagger, to our knowledge, that addresses this problem with promising results. Their work presents an adaptive tagger which benefits from the ability of learning while testing

[12]Brill has in mind a decision tree based model which is used to classify the ambiguous words in a single pass through the corpus.

new examples (on-line/incremental learning). In this way, the tagger is dynamically tuned as the new examples of the corpus of application are presented to it.

Some of the already reviewed learning paradigms applied to POS tagging, namely TBL [**Bri95a**], MBT [**DBW97**] and ME [**Rat97b**] (see also the work by Cardie [**Car96a**] and Roth [**Rot98**]), have proven to be flexible models that can be easily adapted to address other NLP problems, e.g, word-sense disambiguation, PP-attachment disambiguation, etc. This is something that we still need to prove, before presenting our tree-based approach as a general framework to deal with NLP disambiguation problems. Some words are devoted to this issue in the concluding chapter 8.

2.2. Some Quantitative Information. As we have seen, the comparison of POS taggers is a delicate issue since many factors can affect the final accuracy figures: different sizes for training and test sets, evaluation over different corpora —and thus, different tag distributions—, different tag sets, etc. As we have seen in this chapter, even when comparing systems under similar conditions, results might be non-conclusive if noisy material is used for the test.

Nevertheless, an informative comparison between our system and several of the most representative current taggers is described and discussed in section 2.5 of chapter 6. The figures reported come from testing the taggers on the WSJ corpus, under the open vocabulary assumption, and and using training and test sets of similar sizes.

CHAPTER 8

Conclusions

1. Summary

In this dissertation we have exposed the research carried out on applying machine learning algorithms to POS tagging. In particular, we used the induction of decision trees to acquire the statistical knowledge about part–of–speech ambiguities for a posterior use in different disambiguation algorithms. As a first step, we have experimentally proven that DTs provide a flexible (by allowing a rich feature representation), efficient and compact way for acquiring, representing and accessing such information. In addition to that, DTs provide proper estimations of conditional probabilities for tags and words in their particular contexts. Additional machine learning techniques, based on the combination of classifiers, have been applied to address some particular weaknesses of our tree–based approach, and to further improve the accuracy in the most difficult cases.

As a second step, the acquired models have been used to construct simple, accurate and effective taggers, based on different paradigms. In particular, we present three different taggers that include the tree–based models: RTT, STT, and RELAX, which have shown different properties regarding speed, flexibility, accuracy, etc. The idea is that the particular user needs and environment will define which is the most appropriate tagger in each situation. Although we have observed slight differences, the accuracy results for the three taggers, tested on the WSJ test bench corpus, are uniformly very high, and, if not better, they are at least as good as those of a number of current taggers based on automatic acquisition (a qualitative comparison with the most relevant current work is reported in chapter 7).

Additionally, our approach has been adapted to annotate a general Spanish corpus, with the particular limitation of learning from small training sets. A new technique, based on tagger combination and bootstrapping, has been proposed to address this problem and to improve accuracy. Experimental results showed that very high accuracy is possible for Spanish tagging, with a relatively low manual effort. Additionally, the success in this real application has confirmed the validity of our approach, and the validity of the previously presented portability argument in favour of automatically acquired taggers.

The main contributions of this thesis were advanced in the introductory chapter, and they have been described in detail within the subsequent chapters. The next section is a very brief final resume of the main contributions to the dissertation, including the pointers to the chapters in which they appear.

1.1. Contributions. The core of the dissertation presented encompass scientific contributions that can be categorized in the following three groups.

1. The application of decision trees in the POS tagging problem is the basic point of the thesis. This contribution is threefold. First, DTs were used

to acquire, represent and access language models oriented to POS tagging. This work is described in chapter 3, jointly with a very satisfactory test of the acquired models. Second, the acquired models have been applied to three different, accurate and effective taggers. The results presented in chapter 4 show that very good accuracy can be obtained, overcoming some of the current best approaches to tagging. And third, the combination of different kinds of information in a rich language modelling, is performed within a flexible tagger, to further improve the previously reported results (also in chapter 4).

2. Machine learning techniques for constructing ensembles of classifiers and combining individual outcomes have been proposed and applied in our particular domain. In this case, the contribution is twofold. First, we applied ensembles of decision trees to improve results in the special cases of: *difficult* and *sparse* ambiguity classes. The full work is described in chapter 6, except for the method that deals with sparseness which is covered in chapter 3. Second, a bootstrapping method to develop accurate taggers from small training sets was proposed in chapter 5. The method is based on the combination of taggers to generate more accurate *retraining* corpora. It has been successfully tested on a Spanish corpus (see also chapter 5).
3. The evaluation and comparison of taggers is an issue that presents severe difficulties, particularly when commonly used test bench corpora contain noise. This issue is addressed in chapter 7, where a deep analysis demonstrates that usual statistical confidence intervals fail in the presence of noise.

The following two items describe other relevant contributions of the present thesis.

1. With respect to the bibliographical issues, we would like to mention that chapter 2 supplies a detailed survey of POS tagging, and, more important, a valuable broad-coverage compilation of references linking the fields of ML and NLP. Additionally, a qualitative/quantitative comparison of our system to other relevant important approaches to tagging is included in chapter 7.
2. From a practical perspective, we would like to emphasize that several tools and resources have been generated as the result of this work, and the rich research environment in which it has been carried out. Chapter 5 includes pointers to the aforementioned tools, resources and related research projects.

2. Further Research Directions

Further work is still to be done in several directions. Some of this corresponds to open research lines introduced in certain of our chapters, while others simply refer to given specific details of implementation, tuning, etc. We have categorized them in five groups, which are briefly described in the next sections.

2.1. On Acquiring Better Models through Decision Trees.

- Regarding the used TDIDT algorithm, there are some possible extensions that have not been taken into consideration, and we think that they should be studied, i.e., searching with a limited lookahead, to mitigate the effect of the greedy search, and including multivariate splits[1].

[1]However, Murthy [Mur95] notes that such extensions are not guaranteed to work well in all domains, and that, apart from the computational overhead they introduce, they can be

- With respect to the information that this algorithm uses, we would like to consider some extensions. In this thesis, we have studied many variations on the set of features for describing the training examples, however, all of them refer to quite local context information. In order to deal with exceptions, rare events, and difficult cases, we should extend it to the consideration of long distance relations, semantic features, etc. In addition, the inclusion within the RELAX environment of better and richer models, obtained by manual development, would be of considerable interest to improve the results of the already presented Spanish tagging, and for a forthcoming application to Catalan.
- Finally, regarding the sparseness problem of the ambiguity–class approach, better models have been acquired by applying some proposed techniques. Particularly, the application of CPD [**Bre98b**] reported a 10% reduction on the average error rate of sparse ambiguity classes (see chapter 6). However, this improvement results in a very small contribution to the global tagging accuracy since sparse ambiguity classes have a small relative weight. To make the global improvement significant, we should be able to to fix their errors in a much greater proportion than we really did (otherwise the effort could be considered worthless). The techniques applied are certainly a promising starting point to fulfill the preceding goal, however to find a procedure to clearly improve results in sparse ambiguity classes is still an open issue that requires further research.

2.2. On the Application of Tree–based Models.

- There are a number of technical questions about the already implemented taggers that should be addressed further. First, regarding the RTT algorithm, the convergence properties have to be properly studied. Second, all three taggers, namely, RTT, STT, and RELAX, can clearly be accelerated, by means of changing the programming language, or by parallelizing some important parts of the algorithm[2]. Finally, the study of the interaction between the different knowledge included and its effect on the tagging results is especially important in the RELAX environment in order to be able to properly answer the question: How can we add more information? For instance, it has to be determined if redundant constraints[3] can be added without care, or if, conversely, a preprocessing step is needed to eliminate redundancy.
- The application of trees to taggers other than the three previously mentioned could be also of interest. In particular, we are interested on including the knowledge acquired with DTs to automatically enrich the ME model with some specific relevant features.

2.3. On the Construction of Ensembles of Classifiers. The work described in chapter 6 demonstrates that the use of ensembles of decision trees allows to significantly improve the accuracy of the presented taggers. However, there are a number of technical questions regarding the methods used to construct ensembles

counterproductive in some cases. He identifies some of the possible pathologies and suggests the type of domains they are suited for.

[2]Additionally, RTT and STT work with an underlying rule–based model, thus they could probably be recasted to fit the FST technology.

[3]The presence of contradictory information in the constraint–based model would probably represent an even more serious problem.

that are still open questions. Which are the best techniques to generate ensembles in our domain? Which is the algorithm for combining the individual trees that performs best? Which is the best way to incorporate an ensembles into the disambiguation algorithms?

There are other more specific points (already discussed in chapter 6) that we would also like to mention here.

- We think that it should be thoroughly studied why there are methods that do not work well in our domain. Regarding this issue, we are especially interested on studying the application of the boosting (ADABOOST) algorithm, which has proven to be ineffective in the task at hand. It is known that ADABOOST tends to overfit the training data in the presence of noise and that it usually places much weight on 'hard areas' and outliers. Therefore, the manually study of the highly emphasized examples in the final distribution should provide insight about difficult examples, infrequent events, and noise in the training set. The first two would be useful to determine which type of information is needed to resolve the most difficult cases, perhaps treating them separately. The third would be useful to apply some kind of pre–process to filter out noisy examples from training data.
- It is well–known that the success of applying ensembles of classifiers strongly depends on the ability of learning classifiers that commit uncorrelated errors. In our case, once an ensemble of decision trees is constructed it would be interesting to be able to evaluate how 'different' are the trees of the ensembles (i.e., how complementary is the information represented in them), to try to predict whether it is worthwhile to include this ensemble or not, to eliminate highly–correlated members, etc. In this direction we are trying to derive a measure of similarity between decision trees (by exploring the redundant, complementary, and contradictory information), which should be highly correlated with the effectiveness of the ensemble.
- Regarding the combination of decision trees: We are involved in testing more sophisticated methods of classifier combination, such as weighted voting methods, and different levels of stacked generalization [**Wol92**] using alternative learning algorithms.
- We are also interested on extending the experiments involving the combination of more than two taggers in a double direction: first, to obtain less noisy corpora for the retraining steps in bootstrapping processes; and second, to include our best tagger in an ensemble of preexisting taggers to increase global tagging accuracy. This work would be done in the same direction as that of [**HZD98, BW98**], but it would also include an application to Spanish and Catalan.

2.4. On the Tagging Evaluation and Comparison.

- In chapter 7 we pointed out some difficulties in tagging evaluation and comparison of results, and we studied a particular case related to the noise present in the test corpus. With respect to this study, further research is needed to derive a more conservative statistical measure of significance, allowing us to reject or accept the hypothesis that both taggers are different, tested under noisy conditions. Another related issue would be the application of machine–learning algorithms (perhaps in combination with other

semi–automatic techniques) to identify and filter noisy examples, and to properly define some kind of pre–processing step of the training and test annotated corpora.

- The wide–spread application of machine learning techniques in addressing NLP problems is still a very recent phenomenon, and so, rigorous experimental settings for comparing different systems, which have a long tradition in the ML field, are not yet a common practice in NLP works. This lack of rigorousness is extensive to POS tagging, for which indirect comparisons have been generally performed. We think that a comprehensive experimental comparison of the most promising current approaches to tagging would be very valuable to the NLP community. Such a comparison, performed under exactly the same experimental conditions and using clear and well–defined statistical tests, should cover quantitative and qualitative aspects of different systems. In particular, varying features should be compared within several scenarios. These features include: corpus of application, tagset, available information, size of the training set, unknown words, different levels of noise, etc. This is a work that we plan to address in the short term.
- In a close relation to the preceding points, we think that other issues than simply 'accuracy rates' are becoming more important in order to test and evaluate the real utility of different approaches for tagging. Such aspects, which are being the focus of increasing attention in the NLP community, refer to test the ability of adapting to new domains (tuning), to study the types of errors committed and their influence on the task at hand, to study the adequacy of the tagset granularity and composition to the application domain, to verify the language independence assumption, etc.

2.5. On the Application to other NLP Problems and Languages. With the aim of providing general paradigms in which many different ambiguity problems could be properly addressed, some of the already presented machine–learning approaches have been applied to tackle other classical NLP disambiguation problems, typically, word–sense disambiguation, PP–attachment disambiguation, partial parsing, text–to–speech processing, etc. This is the case of Brill's Transformation–based Learning [**Bri95a**], the Instance–based learning environment proposed by Cardie [**Car96a**], the Memory–based approach of Daelemans and colleagues at Tilburg University [**DZBG96**], and the Maximum Entropy approach, reported basically by Ratnaparkhi [**Rat96**]. We think that our approach is also general enough to properly address the different alternative problems. Particularly, we plan to start with WSD, PP–attachment disambiguation problems in the near future. An additional objective to our field of concern is to simultaneously address several disambiguation problems. We believe, as other authors do, that we can take advantage of the interactions between different level tasks.

In another direction, we plan to apply the techniques presented to develop taggers and annotated corpora for other languages, and in particular, for Catalan, for which we have already started to develop linguistic tools, and we plan to carry out substantial work in the near future.

Bibliography

[AAA+95] I. Aduriz, I. Alegria, J.M. Arriola, X. Artola, A. Díaz de Ilarraza, N. Ezeiza, K. Gojenola, and M. Maritxalar. Different Issues in the Desing of a Lemmatizer/Tagger for Basque. In *Proceedings of the EACL SIGDAT Workshop*, Dublin, Ireland, 1995.

[AAC+94] S. Acebo, A. Ageno, S. Climent, J. Farreres, L. Padró, F. Ribas, H. Rodrí.guez, and O. Soler. MACO: Morphological Analyzer Corpus-Oriented. Working Paper #31, ESPRIT BRA-7315 Acquilex II, 1994.

[AB96] C. Aone and W. Bennett. Evaluating Automated and Manual Acquisition of Anaphora Resolution. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[Abn91] S. Abney. *Parsing by Chunks.* R. Berwick, S. Abney and C. Tenny (eds.) Principle-based Parsing . Kluwer Academic Publishers, Dordrecht, 1991. http://www.sfs.nphil.uni-tuebingen.de/~abney.

[Abn96] Steven Abney. *Statistical Methods and Linguistics.* Judith Klavans and Philip Resnik (eds.) The Balancing Act. MIT Press, Cambridge, MA, 1996. http://www.sfs.nphil.uni-tuebingen.de/~abney.

[Abn97] Steven Abney. *Part-of-Speech Tagging and Partial Parsing.* S. Young and G. Bloothooft (eds.) Corpus-Based Methods in Language and Speech Processing. An ELSNET book. Kluwer Academic Publishers, Dordrecht, 1997. http://www.sfs.nphil.uni-tuebingen.de/~abney.

[ACO92] S. Atkins, J. Clear, and N. Ostler. Corpus Design Criteria. *Literary and Linguistic Computing*, 7, 1992.

[ADK98] S. Argamon, I. Dagan, and Y. Krymolowski. A Memory-based Approach to Learning Shallow Natural Language Patterns. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 67–73, Montréal, Canada, 1998. cmp-lg/9806011.

[AH96] C. Aone and K. Hausman. Unsupervised Learning of a Rule-based Spanish Part-of-speech Tagger. In *Proceedings of the 16th International Conference on Computatioal Linguistics, COLING*, pages 53–58, August 1996.

[Aha97] David Aha. *Lazy Learning.* Kluwer Academic Publishers, Dordrecht, 1997. Reprinted from: Artificial Intelligence Review, 11:1–5.

[AK87] E. H. Aarts and J. H. Korst. Boltzmann machines and their applications. In *J. W. de Bakker, A.J. Nijman & P.C. Treleaven (Editors). Proceedings PARLE (Parallel Architectures and Languages Europe)*, 1987. Lecture Notes in Computer Science, Vol. 258.

[AKA91] D. Aha, D. Kibler, and M. Albert. Instance-based Learning Algorithms. *Machine Learning*, 7:37–66, 1991.

[Ali96] K. M. Ali. On explaining degree of error reduction due to combining multiple decision trees. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 1–7, 1996. http://www.cs.fit.edu/~imlm.

[AN93] G. Adams and E. Neufeld. Automated Word-Class Tagging of Unseen Words in Text. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 1993.

[AP96] K. M. Ali and M. J. Pazzani. Error Reduction through Learning Multiple Descriptions. *Machine Learning*, 24(3):173–202, 1996.

[Bau72] L. E. Baum. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of a Markov Process. *Inequalities*, 3:1–8, 1972.

[BB87] M. Ben-Bassat. Use of Distance Measure, Information Measures, and Error Bounds on Feature Evaluation. In P. R. Krishnaiah and L. N. Kanal, editors, *Classification, Pattern Recognition and Reduction of Dimensionality*, pages 773–791. North–Holland Publishing Company, Amsterdam, 1987.

[BBDM89] L. R. Bahl, P. F. Brown, P. V. DeSouza, and R. L. Mercer. A Tree-Based Statistical Language Model for Natural Language Speech Recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(7):1001–1008, 1989.

[BCK96] I. Bratko, B. Cestnik, and I. Kononenko. Attribute-Based Learning. *AI Communications*, 9, 1996.

[BCPS94] E. J. Briscoe, J. Carrol, L. Padró, and I. Serail. Hybrid Techniques for Training Part–of–speech Taggers. Working Paper #45, ESPRIT BRA–7315 Acquilex II, 1994.

[BD99] G. Bakiri and T. G. Dietterich. Achieving High–Accuracy Text–to–Speech with Machine Learning. In B. Damper, editor, *Data Mining in Speech Synthesis*. Chapman and Hall, To appear in 1999.

[BDW96] A. van den Bosch, W. Daelemans, and T. Weijters. Morphological Analysis as Classification: an Inductive–Learning Approach. In *Proceedings of the 2nd NeMLaP*, 1996. cmp-lg/9607021.

[BEKS98] E. Black, S. Eubank, H. Kashioka, and J. Saia. Reinventing Part–of–speech Tagging. *Journal of Natural Language Processing (Japan)*, 5(1), 1998.

[BFHM98] Eric Brill, Radu Florian, John C. Henderson, and Lidia Mangu. Beyond N-Grams: Can Linguistic Sophistication Improve Language Modeling? In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 186–190, Montréal, Canada, 1998. http://www.cs.jhu.edu/~brill/acadpubs.html.

[BFK98] Ezra Black, Andrew Finch, and Hideki Kashioka. Trigger-Pair Predictors in Parsing and Tagging. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 131–136, Montréal, Canada, August 1998.

[BFOS84] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. The Wadsworth Statistics/Probability Series. Wadsworth International Group, Belmont, CA, 1984.

[Bib93] D. Biber. Representativeness in Corpus Design. *Computational Linguistics*, 19(2), 1993.

[BJL+92a] E. Black, F. Jelinek, J. Lafferty, D. Magerman, R. L. Mercer, and S. Roukos. Towards History–based Grammars: Using Richer Models for Probabilistic Parsing. In *Proceedings of the DARPA Workshop on Speech and Natural Language*, San Mateo, CA, 1992. cmp-lg/9405007.

[BJL+92b] E. Black, F. Jelinek, J. Lafferty, R. L. Mercer, and S. Roukos. Decision Tree Models Applied to the Labeling of Text with Parts–of–speech. In *Proceedings of the DARPA Workshop on Speech and Natural Language*, San Mateo, CA, 1992.

[BJM83] L. R. Bahl, F. Jelinek, and R. L. Mercer. A Maximum–Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 5(2):179–190, March 1983.

[BK99] E. Bauer and R. Kohavi. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. *Machine Learning Journal. Special issue on IMLM for Improving and Scaling Machine Learning Algorithms*, 36(1&2), 1999. http://robotics.stanford.edu/~ronnyk.

[BL97] A. L. Blum and P. Langley. Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, 97:245–271, 1997.

[BM76] L. R. Bahl and R. L. Mercer. Part–of–speech Assignment by a Statistical Decision Algorithm. In *IEEE International Symposium on Information Theory*, pages 88–89, 1976.

[BM98] A. Blum and T. Mitchell. Combining Labeled and Unlabeled Data with Co–Training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory, COLT-98*, pages 92–100, Madison, Wisconsin, 1998.

[BN92] W. Buntine and T. Niblett. A Further Comparison of Splitting Rules for Decision-tree Induction. *Machine Learning*, 8:75–85, 1992.

[BPP96] A. Berger, S. Della Pietra, and V. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–71, 1996.

[BPPM91] P. F. Brown, S. Della Pietra, V. Della Pietra, and R. L. Mercer. Word Sense Disambiguation using Statistical Methods. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 264–270, 1991.

[BPS+92] P. Brown, V. Della Pietra, P. de Souza, J. Lai, and R. Mercer. Class-based n-gram Models of Natural Language. *Computational Linguistics*, 18(4):467–480, 1992.

[BR94] E. Brill and P. Resnik. A Rule-Based Approach to Prepositional Phrase Attachment Disambiguation. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING*, Kyoto, Japan, August 1994. http://www.cs.jhu.edu/~brill/acadpubs.html.

[Bre96a] L. Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, 1996.

[Bre96b] L. Breiman. Stacked Regressions. *Machine Learning*, 24(3):49–64, 1996.

[Bre97] L. Breiman. Arcing the Edge. Technical Report 486, Statistics Department. University of California, Berkeley, CA, 1997. http://www.stat.berkeley.edu/users/breiman.

[Bre98a] L. Breiman. Arcing Classifiers. *The Annals of Statistics*, 26(3):801–849, 1998.

[Bre98b] L. Breiman. Using Convex Pseudo-Data to Increase Prediction Accuracy. Technical Report, Statistics Department. University of California, Berkeley, CA, 1998. http://www.stat.berkeley.edu/users/breiman.

[Bri92] Eric Brill. A Simple Rule-Based Part-of-speech Tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing, ANLP*, pages 152–155. ACL, 1992.

[Bri93] E. Brill. Automatic Grammar Induction and Parsing Free Text: A Transformation-based Approach. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 1993. http://www.cs.jhu.edu/~brill/acadpubs.html.

[Bri94a] Eric Brill. Some Advances in Rule-based Part-of-speech Tagging. In *Proceedings of the 12th National Conference on Artificial Intelligence, AAAI*, pages 722–727, 1994. http://www.cs.jhu.edu/~brill/acadpubs.html.

[Bri94b] E. J. Briscoe. *Prospects for Practical Parsing of Unrestricted Text: Robust Statistical Parsing Techniques*. N. Oostdijk and P. de Haan (eds.), Corpus-Based Research into Language. Rodopi, Amsterdam, 1994.

[Bri95a] E. Brill. Transformation-based Error-driven Learning and Natural Language Processing: A Case Study in Part-of-speech Tagging. *Computational Linguistics*, 21(4):543–565, 1995.

[Bri95b] Eric Brill. Unsupervised Learning of Disambiguation Rules for Part-of-speech Tagging. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 1–13, Massachusetts, 1995. http://www.cs.jhu.edu/~brill/acadpubs.html.

[Bro93] C. E. Brodley. Addressing the selective superiority problem: Automatic algorithm/model class selection. In *Proceedings of the 10th International Conference on Machine Learning, ICML'93*, pages 17–24, 1993.

[Bro95] C. E. Brodley. Recursive Automatic Bias Selection for Classifier Construction. *Machine Learning*, 20:63–88, 1995.

[Bro96] C. E. Brodley. Creating and exploiting coverage and diversity. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 8–14, 1996. http://www.cs.fit.edu/~imlm.

[BS95] T. Brants and C. Samuelsson. Tagging the TELEMAN Corpus. In *Proceedings of the 10th Nordic Conference of Computational Linguistics*, 1995. cmp-lg/9505026.

[BSK97] T. Brants, W. Skut, and B. Krenn. Tagging Grammatical Functions. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, Brown University, Providence, RI, 1997. cmp-lg/9707015.

[Bun90] W. Buntine. *A Theory of Learning Classification Rules*. Phd. Thesis, School of Computing Science, University of Technology, Sydney, Australia, 1990.

[BW98] Eric Brill and Jun Wu. Classifier Combination for Improved Lexical Disambiguation. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 191–195, Montréal, Canada, 1998. http://www.cs.jhu.edu/~brill/acadpubs.html.

[BWD98] A. van den Bosch, T. Weijters, and W. Daelemans. Modularity in Inductively-learned Word Pronunciation Systems. In *Proceedings of the NeMLaP-3/CoNLL'98*, pages 185–194, 1998. cmp-lg/9801004.

[Car92] Claire Cardie. Learning to Disambiguate Relative Pronouns. In *Proceedings of the 10th National Conference on Artificial Intelligence, AAAI*, pages 38–43, San Jose, CA, 1992. AAAI Press / MIT Press.

[Car93a] Claire Cardie. A Case-based Approach to Knowledge Acquisition for Domain-specific Sentence Analysis. In *Proceedings of the 11th National Conference on Artificial Intelligence, AAAI*, pages 798–803, 1993. AAAI Press / MIT Press.

[Car93b] Claire Cardie. Using Decision Trees to Improve Case-based Learning. In *Proceedings of the 10th International Conference on Machine Learning, ICML'93*, pages 25–32, Amherst, MA, 1993. Morgan Kaufmann.

[Car94] Claire Cardie. *Domain Specific Knowledge Acquisition for Conceptual Sentence Analysis*. Phd. Thesis, University of Massachusets, 1994. Available as University of Massachusetts, CMPSCI Technical Report 94-74.

[Car96a] C. Cardie. Embedded Machine Learning Systems for Natural Language Processing: A General Framework. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[Car96b] Claire Cardie. Automatic Feature Set Selection for Case-Based Learning of Linguistic Knowledge. In *Proceedings of the 1st Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 113–126, 1996.

[Car96c] R. Caruana. Algorithms and Applications for Multitask Learning. In L. Saitta, editor, *Proceedings of the 13th International Conference on Machine Learning, ICML'96*, pages 87–95, San Francisco, CA, 1996. Morgan Kaufmann.

[CB95] M. Collins and J. Brooks. Prepositional phrase Attachment Trhough a Backed-off Model. In *Proceedings of the 3rd Workshop on Very Large Corpora*, Massachusetts, 1995.

[CCA+94] E. Charniak, G. Carroll, J. Adcock, A. Cassandra, Y. Gotoh, J. Katz, M. Littman, and J. McCann. Taggers for Parsers. Technical Report CS-94-06, Brown University, 1994. To appear in Artificial Intelligence magazine. http://www.cs.brown.edu/people/ec.

[CCA98] I. Castellón, M. Civit, and J. Atserias. Syntactic Parsing of Unrestricted Spanish Text. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, LREC*, pages 603–610, Granada, Spain, May 1998.

[CCM+98] J. Carmona, S. Cervell, L. Màrquez, M.A. Martí, L. Padró, R. Placer, H. Rodríguez, M. Taulé, and J. Turmo. An Environment for Morphosyntactic Processing of Unrestricted Spanish Text. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, LREC*, pages 915–922, Granada, Spain, May 1998.

[CCP95] S. Cervell, S. Climent, and R. Placer. Using MACO and MDS to tag a balanced corpus of Spanish. Working Paper, ESPRIT BRA-7315 Acquilex II, 1995.

[Cer96] Instituto Cervantes. Informe sobre recursos lingüísticos para el español (II). Internal Report, 1996.

[Ces90] B. Cestnik. Estimating Probabilities: A Crucial Task in Machine Learning. In *Proceedings of the European Conference on Artificial Intelligence*, pages 147–149, Stockholm, August 1990.

[CG91] K. W. Church and W. A. Gale. A Comparison of the Enhanced Good-Turing and Deleted Estimation Methods for Estimating Probabilities of English Bigrams. *Computer Speech and Language*, 5:19–54, 1991.

[CG96] S. F. Chen and J. Goodman. An Empirical Study of Smoothing Techniques for Language Modeling. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*. ACL, 1996.

[CGM96] H. Chipman, E. George, and R. McCulloch. Bayesian CART. Technical Report, Department of Statistics, University of Chicago, 1996. http://gsbrem.uchicago.edu/Papers/cart.ps.

[Cha93] E. Charniak. *Statistical Language Learning.* The MIT Press, Cambridge, Massachusetts, 1993.

[Cha97] E. Charniak. Statistical Techniques for Natural Language Parsing. *AI Magazine*, 1997. http://www.cs.brown.edu/people/ec.

[Che96] K.J. Cherkauer. Human Expert-level Performance on a Scientific Image Analysis Task by a System Using Combined Artificial Neural Networks. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 15–21, 1996. http://www.cs.fit.edu/~imlm.

[CHJP93] E. Charniak, C. Hendrickson, N. Jacobson, and M. Perkowitz. Equations for Part-of-speech tagging. In *Proceedings of the Conference of the American Association for Artificial Intelligence, AAAI*, 1993. http://www.cs.brown.edu/people/ec.

[Chr97] R. Christensen. *Log-Linear Models and Logistic Regression.* Springer Texts in Statistics, G. Casella, S. Fienberg and I. Olkin (eds.). Springer, 1997.

[Chu88] K. W. Church. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In *Proceedings of the 1st Conference on Applied Natural Language Processing, ANLP*, pages 136–143. ACL, 1988.

[Chu92] K. W. Church. *Current Practice in Part of Speech Tagging and Suggestions for the Future.* Simmons (ed.), *Sbornik praci: In Honor of Henry Kučera.*, Michigan Slavic Studies, 1992.

[CKB87] B. Cestnik, I. Kononenko, and I. Bratko. ASSISTANT86: A Knowledge Elicitation Tool for Sophisticated Users. In I. Bratko and N. Lavrač, editors, *Progress in Machine Learning.* Sigma Press, Wilmslow, England, 1987.

[CKPS92] D. Cutting, J. Kupiec, J. Pederson, and P. Sibun. A Practical Part-of-speech Tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing, ANLP*, pages 133–140. ACL, 1992.

[Cle89] R. T. Clemen. Combining Forecasts: A Review and Annotated Bibliography. *International Journal of Forecasting*, 5:559–583, 1989.

[CM93] K. W. Church and R. L. Mercer. Introduction to the Special Issue on Computational Linguistics Using Large Corpora. *Computational Linguistics*, 19(1), 1993.

[CM97] M. E. Califf and R. J. Mooney. Applying ILP-based Techniques to Natural Language Information Extraction: An Experiment in Relational Learning. In *Workshop Notes of the IJCAI-97 Workshop on Frontiers of Inductive Logic Programming*, pages 7–11, Nagoya, Japan, 1997.

[CN89] P. Clark and T. Niblett. The CN2 Induction Algorithm. *Machine Learning*, 3:261–284, 1989.

[CS95] P. K. Chan and S. J. Stolfo. Learning Arbiter and Combiner Trees from Partitioned Data for Scaling Machine Learning. In *Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining*, pages 39–44, Menlo Park, CA, 1995. AAAI Press.

[CS96] W. W. Cohen and Y. Singer. Context-sensitive Learning Methods for Text Categorization. In *Proceedings of the 19th Annual Inter. ACM Conference on Research and Development in Information Retrieval*, 1996.

[CSL93] T. Chen, V. Soo, and A. Lin. Learning to Parse with Recurrent Neural Networks. In *Proceedings of European Conference on Machine Learning Workshop on Machine Learning and Text Analysis, ECML*, pages 63–68, 1993.

[CT91] T. M. Cover and J. A. Thomas, editors. *Elements of Information Theory.* John Wiley & Sons, 1991.

[CT95] J-P. Chanod and P. Tapanainen. Tagging French – Comparing a Statistical and a Constraint-Based Method. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, pages 149–156, Dublin, Ireland, 1995.

[Cut93] D. Cutting. Porting a Stochastic Part-of-Speech Tagger to Swedish. In *Proceedings of the 9th Nordic Conference of Computational Linguistics*, Stockholm, Sweden, 1993.

[CWG96] H. Cunningham, Y. Wilks, and R. Gaizauskas. GATE - a General Architecture for Text Engineering. In *Proceedings of 16th International Conference on Computational Linguistics, COLING*, Copenhagen, Denmark, 1996.

[Dae95] W. Daelemans. *Memory-based Lexical Acquisition and Processing.* Machine Translation and the Lexicon, Lecture Notes in Artificial Intelligence 898. P. Steffens editor. Springer, Berlin, 1995.

[Dag98] I. Dagan. Lexical Statistical Methods in Natural Language Processing. In *Joint COLING/ACL Tutorial Program*, Montréal, Canada, 1998. Tutorial Notes.

[DB91] T. G. Dietterich and G. Bakiri. Error–Correcting Output Codes: A General Method for Improving Multiclass Inductive Learning Programs. In *Proceedings of the 9th National Conference on Artificial Intelligence, AAAI*, pages 572–577. AAAI Press / MIT Press, 1991.

[DB95] T. G. Dietterich and G. Bakiri. Solving Multiclass Learning Problems via Error–Correcting Output Codes. *Journal of Artificial Intelligence Research*, 2:263–286, 1995.

[DBG96] W. Daelemans, P. Berck, and S. Gillis. Unsupervised Discovery of Phonological Categories through Supervised Learning of Morphological Rules. In *Proceedings of 16th International Conference on Computational Linguistics, COLING*, pages 95–100, Copenhagen, Denmark, 1996.

[DBUG98] A. Díaz, M. de Buenaga, L. A. Ureña, and M. García. Integrating Linguistic Resources in a Uniform Way for Text Classification Tasks. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, LREC*, pages 1197–1204, Granada, Spain, May 1998.

[DBW97] W. Daelemans, A. van den Bosch, and T. Weijters. *IGTree: Using Trees for Compression and Classification in Lazy Learning Algorithms.* D. Aha (ed.), Artificial Intelligence Review 11, Special issue on Lazy Learning. Kluwer Academic Publishers, 1997.

[DC96] H. Drucker and C. Cortes. Boosting Decision Trees. In D.S. Touretzky, M.C. Mozer and M.E. Hesselmo, editor, *Advances in Neural Information Processing Systems 8, NIPS'95*, pages 470–485. MIT Press, Cambridge, MA, 1996.

[DeM90] C. DeMarcken. Parsing the LOB Corpus. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 243–259, Pittsburgh, PA, 1990. ACL.

[DeR88] S. J. DeRose. Grammatical Category Disambiguation by Statistical Optimization. *Computational Linguistics*, 14:31–39, 1988.

[DH73] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* Wiley, 1973.

[Die97] T. G. Dietterich. Machine Learning Research: Four Current Directions. *AI Magazine*, 18(4):97–136, 1997.

[Die98a] T. G. Dietterich. An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. *Machine Learning*, pages 1–22, 1998.

[Die98b] T. G. Dietterich. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, 10(7), 1998.

[DK95a] E. Dermatas and G. Kokkinakis. Automatic Stochastic Tagging of Natural Language Texts. *Computational Linguistics*, 21(2):137–164, 1995.

[DK95b] T. G. Dietterich and E. B. Kong. Machine Learning Bias, Statistical Bias, and Statistical Variance of Decision Tree Algorithms. Technical Report, Department of Computer Science, Oregon State University, 1995. http://www.cs.orst.edu/~tgd/cv/pubs.html.

[DKM96] T. G. Dietterich, M. Kearns, and Y. Mansour. Applying the Weak Learning Framework to Understand and Improve C4.5. In L. Saitta, editor, *Proceedings of the 13th International Conference on Machine Learning, ICML'96*, pages 96–104, San Francisco, CA, 1996. Morgan Kaufmann.

[DKR97] I. Dagan, Y. Karov, and D. Roth. Mistake-Driven Learning in Text Categorization. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, Brown University, Providence, RI, 1997.

[DLR77] A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data Via the EM Algorithm. *Journal of the Royal Statistical Society*, 39, 1977.

[DM84] A. M. Derouault and B. Merialdo. Language Modelling at the Syntactic Level. In *Proceedings of the 7th International Conference on Pattern Recognition*, 1984.

[DTS98] L. Dini, V. Di Tomaso, and F. Segond. Word Sense Disambiguation with Functional Relations. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, LREC*, pages 1189–1196, Granada, Spain, May 1998.

[DWB97] W. Daelemans, T. Weijters, and A. van den Bosch. *Empirical Learning of Natural Language Processing Tasks*. Lecture Notes in Artificial Intelligence, number 1224. Springer–Verlag, Berlin, 1997.

[DZB96] W. Daelemans, J. Zavrel, and P. Berck. Part-of-Speech Tagging for Dutch with MBT, a Memory-based Tagger Generator. In *Congresboek van de Interdisciplinaire Onderzoeksconferentie Informatiewetenchap*, TU Delft, 1996.

[DZBG96] W. Daelemans, J. Zavrel, P. Berck, and S. Gillis. MBT: A Memory–Based Part–of–speech Tagger Generator. In *Proceedings of the 4th Workshop on Very Large Corpora*, pages 14–27, Copenhagen, Denmark, 1996.

[EAA+98] N. Ezeiza, I. Aduriz, I. Alegria, J.M. Arriola, and R. Urizar. Combining Stochastic and Rule-Based Methods for Disambiguation in Agglutinative Languages. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 379–384, Montréal, Canada, August 1998.

[ED96] S. P. Engelson and I. Dagan. Minimizing Manual Annotation Cost in supervised Training from Corpora. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[Edw93] J. Edwards. Survey of Electronic Corpora and Related Resources for Language Researchers. In J. Edwards and M. Lampert, editor, *Talking Data: Transcription and Coding in Discourse Research*. Erlbaum, London & Hillsdale, 1993.

[EG93] M. Eineborg and B. Gambäck. Tagging Experiments Using Neural Networks. In *Proceedings of the 9th Nordic Conference of Computational Linguistics*, Stockholm, Sweden, 1993.

[Elw93] D. Elworthy. Part–of–speech and Phrasal Tagging. Working Paper #10, ESPRIT BRA–7315 Acquilex II, 1993.

[Elw94] D. Elworthy. Does Baum–Welch Re–estimation Help Taggers? In *Proceedings of the 4th Conference on Applied Natural Language Processing, ANLP*, pages 53–58. ACL, 1994.

[EMS97] F. Esposito, D. Malerba, and G. Semeraro. A Comparative Analysis of Methods for Pruning Decision Trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):476–491, May 1997.

[FCS96] D. W. Fan, P. K. Chan, and S. J. Stolfo. A comparative evaluation of combiner and stacked generalization. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 40–46, 1996. http://www.cs.fit.edu/~imlm.

[Fel95] H. Feldweg. Implementation and Evaluation of a German HMM for POS Disambiguation. In *Proceedings of the EACL SIGDAT Workshop*, Dublin, Ireland, August 1995.

[Fis91] D. H. Fisher. Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning*, 2(2):139–172, 1991.

[FISS98] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An Efficient Boosting Algorithm for Combining Preferences. In *Proceedings of the 15th International Conference on Machine Learning, ICML'98*, 1998.

[FITT98] A. Fujii, K. Inui, T. Tokunaga, and H. Tanaka. Selective Sampling for Example–based Word Sense Disambiguation. *Computational Linguistics*, 24(4):573–598, 1998.

[FK82] W. Francis and H. Kučera. *Frequency Analysis of English Usage*. Houghton Mifflin, Boston, 1982.

[FMPC98] S. Federici, S. Montemagni, V. Pirrelli, and N. Calzolari. Analogy–based Extraction of Lexical Knowledge from Corpora: The SPARKLE Experience. In *Proceedings of the 1st International Conference on Language Resources and Evaluation, LREC*, pages 75–82, Granada, Spain, May 1998.

[FP96] S. Federici and V. Pirrelli. *Analogy, Computation and Linguistic Theory.* D. Jones editor, New Methods in Language Processing. London: UCL Press, 1996.

[FPL91] D. Fisher, M. Pazzani, and P. Langley. *Concept Formation: Knowledge and Experience in Unsupervised Learning.* Morgan Kaufmann, 1991.

[FS95] Y. Freund and R. E. Schapire. A Decision–Theoretic Generalization of On–line Learning and an Application to Boosting. In *Proceedings of the 2nd European Conference on Computational Learning Theory, EuroCOLT'95*, Barcelona, Spain, 1995. Also in: Journal of Computer and System Sciences 55:1, 1997, 119–139.

[FS96] Y. Freund and R. E. Schapire. Experiments with a New Boosting Algorithm. In L. Saitta, editor, *Proceedings of the 13th International Conference on Machine Learning, ICML'96*, pages 148–156, San Francisco, CA, 1996. Morgan Kaufmann.

[GCY93] W. Gale, K. W. Church, and D. Yarowsky. A Method for Disambiguating Word Senses in a Large Corpus. *Computers and the Humanities*, 26:415–439, 1993.

[GLS87] R. Garside, G. Leech, and G. Sampson, editors. *The Computational Analysis of English: A Corpus-Based Approach.* London and New York: Longman, 1987.

[Gol89] D. E. Goldberg, editor. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison–Wesley, Reading, Mass., 1989.

[Gol95] A. R. Golding. A Bayesian–hybrid Method for Context–sensitive Spelling Correction. In *Proceedings of the 3rd Workshop on Very Large Corpora.* ACL, 1995.

[Goo53] I. J. Good. The Population Frequencies of Species and the Estimation of Population Parameters. *Biometrika*, 40, 1953.

[GR71] B. B. Greene and G. M. Rubin. Automatic Grammatical Tagging of English. Technical Report, Department of Linguistics, Brown University, 1971.

[GR98] A. R. Golding and D. Roth. A Winnow-based Approach to Spelling Correction. *Machine Learning, Special issue on Machine Learning and Natural Language Processing*, 1998. To appear. Also in Proceedings of the 13th International Conference on Machine Learning, ICML'96. http://l2r.cs.uiuc.edu/~danr.

[HA97] P. A. Heeman and J. F. Allen. Incorporating POS Tagging into Language Modeling. In *Proceedings of Eurospeech Conference*, 1997. cmp–lg/9705014.

[Hal96] H. van Halteren. *Comparison of Tagging Strategies, a Prelude to Democratic Tagging.* S. Hockney and N. Ide (eds.), Clarendon Press, 1996. Research in Humanities Computing 4. Selected papers for the ALLC/ACH Conference, Christ Church, Oxford.

[HAP89] R. C. Holte, L. E. Acker, and B. W. Porter. Concept Learning and the Problem of Small Disjuncts. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, pages 813–818. Morgan Kufmann, 1989.

[Har62] Z. Harris. *String Analysis of Language Structure.* Mouton and Co. The Hague, 1962.

[Har92] D. Harman. *Relevance Feedback and other Query Modification Techniques.* W. B. Frakes and R. Baeza–Yates (eds.), Information Retrieval: Data Structures and Algorithms. Prentice–Hall, Inc., Englewood Cliffs, NJ, 1992.

[Hay94] S. Haykin. *Neural Networks.* Macmillan College Publishing Company, Inc., 1994.

[HH97] Jan Hajič and Barbora Hladká. Probabilistic and Rule-Based Tagger of an Inflective Language - A Comparison. In *Proceedings of the 5th Conference on Applied Natural Language Processing, ANLP*, Washington DC, 1997. ACL.

[HH98] Jan Hajič and Barbora Hladká. Tagging Inflective Languages: Prediction of Morphological Categories for a Rich Structured Tagset. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 483–490, Montréal, Canada, August 1998.

[Hin89] D. Hindle. Acquiring Disambiguation Rules from Text. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, Vancouver, DC, 1989. ACL.

[HM97] M. Haruno and Y. Matsumoto. Mistake–driven Mixture of Hierarchical Tag Context Trees. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 230–237, Madrid, Spain, July 1997.

[HP94] M. Hearst and D. D. Palmer. Adaptive Sentence Boundary Disambiguation. In *Proceedings of the 3rd Conference on Applied Natural Language Processing, ANLP*, Stuttgart, Germany, October 1994. ACL.

[HSO98] M. Haruno, S. Shirai, and Y. Ooyama. Using Decision Trees to Construct a Practical Parser. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, Montréal, Canada, August 1998.

[HT98] T. Hastie and R. Tibshirani. Classification by Pairwise Coupling. In D.S. Touretzky, M.C. Mozer and M.E. Hesselmo, editor, *Advances in Neural Information Processing Systems 9, NIPS'97*. MIT Press, Cambridge, MA, 1998.

[HZD98] H. van Halteren, J. Zavrel, and W. Daelemans. Improving Data Driven Wordclass Tagging by System Combination. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 491–497, Montréal, Canada, August 1998.

[IV98] N. Ide and J. Véronis. Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art. *Computational Linguistics*, 24(1):1–40, 1998.

[Jär94] T. Järvinen. Annotating 200 Million Words: The Bank of English Project. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING*, Kyoto, Japan, 1994.

[Jel96] J. Jelonek. Generalization capability of homogeneous voting classifier based on partially replicated data. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 47–52, 1996. http://www.cs.fit.edu/~imlm.

[JJ94] M. I. Jordan and R. A. Jacobs. Hierarchical Mixtures of Experts and the EM Algorithm. *Neural Computation*, 6(2):181–214, 1994.

[JM99] D. Jurafsky and J.H. Martin. *Speech and Language Processing*. Prentice Hall, To appear in 1999. http://www.cs.colorado.edu/~martin/slp.html.

[JMR92] F. Jelineck, R. Mercer, and S. Roukos. *Principles of Lexical Language Modeling for Speech Recognition*. S. Furni and M. M. Sondhi (eds.), Advances in Speech Processing. Marcel Dekker, Inc., New York, 1992.

[Jon96] D. Jones. *Analogical Natural Language Processing*. London: UCL Press, 1996.

[JPCK96] S. Y. Jung, Y. C. Park, K. S. Choi, and Y. Kim. Markov Random Field Based English Part–of–speech Tagging System. In *Proceedings of 16th International Conference on Computational Linguistics, COLING*, Copenhagen, Denmark, August 1996.

[Kar95] L. Karttunen. The Replace Operator. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics, ACL*, Cambridge, MA, 1995. cmp–lg/9504032.

[Kat87] S. M. Katz. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35, 1987.

[Kay97] J. Kay. Comments on: A Comparative Analysis of Methods for Pruning Decision Trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):492–493, May 1997.

[KD95] E. B. Kong and T. G. Dietterich. Error–Correcting Output Coding Corrects Bias and Variance. In *Proceedings of the 12th International Conference on Machine Learning, ICML'95*, San Francisco, CA, 1995. Morgan Kaufmann.

[KE96] M. Koppel and S. P. Engelson. Integrating multiple classifiers by finding their areas of expertise. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 53–58, 1996. http://www.cs.fit.edu/~imlm.

[Kem94] A. Kempe. Probabilistic Tagging with Feature Structures. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING*, pages 161–165, Kyoto, Japan, August 1994. cmp–lg/9410027.

[Kem97] A. Kempe. Finite State Transducers Approximating Hidden Markov Models. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 460–467, Madrid, Spain, July 1997. cmp–lg/9707006.

[Kem98] A. Kempe. Look-Back and Look-Ahead in the Conversion of Hidden Markov Models into Finite State Transducers. In *Proceedings of the Joint Conference on New Methods in Language Processing and Computational Natural Language Learning (NeMLaP3 / CoNLL98)*, pages 29–37, Sydney, Australia, January 1998. cmp–lg/9802001.

[KHDM98] J. Kittler, M. Hatef, R. Duin, and J. Matas. On Combining Classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[KK94] R. Kaplan and M. Kay. Regular Models of Phonological Rule Systems. *Computational Linguistics*, 1994.

[KK96] A. Kempe and L. Karttunen. Parallel Replacement in Finite State Calculus. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING*, Copenhagen, Denmark, August 1996. cmp-lg/9707006.

[KK97] R. Kohavi and C. Kunz. Option Decision Trees with Majority Votes. In *Proceedings of the 14th International Conference on Machine Learning, ICML'97*, San Francisco, CA, 1997. Morgan Kaufmann.

[KN95] R. Kneser and H. Ney. Improved Backing-off for n-gram Language Modelling. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184, 1995.

[Koh96] R. Kohavi. Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pages 202–207, Menlo Park, CA, 1996. AAAI Press.

[Kon94] I. Kononenko. Estimating Attributes: Analysis and Extensions of Relief. In *Proceedings of the 6th European Conference on Machine Learning*, pages 171–182. Springer Verlag, 1994.

[Kos83] K. Koskenniemi. Two-level Morphology: A General Computation Model for Word-form Recognition and Production. Technical Report, Department of General Linguistics, University of Helsinki, 1983.

[KR92] K. Kira and L. A. Rendell. A Practical Approach to Feature Selection. In *Proceedings of the 9th International Conference on Machine Learning, ICML'92*, pages 249–256, San Francisco, CA, 1992. Morgan Kauffman.

[KR98] Y. Krymolowski and D. Roth. Incorporating Knowledge in Natural Language Learning: A Case Study. In *Proceedings of the COLING-ACL Workshop on the Usage of WordNet in Natural Language Processing Systems*, Montréal, Canada, August 1998. http://l2r.cs.uiuc.edu/~danr.

[Kro97] Robert Krovetz. Homonymy and Polysemy in Information Retrieval. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 72–79, Madrid, Spain, July 1997.

[KS63] S. Klein and R. Simmons. A Computational Approach to Grammatical Coding of English Words. *JACM*, 10:334–337, 1963.

[KS97] B. Krenn and C. Samuelsson. The Linguists' Guide to Statistics: Don't Panic. Technical report, Universität des Saarlandes, 1997. Postscript version of December 19, 1997 at URL: http://coli.uni-sb.de/~christer.

[KŠR95] I. Kononenko, E. Šimec, and M. Robnik-Šikonja. Overcoming the Myopia of Inductive Learning Algorithms with RELIEFF. *Applied Intelligence*, 10:39–55, 1995.

[Kup92] J. Kupiec. Robust Part-of-speech Tagging Using a Hidden Markov Model. *Computer Speech and Language*, 6, 1992.

[Kup93] J. Kupiec. Murax: A Robust Linguistic Approach for Question Answering Using an Online Encyclopedia. In *Proceedings of SIGIR '93*, pages 181–190, 1993.

[KVHA95] F. Karlsson, A. Voutilainen, J. Heikkilä, and A. Anttila, editors. *Constraint Grammar: A Language-Independent System for Parsing Unrestricted Text*. Mouton de Gruyter, Berlin and New York, 1995.

[KW94] J. Kivinen and M. K. Warmuth. Exponentiated Gradient versus Gradient Descent for Linear Predictors. Technical Report UCSC-CRL-94-16, Basking Center for Computer Engineering and Information Sciences. University of California, Santa Cruz, CA, 1994.

[Lan94a] M. M. Lankhorst. Automatic Word Categorization with Genetic Algorithms. Technical Report, Dept. of CS. University of Groningen, Groningen, The Netherlands, 1994.

[Lan94b] M. M. Lankhorst. Breeding Grammars. Grammatical Inference with a Genetic Algorithm. Technical Report, Dept. of CS. University of Groningen, Groningen, The Netherlands, 1994.

[LCG96] R. López de Mántaras, J. Cerquides, and P. Garcia. Comparing Information-theoretic Attribute Selection Measures: A Statistical Approach. Research Report 96-16, IIIA, 1996. To appear in Artificial Intelligence Communications.

[LCM98] C. Leacock, M. Chodorow, and G. A. Miller. Using Corpus Statistics and WordNet Relations for Sense Identification. *Computational Linguistics*, 24(1):147–166, 1998.

[LD95] C. Lyon and B. Dickerson. A Fast Partial Parse of Natural Language Sentences using a Connectionist Method. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, pages 149–156, Dublin, Ireland, 1995.

[Lee96] B. J. Lee. Applying Parallel Learning Models of Artificial Neural Networks to Letters Recognition from Phonemes. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 66–71, 1996. http://www.cs.fit.edu/~imlm.

[Leh91] W. Lehnert. *Symbolic/subsymbolic Sentence Analysis: Exploiting the Best of two Worlds*. J. Barnden and J. Pollack, editors, Advances in Connectionist and Neural Computation. Ablex Publishers, Norwood, NJ, 1991.

[Lew98] Mark Lewellen. Neural Network Recognition of Spelling Errors. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 1490–1493, Montréal, Canada, August 1998.

[LF92] G. Leech and S. Fligelstone. *Computers and Corpus Analysis*. C. S. Butler, editor, Computers and Written Texts. Blackwell, Oxford UK & Cambridge USA, 1992.

[LGA83] G. Leech, R. Garside, and E. Atwell. Automatic Grammatical Tagging of the LOB Corpus. *ICAME News*, 7:13–33, 1983.

[LGB94] G. Leech, R. Garside, and M. Bryant. CLAWS4: The Tagging of the British National Corpus. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING*, Kyoto, Japan, 1994.

[Lip89] R. P. Lippmann. Review of Neural Networks for Speech Recognition. *Neural Computation*, 1:1–38, 1989.

[Lit88] N. Littlestone. Learning Quickly when Irrelevant Attributes Abound. *Machine Learning*, 2:285–318, 1988.

[Lit94] D. J. Litman. Classifying Cue Phrases in Text and Speech Using Machine Learning. In *Proceedings of the 12th National Conference on Artificial Intelligence, AAAI*, pages 806–813, 1994. AAAI Press / MIT Press.

[Llo83] S. A. Lloyd. An optimization approach to relaxation labelling algorithms. *Image and Vision Computing*, 1(2):85–91, 1983.

[LM95a] J. Larrosa and P. Meseguer. An Optimization-based Heuristic for Maximal Constraint Satisfaction. In *Proceedings of the 1st International Conference on Principles and Practice of Constraint Programming, CP'95*, pages 103–120, Cassis, France, 1995.

[LM95b] J. Larrosa and P. Meseguer. Constraint Satisfaction as Global Optimization. In *Proceedings of 14th International Joint Conference on Artificial Intelligence, IJCAI '95*, pages 579–584, 1995.

[Lop91] R. Lopez de Mántaras. A Distance-Based Attribute Selection Measure for Decision Tree Induction. *Machine Learning*, 6(1):81–92, 1991.

[Lóp98] Joan López. *Un enfoque neuronal para la desambiguación del significado*. Phd. Thesis, Dep. Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya, 1998.

[LORP+96] S. Lehmann, S. Oepen, S. Regnier-Prost, K. Netter, V. Lux, J. Klein, K. Falkedal, F. Fouvry, D. Estival, E. Dauphin, H. Compagnion, J. Baur, L. Balkan, and D. Arnold. TSNLP - Test Suites for Natural Language Processing. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING*, Copenhagen, Denmark, August 1996.

[Los94] R. M. Losee. Learning Syntactic Rules and Tags with Genetic Algorithms for Information Retrieval and Filtering: An Empirical Basis for Grammatical Rules. *Information Processing & Management*, May 1994.

[LP97] R. López de Mántaras and E. Plaza. Case Based Reasoning: An Overview. *AI Communications*, 10:21–29, 1997.

[LR94] D. Lewis and M. Ringuette. A Comparison of two Learning Algorithms for Text Categorization. In *Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval*, 1994.

[LRR93] R. Lau, R. Rosenfeld, and S. Roukos. Adaptive Language Modelling Using the Maximum Entropy Principle. In *Proceedings of Human Language Technology Workshop, ARPA*, 1993.

[LRW96] W. Lezius, R. Rapp, and M. Wettler. A Morphology–System and Part–of–speech Tagger for German. In D. Gibbon, editor, *Natural Language Processing and Speech Technology. Results of the 3rd KONVENS Conference (Bielefeld)*, pages 369–378, Berlin, 1996. Mouton de Gruyter.

[LS93] M. Liberman and Y. Schabes. Statistical Methods in Natural Language Processing. Tutorial Notes, Computer and Information Science Department. University of Pennsylvania, 1993.

[LSCP96] D. Lewis, R. E. Schapire, J. P. Callan, and R. Papka. Training Algorithms for Linear Text Classifiers. In *Proceedings of the 19th International Conference on Research and Development in Information Retrieval, SIGIR*, pages 298–306, 1996.

[LSG95] S. Lawrence, F. Sandiway, and C. L. Giles. Natural Language Grammatical Inference: A Comparison of Recurrent Neural Networks and Machine Learning Methods. In *Proceedings of the IJCAI Workshop in New Approaches for NLP*, 1995. Also in S. Wermter, E. Riloff and G. Scheler (editors), Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing. Computer Notes in Artificial Intelligence 1040, Springer 1996.

[Lud97] Bernd Ludwig. A Contribution to the Question of Authenticity of Resus Using Part-of-Speech Tagging. In *Proceedings of the 21st Annual German Conference on Artificial Intelligence*, pages 231–242, Freiburg, Germany, 1997. Lecture Notes in Computer Science, Vol. 1303, Springer, 1997, ISBN 3-540-63493-2.

[LW94] N. Littlestone and M. K. Warmuth. The Weighted Majority Algorithm. *Information and Computation*, 108(2):212–261, 1994.

[Lyo94] C. Lyon. *The Representation of Natural Language to Enable Neural Networks to Detect Syntactic Structures.* Phd. Thesis, Computer Science Department, University of Hertfordshire, UK, 1994.

[Mac92] D. MacKay. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation*, 4(3):448–472, 1992.

[Mag95a] D. M. Magerman. Statistical Decision-Tree Models for Parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics.* ACL, 1995.

[Mag95b] David M. Magerman. Review of Charniak's *Statistical Language Learning. Computational Linguistics*, 21(1):103–111, 1995.

[Mag96] D. M. Magerman. Learning Grammatical Structure Using Statistical Decision–Trees. In *Proceedings of the 3rd International Colloquium on Grammatical Inference, ICGI*, pages 1–21, 1996. Springer-Verlag Lecture Notes Series in Artificial Intelligence 1147.

[Mat96] O. Matan. On Voting Ensembles of Classifiers. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 84–88, 1996. http://www.cs.fit.edu/~imlm.

[MB97] L. Mangu and E. Brill. Automatic Rule Acquisition for Spelling Correction. In *Proceedings of the 14th International Conference on Machine Learning, ICML'97*, pages 734–741, 1997. http://www.cs.jhu.edu/~brill/acadpubs.html.

[MB98] I. Mani and E. Bloedorn. Machine Learning of Generic and User-Focused Summarization. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference, AAAI/IAAI*, pages 821–826. AAAI Press / The MIT Press, 1998.

[MBB98] L. Mason, P. Bartlett, and J. Baxter. Direct Optimization of Margins Improves Geralization in Combined Classifiers. Technical Report, Department of Systems Engineering, Australian National University, 1998. http://www.research.att.com/~schapire/boost.html.

[MBF+91] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Five Papers on WordNet. *International Journal of Lexicography*, 1991.

[MBS98] R. Mitkov, L. Belguith, and M. Stys. Multilingual Robust Anaphora Resolution. In *Proceedings of the 3rd International Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 7–16, Granada, Spain, 1998.

[MC95] R. J. Mooney and M. E. Califf. Induction of First–order Decision Lists: Results on Learning the Past Tense of English Verbs. *Journal of Artificial Intelligence Research*, 3:1–24, 1995.

[MC96] R. J. Mooney and M. E. Califf. Learning the Past Tense of English Verbs Using Inductive Logic Programming. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[MCR92] R. Musick, J. Catlett, and S. Rusell. Decision Theoretic Subsampling for Induction on Large Databases. In P. E. Utgoff, editor, *Proceedings of the 10th International Conference on Machine Learning, ICML'92*, pages 212–219, San Francisco, CA, 1992. Morgan Kaufmann.

[MD97] D. D. Margineantu and T. G. Dietterich. Pruning Adaptive Boosting. In *Proceedings of the 14th International Conference on Machine Learning, ICML'97*, pages 211–218, San Francisco, CA, 1997.

[Mer94] B. Merialdo. Tagging English Text with a Probabilistic Model. *Computational Linguistics*, 20(2):155–171, 1994.

[Mer99] C. J. Merz. Using Correspondence Analysis to Combine Classifiers. *Machine Learning Journal. Special issue on IMLM for Improving and Scaling Machine Learning Algorithms*, 36(1&2), 1999.

[MI98] Qing Ma and Hitoshi Isahara. A Multi-Neuro Tagger Using Variable Lengths of Contexts. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 802–806, Montréal, Canada, August 1998.

[Mik96a] A. Mikheev. Learning Part-of-Speech Guessing Rules from Lexicon: Extension to Non-Concatenative Operations. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING'96*, Copenhagen, Denmark, 1996.

[Mik96b] A. Mikheev. Unsupervised Learning of Word-Category Guessing Rules. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, CA, 1996.

[Mik97] A. Mikheev. Automatic Rule Induction for Unknown–Word Guessing. *Computational Linguistics*, 23(3):405–424, 1997.

[Min89] J. Mingers. An Empirical Comparison of Selection Measures for Decision–Tree Induction. *Machine Learning*, 3:319–342, 1989.

[Mit98] R. Mitkov. Robust Pronoun Resolution with Limited Knowledge. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 869–875, Montréal, Canada, 1998.

[ML95] J. F. McCarthy and W. G. Lehnert. Using Decision Trees for Coreference Resolution. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI*, pages 1050–1055, 1995.

[ML96] N. M. Marques and G. C. Lopes. A Neural Network Approach to Part–of–speech Tagging. In *Proceedings of the 2nd Workshop on Computational Processing of Written and Spoken Portuguese*, pages 21–22, Brazil, 1996.

[MLC98] N. M. Marques, G. P. Lopes, and C. A. Coelho. Learning Verbal Transitivity Using LogLinear Models. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European Conference on Machine Learning, ECML'98*, pages 19–24, Chemnitz, Germany, 1998. Springer.

[MLTB93] G. A. Miller, C. Leacock, R. Tengi, and R. T. Bunker. A Semantic Concordance. In *Proceedings of the ARPA Workshop on Human Language Technology*, 1993.

[MM96] C. J. Merz and P. M. Murphy. UCI Repository of Machine Learning Databases. Technical report, Irvine, CA, Department of Information and Computer Sciences, University of California, 1996. http://www.ics.uci.edu/~mlearn/MLRepository.html.

[MM97] E. Mayoraz and M. Moreira. On the Decomposition of Polychotomies into Dichotomies. In D. H. Fisher, editor, *Proceedings of the 14th International Conference*

on Machine Learning, ICML'97, pages 219–226, San Francisco, CA, 1997. Morgan Kaufmann.

[MM98] M. Moreira and E. Mayoraz. Improved Pairwise Coupling Classification with Correcting Classifiers. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European Conference on Machine Learning, ECML'98*, pages 160–171, Chemnitz, Germany, 1998. Springer.

[MMS93] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2), 1993.

[MMW93] T. Matsukawa, S. Miller, and R. Weischedel. Example-Based Correction of Word Segmentation and Part of Speech Labelling. In *Proceedings of ARPA*, 1993.

[MO97] R. Maclin and D. Opitz. An Empirical Evaluation of Bagging and Boosting. In *Proceedings of the 14th National Conference on Artificial Intelligence*, pages 546–551. AAAI Press, 1997.

[Moh97] M. Mohri. Finite–State Transducers in Language and Speech Processing. *Computational Linguistics*, 23(2):269–312, 1997.

[Moo95] R. J. Mooney. Encouraging Experimental Results on Learning CNF. *Machine Learning*, 19(1):79–92, 1995.

[Moo96] R. J. Mooney. Comparative Experiments on Disambiguating Word Senses: An Illustration of the Role of Bias in Machine Learning. In *Proceedings of the 1st Conference on Empirical Methods in Natural Language Processing, EMNLP*, 1996.

[Moo97] R. J. Mooney. *Inductive Logic Programming for Natural Language Processing.* S. Muggleton (Ed.), Inductive Logic Programming: Selected Papers from the 6th International Workshop. Springer Verlag, Berlin, 1997.

[MP97] Lluís Màrquez and Lluís Padró. A Flexible POS Tagger Using an Automatically Acquired Language Model. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 238–245, Madrid, Spain, July 1997.

[MP98] P.W. Munro and B. Parmanto. Competition among Networks Improves Committee Performance. In D.S. Touretzky, M.C. Mozer and M.E. Hesselmo, editor, *Advances in Neural Information Processing Systems 9, NIPS'97*. MIT Press, Cambridge, MA, 1998.

[MP99] C. J. Merz and M. J. Pazzani. A Principal Components Approach to Combining Regression Estimates. *Machine Learning Journal. Special issue on IMLM for Improving and Scaling Machine Learning Algorithms*, 36(1&2), 1999.

[MPR98] Lluís Màrquez, Lluís Padró, and Horacio Rodríguez. Improving Tagging Accuracy by Voting Taggers. In *Proceedings of the 2nd Conference on Natural Language Processing & Industrial Applications, NLP+IA/TAL+AI*, pages 149–155, New Brunswick, Canada, August 1998.

[MR95] Lluís Màrquez and Horacio Rodríguez. Towards Learning a Constraint Grammar from Annotated Corpora Using Decision Trees. Working Paper #21, ESPRIT BRA–7315 Acquilex II, 1995.

[MR97] Lluís Màrquez and Horacio Rodríguez. Automatically Acquiring a Language Model for POS Tagging Using Decision Trees. In *Proceedings of the Second Conference on Recent Advances in Natural Language Processing, RANLP*, pages 27–34, Tzigov Chark, Bulgaria, September 1997.

[MR98] Lluís Màrquez and Horacio Rodríguez. Part-of-Speech Tagging Using Decision Trees. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European Conference on Machine Learning, ECML'98*, pages 25–36, Chemnitz, Germany, 1998. Springer.

[MS99] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, MA, 1999. In Press. Draft available at: http://www.sultry.arts.usyd.edu.au/fsnlp.

[MSW91] M. Meteer, R. Schwartz, and R. Weischedel. Empirical Studies in Part of Speech Labelling. In *Proceedings of the DARPA Speech and Natural Language Workshop.* Morgan Kaufmann, 1991.

[MT94] I. Moreno-Torres. A Morphological Disambiguation Tool (MDS). An Application to Spanish. Working Paper #24, ESPRIT BRA–7315 Acquilex II, 1994.

[Mur95] S. K. Murthy. *On Growing Better Decision Trees from Data.* Phd. Thesis, Johns Hopkins University, Baltimore, Maryland, 1995.

[MW96] T. McEnery and A. Wilson. *Corpus Linguistics.* Edinburgh University Press, Edinburgh, UK, 1996. Complementary course available at: http://www.ling.lancs.ac.uk/monkey/ihe/linguistics/contents.htm.

[NB86] T. Niblett and I. Bratko. Learning Decision Rules in Noisy Domains. In *Proceedings of Expert Systems '86.* Cambridge University Press, 1986.

[Nea93] R. Neal. Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report CRG-TR-93-1, Department of Computer Science, University of Toronto, Toronto, CA, 1993.

[Ng97] H. T. Ng. Exemplar-Base Word Sense Disambiguation: Some Recent Improvements. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, 1997.

[NL96] H. T. Ng and H. B. Lee. Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-based Approach. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics.* ACL, 1996.

[NMKS90] M. Nakamura, K. Maruyama, T. Kawabata, and K. Shikano. Neural Network Approach to Word Category Prediction for English Texts. In *Proceedings of 13th International Conference on Computational Linguistics, COLING*, pages 213–218, Helsinki, Finland, 1990. Karlgren, H (ed.) COLING 90.

[NMTM98] K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Learning to Classify Text from Labeled and Unlabeled Documents. In *Proceedings of the 15th National Conference on Artificial Intelligence, AAAI-98*, Madison, Wisconsin, 1998.

[OK94] K. Oflazer and I. Kuruöz. Tagging and Morphological Disambiguation of Turkish Text. In *Proceedings of the 4th Conference on Applied Natural Language Processing, ANLP.* ACL, 1994.

[Oos91] N. Oostdijk. *Corpus Linguistic and the Automatic Analysis of English.* Rodopi, Amsterdam, 1991.

[Ort96] J. Ortega. Exploiting multiple existing models and learning algorithms. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 101–106, 1996. http://www.cs.fit.edu/~imlm.

[OS96] D. W. Opitz and J. W. Shavlik. Generating Accurate and Diverse Members of a Neural-Network Ensemble. In D.S. Touretzky, M.C. Mozer and M.E. Hesselmo, editor, *Advances in Neural Information Processing Systems 8, NIPS'95*, pages 535–541. MIT Press, Cambridge, MA, 1996.

[OT96] K. Oflazer and G. Tür. Combining Hand-craftet Rules and Unsupervised Learning in Constraint-Based Morphological Disambiguation. In *Proceedings of the ACL-SIGDAT Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, 1996. cmp-lg/9604001.

[Pad96] Lluís Padró. POS Tagging Using Relaxation Labelling. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING*, pages 877–882, Copenhagen, Denmark, August 1996.

[Pad98] Lluís Padró. *A Hybrid Environment for Syntax-Semantic Tagging.* Phd. Thesis, Dep. Llenguatges i Sistemes Informàtics. Universitat Politècnica de Catalunya, February 1998. http://www.lsi.upc.es/~padro.

[PKPD95] D. Price, S. Knerr, L. Personnaz, and G. Dreyfus. Pairwise Neural Network Classifiers with Probabilistic Outputs. In D.S. Touretzky G. Tesauro and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7, NIPS'94*, pages 1109–1116. MIT Press, Cambridge, MA, 1995.

[PM94] M. Pelillo and A. Maffione. Using Simulated Annealing to Train Relaxation Labelling Processes. In *Proceedings of ICANN '94*, 1994.

[PM98] Lluís Padró and Lluís Màrquez. On the Evaluation and Comparison of Taggers: the Effect of Noise in Testing Corpora. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 997–1002, Montréal, Canada, August 1998.

[PMD96] B. Parmanto, P.W. Munro, and H.R. Doyle. Improving Committee Diagnosis with Resampling Techniques. In M.C. Mozer D.S. Touretzky and M.E. Hesselmo, editors,

Advances in Neural Information Processing Systems 8, NIPS'95, volume 8, pages 882–888. MIT Press, Cambridge, MA, 1996.

[Pow97] D. M. Powers. Machine Learning of Natural Language. In *Joint ACL/EACL Tutorial Program*, Madrid, Spain, 1997.

[PP98] F. Pla and N. Prieto. Using Grammatical Inference Methods for Automatic Part–of–speech Tagging. In *Proceedings of 1st International Conference on Language Resources and Evaluation, LREC*, Granada, Spain, 1998.

[PPL95] S. Della Pietra, V. Della Pietra, and John Lafferty. Inducing Features of Random Fields. Technical Report CMU–CS95–144, School of Computer Science, Carnegie–Mellon University, 1995.

[PR94] M. Pelillo and M. Refice. Learning Compatibility Coefficients for Relaxation Labeling Processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9), 1994.

[PRS94] F. Pereira, M. Riley, and R. W. Sproat. Weighted Rational Transductions and their Application to Human Language Processing. In *Proceedings of the ARPA Human Language Technology Workshop*, pages 262–267, San Francisco, CA, 1994. Morgan Kaufmann.

[PS92] F. Pereira and Y. Schabes. Inside-Outside Re-estimation from Partially Bracketed Corpora. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 128–135, 1992.

[PST95] F. Pereira, Y. Singer, and N. Tishby. Beyond Word N–Grams. In *David Yarowsky and Kenneth Church, editors, Proceedings of the Third Workshop on Very Large Corpora*, pages 95–106, Massachusetts Institute of Technology, 1995. ACL. cmp–lg/9607016.

[Qui79] J. R. Quinlan. *Discovering Rules from Large Collections of Examples.* Edimburgh University Press, 1979. ???

[Qui86] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1:81–106, 1986.

[Qui87] J. R. Quinlan. Simplifying Decision Trees. *International Journal on Man–Machine Studies*, 27:221–234, 1987.

[Qui90] J. R. Quinlan. Learning Logical Definitions from Relations. *Machine Learning*, 5(3):239–266, 1990.

[Qui93] J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1993.

[Qui96a] J. R. Quinlan. Bagging, Boosting, and C4.5. In *Proceedings of the 13th National Conference on Artificial Intelligence*, pages 725–730, Cambridge, MA, 1996. AAAI Press/MIT Press.

[Qui96b] J. R. Quinlan. Boosting First–Order Learning. In *Proceedings of the ALT'96 conference*, 1996. http://www.cse.unsw.EDU.AU/~quinlan.

[Qui98] J. R. Quinlan. MiniBoosting Decision Trees. *Journal of Artificial Intelligence Research*, 1998. To appear. Available at http://www.cse.unsw.EDU.AU/~quinlan.

[RA98] F. Ricci and D. W. Aha. Extending Local Learners with Error–Correcting Output Codes. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European Conference on Machine Learning, ECML'98*, pages 280–291, Chemnitz, Germany, 1998. Springer.

[RAA97] German Rigau, Jordi Atserias, and Eneko Agirre. Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, pages 48–55, Madrid, Spain, July 1997.

[Rab90] L. R. Rabiner. *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition.* Readings in Speech Recognition (eds. A. Waibel, K. F. Lee). Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1990.

[Rat96] A. Ratnaparkhi. A Maximum Entropy Part–of–speech Tagger. In *Proceedings of the 1st Conference on Empirical Methods in Natural Language Processing, EMNLP*, 1996.

[Rat97a] A. Ratnaparkhi. A Linear Observed Time Statistical Parser Based on Maximum Entropy Models. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, Brown University, Providence, RI, 1997.

[Rat97b] A. Ratnaparkhi. A Simple Introduction to Maximum Entropy Models for Natural Language Processing. Technical Report 97-08, Institute for Research in Cognitive Science, University of Pennsylvania, 1997.

[Rat98] A Ratnaparkhi. *Maximum Entropy Models for Natural Language Ambiguity Resolution.* Phd. Thesis, University of Pennsylvania, 1998. http://www.cis.upenn.edu/~adwait.

[RHZ76] R. Rosenfeld, R. Hummel, and S. Zucker. Scene labelling by relaxation operations. *IEEE Transactions on Systems, Man and Cybernetics*, 6(6):420–433, 1976.

[RI96] Y. Raviv and N. Intrator. Bootstrapping with Noise: An Effective Regularization Technique. *Connection Science*, 8(3–4):355–372, 1996.

[Ris97] E. S. Ristad. Maximum Entropy Modeling for Natural Language. In *Joint ACL/EACL Tutorial Program*, Madrid, Spain, 1997.

[Riv87] R. L. Rivest. Learning Decision Lists. *Machine Learning*, 2:229–246, 1987.

[RL94] E. Riloff and W. Lehnert. Information Extraction as a Basis for High-precision Text Classification. *ACM Transactions on Information Systems*, 12(3):296–333, 1994.

[RLS81] J. Richards, D. Landgrebe, and P. Swain. On the accuracy of pixel relaxation labelling. *IEEE Transactions on Systems, Man and Cybernetics*, 11(4):303–309, 1981.

[RM94] L. Ramshaw and M. Marcus. Exploring the Statistical Derivation of Transformational Rule Sequences for Part-of-Speech Tagging. In *Proceedings of the ACL Workshop on Combining Symbolic and Statistical Approaches to Language*, New Mexico State University, 1994.

[Roc71] J. Rocchio. *Relevance Feedback in Information Retrieval.* G. Salton (editor), The SMART Retrieval System: Experiments in Automatic Document Processing. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971.

[Ros94] R. Rosenfeld. *Adaptive Statistical Language Modelling: A Maximum Entropy Approach.* Phd. Thesis, School of Computer Science, Carnegie Mellon University, 1994.

[Ros96a] B. E. Rosen. Ensemble Learning Using Decorrelated Neural Networks. *Connection Science*, 8(3–4):420–433, 1996.

[Ros96b] R. Rosenfeld. A Maximum Entropy Approach to Adaptive Statistical Language Modeling. *Computer Speech and Language*, 10:187–228, 1996.

[Rot98] D. Roth. Learning to Resolve Natural Language Ambiguities: A Unified Approach. In *Proceedings of the National Conference on Artificial Intelligence, AAAI '98*, July 1998. http://l2r.cs.uiuc.edu/~danr.

[RP93] X. Ren and F. Perrault. The Typology of Unknown Words: An Experimental Study of Two Corpora. In *Proceedings of 14th International Conference on Computational Linguistics, COLING'92*, Nantes, France, 1993.

[RR97] C. Reynar and A. Ratnaparkhi. A Maximum Entropy Approach to Identifying Sentence Boundaries. In *Proceedings of the 5th Conference on Applied Natural Language Processing, ANLP*, Washington DC, 1997. ACL.

[RRR94] A. Ratnaparkhi, J. Reynar, and S. Roukos. A Maximum Entropy Model for Prepositional Phrase Attachment. In *Proceedings of the ARPA Human Language Technology Workshop*, pages 250–255, 1994.

[RS95] E. Roche and Y. Schabes. Deterministic Part-of-speech Tagging with Finite State Transducers. *Computational Linguistics*, 21(2):227–253, 1995.

[RT96] E. Ristad and R. G. Thomas. Nonuniform Markov Models. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Munich, Germany, 1996.

[RT97] E. Ristad and R. G. Thomas. Hierarchical Non-emitting Markov Models. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, Madrid, Spain, 1997.

[RZ98] D. Roth and D. Zelenko. Part of Speech Tagging Using a Network of Linear Separators. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 1136–1142, Montréal, Canada, 1998. http://l2r.cs.uiuc.edu/~danr.

[SA94] C. Souter and A. Atwell. Using Parsed Corpora: A Review of Current Practice. In N. Oostdijk and P. de Haan, editors, *Corpus-Based Research into Language.* Rodopi, Amsterdam, 1994.

[Sam93] C. Samuelsson. Morphological Tagging Based Entirely on Bayesian Inference. In *Proceedings of the 9th Nordic Conference of Computational Linguistics*, Stockholm, Sweden, 1993.

[Sam95] C. Samuelsson. A Novel Framework for Reductionistic Statistical Parsing. In *Proceedings of the 4th International Workshop on Parsing Technologies*, pages 208–215, Prague/Karlovy Vary, Czech Republic, 1995.

[Sam96] C. Samuelsson. Handling Sparse Data by Successive Abstraction. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING*, Copenhagen, Denmark, August 1996. http://www.coli.uni-sb.de/~christer.

[Sam97] C. Samuelsson. Extending N-gram Tagging to Word Graphs. In *Proceedings of the Second Conference on Recent Advances in Natural Language Processing, RANLP*, pages 21–26, Tzigov Chark, Bulgaria, September 1997. http://www.coli.uni-sb.de/~christer.

[Sam98] Ken Samuel. Lazy Transformation-Based Learning. In *Proceedings of the 11th International Florida Artificial Intelligence Research Symposium Conference*, pages 235–239, 1998. cmp-lg/9806003.

[SB98a] H. Schwenk and Y. Bengio. Training Methods for Adaptive Boosting of Neural Networks for Character Recognition. *Advances in Neural Information Processing Systems*, 10, 1998.

[SB98b] W. Skut and T. Brants. A Maximum-Entropy Partial Parser for Unrestricted Text. In *Proceedings of the 6th Workshop on Very Large Corpora*, Montréal, Canada, August 1998. cmp-lg/9807006.

[SB98c] W. Skut and T. Brants. Chunk Tagger – Statistical Recognition of Noun Phrases. In *Proceedings of the ESSLLI'98 Workshop on automated Acquisition of Syntax and Parsing*, University of Saarbrücken, 1998. cmp-lg/9807007.

[Sch93] H. Schütze. Part-of-speech Induction from Scratch. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 251–258, Columbus, OH, 1993. ACL.

[Sch94a] H. Schmid. Part-of-speech Tagging with Neural Networks. In *Proceedings of the 15th International Conference on Computational Linguistics, COLING*, pages 172–176, Kyoto, Japan, 1994.

[Sch94b] H. Schmid. Probabilistic Part-of-speech Tagging Using Decision Trees. In *Proceedings of the Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK, 1994.

[Sch95a] H. Schmid. Improvements in Part-of-speech Tagging with an Application to German. In *Proceedings of the EACL SIGDAT Workshop*, Dublin, Ireland, August 1995.

[Sch95b] H. Schütze. Distributional Part-of-speech Tagging. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, Dublin, Ireland, 1995.

[Sch97] R. E. Schapire. Using Output Codes to Boost Multiclass Learning Problems. In D. H. Fisher, editor, *Proceedings of the 14th International Conference on Machine Learning, ICML'96*, San Francisco, CA, 1997. Morgan Kaufmann.

[Sch98] H. Schütze. Automatic Word Sense Discrimination. *Computational Linguistics*, 24(1):97–124, 1998.

[Sch99] R. E. Schapire. Theoretical Views of Boosting. In *Proceedings of the 4th European Conference on Computational Learning Theory, EuroCOLT'99*, 1999.

[SCVS98a] Ken Samuel, Sandra Carberry, and K. Vijay-Shanker. An Investigation of Transformation-Based Learning in Discourse. In *Proceedings of the 15th International Conference on Machine Learning, ICML'98*, 1998. cmp-lg/9806006.

[SCVS98b] Ken Samuel, Sandra Carberry, and K. Vijay-Shanker. Dialogue Act Tagging with Transformation-Based Learning. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 1150–1156, Montréal, Canada, 1998. cmp-lg/9806006.

[SD89] T. J. Santner and D. E. Duffy. *The Statistical Analysis of Discrete Data*. Springer-Verlag, New York, 1989.

[SD94] S. Sestito and T. S. Dillon. *Automated Knowledge Acquisition*. T. S. Dillon (ed.), Series in Computer Systems Science and Engineering. Prentice Hall, New York/London, 1994.

[SFBL97] R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the Margin: A New Explanation for the Efectiveness of Voting Methods. In D. H. Fisher, editor, *Proceedings of the 14th International Conference on Machine Learning, ICML'97*, pages 322–330, San Francisco, CA, 1997. Morgan Kaufmann.

[Sie97] E. V. Siegel. Learning Methods for Combining Linguistic Indicators to Classify Verb. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, Brown University, Providence, RI, 1997. cmp-lg/9707015.

[Ska96] D. B. Skalak. The Sources of Increased Accuracy for Two Proposed Boosting Algorithms. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 120–125, 1996. http://www.cs.fit.edu/~imlm.

[SLL98] J. M. Sopena, A. Lloberas, and J. López. A Connectionist Approach to Prepositional Phrase Attachment. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 1233–1237, Montréal, Canada, August 1998.

[SN95] F. Sánchez and A. F. Nieto. Desarrollo de un etiquetador morfosintáctico para el español. In *Proceedings of the 11th Congreso de la Sociedad Española para el Procesamiento del Lenguage Natural, SEPLN*, Universidad de Deusto, Bilbo, Spain, 1995. cmp-lg/9505035.

[Sou40] R. Southwell. *Relaxation Methods in Engineering Science*. Clarendon, 1940.

[SP94] M. Srinivas and L. M. Patnaik. Genetic Algorithms: A survey. *Computer*, 27(6):17–26, 1994.

[SP95] H. Schütze and O. Pedersen. Information Retrieval based on Word Senses. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, NV, 1995.

[SP97] L. Saul and F. Pereira. Aggregate and Mixed-order Markov Models for Statistical Language Processing. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP*, 1997.

[SR87] T. J. Sejnowski and C. S. Rosenberg. Parallel Networks that Learn to Pronounce. *Complex Systems*, 1:145–168, 1987.

[SS94] H. Schütze and Y. Singer. Part–of–speech Tagging Using a Variable Memory Markov Model. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, Las Cruces, New Mexico, 1994. ACL.

[SS98a] R. E. Schapire and Y. Singer. BoosTexter: A system for multiclass multi-label text categorization. Unpublished. Postscript version available at URL: http://www.research.att.com/~schapire, AT&T Labs, 1998.

[SS98b] R. E. Schapire and Y. Singer. Improved Boosting Algorithms Using Confidence–rated Predictions. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, 1998.

[SSGC97] F. Segond, A. Schiller, G. Grefenstette, and J-P. Chanod. An Experiment in Semantic Tagging using Hidden Markov Model Tagging. In *Proceedings of the Joint ACL/EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources*, pages 78–81, Madrid, Spain, 1997.

[SSS98] R. E. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In *In Proceedings of the 21st Annual International Conference on Research and Development in Information Retrieval, SIGIR '98*, 1998.

[STV96] C. Samuelsson, P. Tapanainen, and A. Voutilainen. Inducing Constraint Grammars. In *Proceedings of the 3rd International Colloquium on Grammatical Inference, ICGI'96*, pages 146–155, Montpellier, France, 1996. cmp–lg/9607002.

[SV97] C. Samuelsson and A. Voutilainen. Comparing a Linguistic and a Stochastic Tagger. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 246–253, Madrid, Spain, July 1997. http://www.coli.uni-sb.de/~christer.

[SW95] T. C. Smith and I. H. Witten. Learning Language Using Genetic Algorithms. In *Proceedings of the IJCAI Workshop in New Approaches for NLP*, 1995. Also in S.

Wermter, E. Riloff and G. Scheler (editors), Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing. Computer Notes in Artificial Intelligence 1040, Springer 1996.

[SW99] P. Smyth and D. Wolpert. Linearly Combining Density Estimators via Stacking. *Machine Learning Journal. Special issue on IMLM for Improving and Scaling Machine Learning Algorithms*, 36(1&2), 1999.

[SY92] R. F. Simmons and Y. Yu. The Acquisition and Use of Context–dependent Grammars for English. *Computational Linguistics*, 18(4):391–418, 1992.

[Tan96] H. Tanaka. Decision Tree Learning Algorithm with Structured Attributes: Application to Verbal Case Frame Acquisition. In *Proceedings of the 16th International Conference on Computational Linguistics, COLING*, pages 943–948, Copenhagen, Denmark, August 1996.

[Tap96] P. Tapanainen. The Constraint Grammar Parser CG–2. Technical Report n.27, Department of General Linguistics, University of Helsinki, 1996.

[TG96a] K. Tumer and J. Ghosh. Classifier Combining: Analytical Results and Implications. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 126–132, 1996. http://www.cs.fit.edu/~imlm.

[TG96b] K. Tumer and J. Ghosh. Error Correlation and Error Reduction in Ensemble Classifiers. *Connection Science. Special issue on combining artificial neural networks: ensemble approaches*, 8(3 and 4):385–404, 1996.

[Tin94] K. M. Ting. The Problem of Small Disjuncts: its remedy in Decision Trees. In *Proceedings of the 10th Canadian Conference on Artificial Intelligence*, pages 91–97, 1994.

[Tin97] K. M. Ting. Decision Combination Based on the Characterisation of Predictive Accuracy. *International Journal of Intelligent Data Analysis*, 1(3), 1997.

[TMT97] C. A. Thompson, R. J. Mooney, and L. R. Tang. Learning to Parse Natural Language Database Queries into Logical Form. In *Proceedings of the ML-97 Workshop on Automata Induction, Grammatical Inference, and Language Acquisition*, 1997.

[TO98] Gökhan Tür and Kemal Oflazer. Tagging English by Path Voting Constraints. In *Proceedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL*, pages 1277–1281, Montréal, Canada, August 1998.

[Tor89] C. Torras. Relaxation and Neural Learning: Points of Convergence and Divergence. *Journal of Parallel and Distributed Computing*, 6:217–244, 1989.

[TR96] E. Tzoukermann and D. R. Radev. Using Word Class for Part–of–speech Disambiguation. In *Proceedings of the 4th Workshop on Very Large Corpora, COLING*, pages 1–13, Copenhagen, Denmark, 1996.

[TR97] E. Tzoukermann and D. R. Radev. Use of Weighted Finite State Transducers in Part–of–speech Tagging. Technical report, Dept. of Computer Science. Columbia University, 1997. cmp-lg/9710001.

[TRG95] E. Tzoukermann, D. R. Radev, and W. A. Gale. Combining Linguistic Knowledge and Statistical Learning in French Part–of–speech Tagging. In *Proceedings of the EACL SIGDAT Workshop*, Dublin, Ireland, 1995.

[TRG97] E. Tzoukermann, D. R. Radev, and W. A. Gale. *Tagging French Without Lexical Probabilities*. Natural Language Processing using Very Large Corpora (eds. S. Amstrong, K. Church, P. Isabelle, E. Tzoukermann and D. Yarowsky). Kluwer, 1997. To appear.

[TSHS96] S. Teufel, H. Schmid, U. Heid, and A. Schiller. EAGLES Validation (WP4) Task on Tagset and Tagger Interaction. Technical Report n.27, IMS, Universität Stuttgart, 1996.

[TV98] G. Towell and E. M. Voorhees. Disambiguating Highly Ambiguous Words. *Computational Linguistics*, 24(1):125–146, 1998.

[TW97] K. M. Ting and I. H. Witten. Stacked Generalization: When Does It Work? In *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, pages 866–871, 1997.

[TZ98] K. M. Ting and Z. Zheng. Boosting Trees for Cost–Sensitive Classifications. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European*

Conference on Machine Learning, ECML'98, pages 190–195, Chemnitz, Germany, 1998. Springer.

[Van96] K. Vanhoof. Combining Rules and Cases. In P. Chan, editor, *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pages 139–143, 1996. http://www.cs.fit.edu/~imlm.

[Vee98] J. Veenstra. Fast NP Chunking Using Memory-based Learning Techniques. In *Proceedings of Benelearn*, Wageningen, the Netherlands, 1998. To appear.

[VJ95] A. Voutilainen and T. Järvinen. Specifying a Shallow Grammatical Representation for Parsing Purposes. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, Dublin, Ireland, 1995.

[Vou94] Atro Voutilainen. *Three Studies of Grammar-Based Surface Parsing on Unrestricted English Text.* Phd. Thesis, Department of General Linguistics. University of Helsinki, 1994.

[Vou95] Atro Voutilainen. A Syntax-Based Part-of-speech Analyzer. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics, EACL*, Dublin, Ireland, 1995.

[VP97] Atro Voutilainen and Lluís Padró. Developing a Hybrid NP Parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing, ANLP*, pages 80–87, Washington DC, 1997. ACL.

[VS98] M. Volk and G. Schneider. Comparing a Statistical and a Rule-Based Tagger for German. In *Proceedings of the 4th Conference on Natural Language Processing, KONVENS'98*, pages 125–137, Bonn, Germany, 1998.

[Wal75] D. Waltz. *Understanding line drawings of scenes with shadows: Psychology of Computer Vision.* McGraw-Hill, New York, 1975.

[Wau95] O. Wauschkuhn. The Influence of Tagging on the Results of Partial Parsing in German Corpora. In *Proceedings of the 4th International Workshop on Parsing Technologies (IWPT'95)*, Prague/Karlovy Vary, Czech Republic, 1995.

[WBM95] K. Wnek, E. Bloedorn, and R. Michalski. Selective Inductive Learning Method AQ15C: The Method and User's Guide. Laboratory Report ML95-4, Machine Learning and Inference Laboratory, George Mason University, Fairfax, Virginia, 1995.

[WL94] A. P. White and W. Z. Liu. Bias in Information-Based Measures in Decision-tree Induction. *Machine Learning*, 15:321–329, 1994.

[Wol92] D. H. Wolpert. Stacked Generalization. *Neural Networks, Pergamon Press*, 5:241–259, 1992.

[WPW95] E. J. Wiener, J. Pedersen, and A. Weigend. A Neural Network Approach to Topic Spotting. In *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, 1995.

[WRS96] S. Wermter, E. Riloff, and G. Scheler (editors). *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing.* Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[WS85] B. Widrow and S. D. Stearns. *Adaptive Signal Processing.* Prentice-Hall, Englewood Cliffs, NJ, 1985.

[WS97] Y. Wilks and M. Stevenson. Combining Independent Knowledge Sources for Word Sense Disambiguation. In *Proceedings of the Second Conference on Recent Advances in Natural Language Processing, RANLP*, pages 1–7, Tzigov Chark, Bulgaria, September 1997.

[WSG96] Y. Wilks, B. Slator, and L. Guthrie. *Electric Words: Dictionaries, Computers and Meanings.* The MIT Press, Cambridge, MA, 1996.

[WSP+93] R. Weischedel, R. Schwartz, J. Palmucci, M. Meteer, and L. Ramshaw. Coping with Ambiguity and Unknown Words through Probabilistic Models. *Computational Linguistics*, 19(2):260–269, 1993.

[WW96] V. Weber and S. Wermter. Using Hybrid Connectionist Learning for Speech/Language Analysis. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[Yan93] J. J. Yang. *Use of Genetic Algorithms for Query Improvement in Information Retrieval Based on a Vector Space Model.* Phd. Thesis, University of Pittsburgh, Pittsburgh, PA, 1993.

[Yar93] D. Yarowsky. One Sense per Collocation. In *DARPA Workshop on Human Language Technology*, Princeton, 1993.

[Yar94a] D. Yarowsky. Decision Lists for Lexical Ambiguity Resolution: Application to Accent Restoration in Spanish and French. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, pages 88–95, Las Cruces, NM, 1994. ACL.

[Yar94b] D. Yarowsky. Homograph Disambiguation in Speech Synthesis. In *Proceedings of 2nd ESCA/IEEE Workshop on Speech Synthesis*, New Paltz, NY, 1994.

[YB97] S. Young and G. Bloothooft (editors). *Corpus-based Methods in Language and Speech Processing*. An ELSNET book. Kluwer Academic Publishers, Dordrecht, 1997.

[YC94] Y. Yang and C. G. Chute. An Example-based Mapping Method for Text Classification and Retrieval. *ACM Transactions on Information Systems*, 12(3), 1994.

[YPM96] T. Yamazaki, M. J. Pazzani, and C. Merz. Acquiring and Updating Hierarchical Knowledge for Machine Translation based on a Clustering Technique. In E. Riloff S. Wermter and G. Scheler, editors, *Connectionist, Statistical an Symbolic Approaches to Learning for Natural Language Processing*, Lecture Notes in Artificial Intelligence, 1040. Springer, 1996.

[ZD91] X. Zhou and T. S. Dillon. A Statistical-Heuristic Feature Selection Criterion for Decision Tree Induction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):834–841, 1991.

[ZD97] J. Zavrel and W. Daelemans. Memory-Based Learning: Using Similarity for Smoothing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics. Joint ACL/EACL*, Madrid, Spain, July 1997.

[ZDV97] J. Zavrel, W. Daelemans, and J. Veenstra. Resolving PP attachment Ambiguities with Memory-Based Learning. In *Proceedings of the Conference on Computational Natural Language Learning, CoNLL97*, pages 136–144, Madrid, Spain, 1997.

[Zhe98] Z. Zheng. Naive Bayesian Classifier Committees. In C. Nédellec and C. Rouveirol, editor, *LNAI 1398: Proceedings of the 10th European Conference on Machine Learning, ECML'98*, pages 196–207, Chemnitz, Germany, 1998. Springer.

[ZM93] J. M. Zelle and R. J. Mooney. Learning Semantic Grammars with Constructive Inductive Logic Programming. In *Proceedings of the 11th National Conference on Artificial Intelligence, AAAI*, pages 817–822, 1993. AAAI Press / MIT Press.

[ZM94] J. M. Zelle and R. J. Mooney. Inducing Deterministic Prolog Parsers from Treebanks. In *Proceedings of the 12th National Conference on Artificial Intelligence, AAAI*, pages 748–753, 1994. AAAI Press / MIT Press.

[ZM96] J. M. Zelle and R. J. Mooney. Learning to Parse Database Queries Using Inductive Logic. In *Proceedings of the 13th National Conference on Artificial Intelligence, AAAI '96*, pages 1050–1055, Portland, OR, 1996.

APPENDIX A

Relevant Related Publications

Some parts of this thesis are partially covered by several previously published papers. References to such publications are given below along with the Chapters and Sections in which the material appears in the thesis. They are chronologically listed and divided in two groups depending on the kwnowledge area of the Conferences in which they appeared, that is, Natural Language Processing and Machine Learning.

- Conferences on NLP.
  1. Sections **3.1**, **3.2** and **4.4**:
     L. Màrquez & L. Padró. A Flexible POS Tagger Using an Automatically Acquired Language Model. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, E/ACL '97. Madrid, Spain. July 1997.

  2. Sections **3.1**, **3.2**, **3.4** and **4.1**:
     L. Màrquez & H. Rodríguez. Automatically Acquiring a Language Model for POS Tagging Using Decision Trees. In *Proceedings of the Second Conference on Recent Advances in Natural Language Processing*, RANLP '97. Tzigov Chark, Bulgaria. September 1997.
     A revised version of this paper will appear in a book compiling a selection of papers from the RANLP '97 conference (*Title*. N. Nicolov and R. Mitkov (eds.). John Benjamins: Amsterdam & Philadelphia, 1999).

  3. Section **5.1**:
     J. Carmona, S.Cervell, L. Màrquez, M.A. Martí, L. Padró, R. Placer, H. Rodríguez, M. Taulé & J. Turmo. An Environment for Morphosyntactic Processing of Unrestricted Spanish Text. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, LREC '98. Granada, Spain. May 1998.

  4. Section **7.1**:
     L. Padró & L. Màrquez. On the Evaluation and Comparison of Taggers: the Effect of Noise in Testing Corpora. In *Proccedings of the joint 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics*, COLING-ACL '98. Montréal, Canada. August 1998.

  5. Section **5.2**:
     L. Màrquez & L. Padró. Improving Tagging Accuracy by Voting Taggers. In *Proccedings of the 2nd Conference on Natural Language Processing &*

Industrial Applications, NLP+IA/TAL+AI '98. Moncton, New Brunswick, Canada. August 1998.

6. Sections **3.4**, **4.2**, **6.2**, **6.3** and **6.4**:
   L. Màrquez, H. Rodríguez, J. Carmona & J. Montolio. Improving POS Tagging Using Machine–Learning Techniques. In *Proccedings of the joint SIGDAT conference on Empirical Methods for Natural Language Processing and Very Large Corpora*, EMNLP-VLC '99. Maryland, USA. June 1999.

- Conferences/Journals on ML.
  1. Summary of previous papers 1 and 2, and Section **3.4**:
     L. Màrquez & H. Rodríguez. Part-of-Speech-Tagging Using Decision Trees. In *Proceedings of the 10th European Conference on Machine Learning*, ECML '98. Chemnitz, Germany. April 1998.

  2. Comprehensive compilation of all previous papers, except numbers 4 and 6:
     L. Màrquez & L. Padró. A Machine Learning Approach to POS Tagging. To be published in the *Machine Learning Journal* in 1999.
     Also published as: Technical Report LSI-97-57-R, Departament LSI, Universitat Politècnica de Catalunya. December '97.

Note that the postscript versions of all previous publications are available through the personal URL: `http://www.lsi.upc.es/~lluism`

APPENDIX B

Technical Details

1. Attribute Selection Functions

In the following subsections we describe the alternative functions used for selecting attributes in our tree-induction algorithm, which are initially referenced in section 2.2.1 of chapter 3.

The basic notation followed in the descriptions is:

- X stands for a concrete set of examples, $X = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, where y_i is the class label for example $\mathbf{x}_i$.
- The values of each example $\mathbf{x}_i$ are vectors of the form $\langle x_{i,1}, x_{i,2}, \ldots, x_{i,m} \rangle$ whose components are discrete–valued attributes.
- $P(X)$ stands for a disjoint partition of X.
- $P_A(X)$ stands for the partition of X according to the values of attribute A.
- $\mathcal{C} = \{1, \ldots, K\}$ stands for the set of classes.
- $P_{\mathcal{C}}(X)$ stands for the partition of X according to the values of $\mathcal{C}$.

1.1. RLM. This measure, belonging to the distance–based family (and also information–based heuristics), was introduced by López de Mántaras in the early nineties [**Lop91**]. Roughly speaking, it defines a distance measurement between partitions and selects for branching the attribute that generates the closest partition of X to the *correct partition*, i.e., $P_{\mathcal{C}}(X)$ (the one that perfectly classifies the training data). For that, it is necessary to define a distance measurement between partitions, which will be introduced in the following steps.

Let $P(X)$ be a partition of X. The average information of such partition is defined as:

$$I(P(X)) = -\sum_{S \in P(X)} p(X,S) \log_2 p(X,S),$$

where $p(X,S)$ is the probability for an element of X belonging to the set S, and it is estimated by the ratio

$$\hat{p}(X,S) = \frac{\|X \cap S\|}{\|X\|}.$$

This average information measurement reflects the randomness of distribution of the elements of X between the classes of the partition.

If we now consider the intersection between two different partitions, $P(X)$ and $P'(X)$, we obtain:

$$I(P(X) \cap P'(X)) = -\sum_{S \in P(X)} \sum_{S' \in P'(X)} p(X, S \cap S') \log_2 p(X, S \cap S').$$

Additionally, the conditioned information of $P'(X)$ given $P(X)$, is defined as:

$$I(P'(X)|P(X)) = I(P(X) \cap P'(X)) - I(P(X)),$$

and it can be expressed as:

$$I(P'(X)|P(X)) = -\sum_{S \in P(X)} \sum_{S' \in P'(X)} p(X, S \cap S') \log_2 \frac{p(X, S \cap S')}{p(X, S)}.$$

It is easy to show that the measurement

$$d(P(X), P'(X)) = I(P'(X)|P(X)) + I(P(X)|P'(X))$$

is a distance. Normalizing, we obtain

$$d_N(P(X), P'(X)) = \frac{d(P(X), P'(X))}{I(P(X) \cap P'(X))},$$

with values in [0,1]. So, the selected attribute will be that one that minimizes the normalized distance between the partition induced by its values and the partition $P_C(X)$, that is:

$$\arg\min_A \; d_N(P_C(X), P_A(X)).$$

1.2. Information Gain. This information-based feature selection measure was initially proposed by Quinlan [**Qui79**] and it was used in the popular ID3 system and its successors [**Qui86, Qui93**].

Given a set of examples X, the information associated to the partition of the set X according to the classes of C, $P_C(X)$, is defined as

$$I(P_C(X)) = -\sum_{S \in P_C(X)} p(X, S) \log_2 p(X, S),$$

where $p(X, S)$ is the probability, for any instance of X belonging to the set S, and it is simply estimated by the proportion:

$$\hat{p}(X, S) = \frac{\|X \cap S\|}{\|X\|}.$$

In this way, $I(P_C(X))$ estimates the randomness of the distribution of the examples of X over the classes of C, in other words, it measures the amount of information that is necessary to obtain the class, among the classes of C, of any example of X.

In a similar way, when a certain node of a tree splits the set of examples X according the values of the attribute A it is possible to calculate the information needed to correctly classify any example of X using the depth-2 tree with the attribute A in its root, as the average information of the leaf nodes, weighted by its number of examples, that is:

$$E(X, A) = -\sum_{S \in P_A(X)} \frac{\|S\|}{\|X\|} I(P_C(S)).$$

Finally, the information gained by branching on attribute A is:

$$\text{Gain}(X, A) = I(P_C(X)) - E(X, A).$$

And, so, the selected attribute is the one with the highest information gain:

$$\arg\max_A \text{Gain}(X, A).$$

1.3. Gain Ratio. One problem of Quinlan's Information Gain is that it is biased in favour of attributes with many values, which are not necessarily the most useful ones. The Gain Ratio measure, proposed by the same author **[Qui86]**, is an attempt to overcome this problem. It works by normalizing the Information Gain measure, multiplying it by a factor that represents the amount of information –for any example– which is necessary to know the value of a certain attribute.

More precisely, the normalized information gain, or Gain Ratio, is defined as follows:

$$\text{Gain}_N(X, A) = \frac{\text{Gain}(X, A)}{IV(X, A)},$$

where the normalizing factor, $IV(X, A)$, is defined as:

$$IV(X, A) = -\sum_{S \in P_A(X)} p(X, S) \log_2 p(X, S).$$

and, so, it is equal to $I(P_A(X))$ of the RLM criterion. Again, the selected attribute is the one that maximizes the gain ratio:

$$\arg\max_A \text{Gain}_N(X, A).$$

1.4. Gini Diversity Index. The most commonly used impurity function for the CART algorithm is the Gini index **[BFOS84]**. It measures the class diversity at a node. Given a node with a set X of examples, the Gini diversity index of node impurity has the form:

$$I(X) = \sum_{S, S' \in P_C(X) \wedge S \neq S'} p(X, S) \cdot p(X, S'),$$

where again $p(X, S)$ is the probability for an element of X belonging to the set S (i.e. the different class probabilities given the set of examples X), and it is estimated by the ratio:

$$\hat{p}(X, S) = \frac{\|X \cap S\|}{\|X\|}.$$

When a certain node of a tree splits the set of examples X according the values of the attribute A the average impurity measure of the child nodes is calculated as:

$$I(X, A) = -\sum_{S \in P_A(X)} \frac{\|S\|}{\|X\|} I(S),$$

and the impurity reduction due to the splitting on attribute A is: $I(X) - I(X, A)$. Finally, the selected attribute is the one that produces a highest impurity reduction (in the case that all attributes produce a negative impurity decreasing, the recursion is stopped):

$$\arg\max_A (I(X) - I(X, A)).$$

One problem of the Gini criterion is that it is biased in favour of those attributes having more values.

1.5. Chi–square Test. Pearson's Chi–square (χ^2) statistic provides a test of significance with regard to the independence between variables.

For the Chi–square statistic to be used, a definition of the null hypothesis must be formulated. First, let the problem domain be specified by.

- An attribute A with M values $\{1,\ldots,M\}$.
- The set of classes: $\mathcal{C} = \{1,\ldots,K\}$.
- A set of examples X, with $\|X\| = N$.

This can be arranged in the form of a $M * K$ contingency table.

One is seeking to determine whether or not the j-th value of attribute A is a good predictor of the i-th class. If the values of A are randomly distributed among the classes, then A is not a good predictor of the class. Hence, the null hypothesis can be formulated as:

H_0: the values of A are randomly distributed over the classes of $\mathcal{C}$

If this null hypothesis is true for each value j then none of the attribute values is a good predictor. This implies that, given a particular value j for attribute A, the conditional probability of the example being in i-th class is no different from the total probability that the example is in i-th class. Thus the null hypothesis is reformulated as:

H_0: p(example-in-class i | value-of-attribute-is j) = p(example-in-class i), for **all** class i and value j

If the null hypothesis is rejected, the alternative hypothesis:

H_1: some value j of attribute A is not randomly distributed over classes of $\mathcal{C}$

is accepted. This means that some value a_j of attribute A is more likely to be associated with i–th class, and thus it can be used as a discriminator for that class.

For calculating the Chi–square statistic, the *observed* frequency O_{ij} is the number of examples (of X) in class i, with value j of attribute A. The *expected* frequency E_{ij} is defined as:

$$E_{ij} = \frac{\sum_{k=1}^{M} O_{ik} \sum_{l=1}^{K} O_{lj}}{N}$$

where O_{ik} is the observed number of examples in i-th class, and O_{lj} is the observed number of examples with value j of attribute A. The Chi–square can be now calculated as:

$$\chi^2(X,A,\mathcal{C}) = \sum_{i=1}^{K}\sum_{j=1}^{M} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

In this case the number of degrees of freedom is $M * k$. Using the comparison with the Chi–square distribution value, the null hypothesis will be rejected if the the $\chi^2(X,A,\mathcal{C})$ value is sufficiently high. This would suggest that the corresponding

attribute A is not randomly distributed over the i-th attribute and thus did discriminate between the classes. The best discriminating attribute will be the attribute with the highest Chi-square value, which will be the least likely to occur by chance (and which be selected if it is significant at say, the 10% level):

$$\arg\max_A \ \chi^2(X, A, \mathcal{C}).$$

It has been shown that the Chi-square based criterion does not favour attributes with many values, however it has other problems when working as a feature selection criterion: (1) it produces trees which are significantly larger than those produced by Information Gain, and (2) it is very sensitive to small expected frequencies (high instability for frequencies less than 5). For a detailed list of pros and cons of the χ^2 criterion, see [**SD94**].

1.6. Symmetrical Tau. The Symmetrical Tau measure [**ZD91**] is a variation of a measure of association, called Asymmetrical Tau, used for cross-classification tasks in the statistical area [**ZD91**]. The Asymmetrical Tau is measure of the relative usefulness of one variable X in improving the ability to predict the classification of members of the population with respect to a second variable Y: Tau$(Y|X)$.

Combining the two Asymmetrical measures Tau$(Y|X)$ and Tau$(X|Y)$ a balanced statistical heuristic for building multi-branching decision trees is derived: Tau(X, Y). In the context of feature selection criterion, variable X is identified with a feature A (with $\{1, \ldots, M\}$ values), and variable Y is identified with the set of classes $\mathcal{C}$ (with $\{1, \ldots, K\}$ values).

Now, let's consider the contingency table of variables A and $\mathcal{C}$, which will contain M rows and K columns. Let:

- $p(ij)$ = the probability that a variable belongs both to row category i and to column category j.
- $p(i+)$ and $p(+j)$ are the marginal probabilities in row category i and column category j, respectively.

Given that these probabilities are estimated from frequency counts on a set of examples X, the Symmetrical Tau measure is defined as follows:

$$Tau(X, A, \mathcal{C}) = \frac{\sum_{j=1}^{K}\sum_{i=1}^{M}\frac{p(ij)^2}{p(+j)} + \sum_{i=1}^{M}\sum_{i=j}^{K}\frac{p(ij)^2}{p(i+)} - \sum_{i=1}^{M} p(i+)^2 - \sum_{j=1}^{K} p(+j)^2}{2 - \sum_{i=1}^{M} p(i+)^2 - \sum_{j=1}^{K} p(+j)^2}$$

Tau has a natural and clear probabilistic interpretation. Suppose that a member of the population is selected at random and the task is to predict this member's X and Y category simultaneously. In this case, Tau is interpreted as the reduction in the probability of prediction error. This reduction results from the knowledge of the individual's classification on the second variable, relative to the probability of the error in the absence of that information.

Finally, the selected attribute will be:

$$\arg\max_A \ Tau(X, A, \mathcal{C}).$$

The Symmetrical Tau criterion has a number of interesting properties. Among others: (1) it does not favour attributes with many or few values, (2) it is a measure of association and has a *built-in* statistical strength to cope with noise, (3) it is not proportional to the sample size, (4) it is especially suited for probabilistic decision tree induction, and (5) it is able to deal with Boolean combinations of logical features. We refer the reader again to [**SD94**] for a broad description of the main properties of the Tau measure.

1.7. RELIEFF. The idea of RELIEF [**KR92**] (which is the precursor of RELIEFF) is to weight attributes according to how well their values distinguish among the instances that are near to each other. The original algorithm is described in figure 1. It randomly selects L training instances —where L is a user defined parameter— in order to iteratively adjust the weights w_k corresponding to each attribute.

For that purpose, given an instance $\mathbf{x}_t$, RELIEF searches for its two nearest neighbours: one from the same class ($\mathbf{x}_i$, called *nearest hit*) and the other from the other class ($\mathbf{x}_j$, called *nearest miss*). Note that the original RELIEF is limited to two–class problems.

The function $\delta(x_{i,k}, x_{j,k})$ returns the difference between the values of feature k of both examples $\mathbf{x}_i$ and $\mathbf{x}_j$. For discrete attributes the difference is either 1 (the values are different) or 0 (the values are equal), while for contiguous attributes the difference is the actual difference normalized to the interval $[0, 1]$. The weights w_k are estimates of the quality of the attributes, thus the justification for the updating formula is that a good attribute should have the same value for instances from the same class (subtraction of the difference $\delta(x_{t,k}, x_{i,k})$ corresponding to the nearest hit) and should differentiate between instances from different classes (adding the difference $\delta(x_{t,k}, x_{j,k})$ corresponding to the nearest miss).

---

procedure RELIEF (**in:** X, L)
X = the set of training examples
L = the number of random examples to draw
for each feature k $\{w_k := 0.0\}$
for l:=1 **to** L **do**
  randomly select an instance $(\mathbf{x}_t, y_t)$
  let $\mathbf{x}_i$ be the nearest example to $\mathbf{x}_t$ such that $y_i = y_t$
  let $\mathbf{x}_j$ be the nearest example to $\mathbf{x}_t$ such that $y_j \neq y_t$
  for each feature k
    $w_k := w_k - \delta(x_{t,k}, x_{i,k}) + \delta(x_{t,k}, x_{j,k})$
  end-for
end-for
return the set of weights w_k
end RELIEFF

FIGURE 1. Pseudo code of the RELIEF algorithm

If N is the total number of instances then the complexity of the algorithm is: $O(L \times N \times \#\text{Features})$. When used for feature selection, RELIEF simply selects the attribute which has the maximum weight, i.e. $\arg\max_k \; w_k$.

RELIEFF [**Kon94**] is an extension of RELIEF, which is able to deal with incomplete data and multi-class data sets. Its algorithm is presented in figure 2.

This new algorithm computes more reliable probability estimates by selecting groups of the B nearest neighbours instead of a single nearest neighbour. Again, the number B is a user-defined parameter that is usually set to 10.

In this way, for each randomly selected example $\mathbf{x}_t$, RELIEFF calculates the group of the B nearest hits (the set *Hit* in the algorithm) and a group of the B nearest misses for each possible class c different from the actual class of the target example (the M_c sets in the algorithm).

The updating of weights is performed in the same way than RELIEF, but the differences are averaged among all the examples in the set of B nearest neighbours and the contribution of each class different from the target class is weighted by its relative importance (the proportions p_c in the algorithm).

---

procedure RELIEFF (**in:** X, L, B)
X = the set of training examples
L = the number of random examples to draw
B = the number of nearest neighbours to compute
for each feature k $\{w_k := 0.0\}$
for each class c $\{p_c :=$ the fraction of X belonging to class $c\}$
for l:=1 **to** L **do**
  randomly select an instance $(\mathbf{x}_t, y_t)$
  let *Hit* be the set of B examples $(\mathbf{x}_i, y_i)$ nearest to $\mathbf{x}_t$ such that $y_i = y_t$
  for each class $c \neq y_t$
    let M_c be the set of B examples $(\mathbf{x}_i, y_i)$ nearest to $\mathbf{x}_t$ such that $y_i = c$
  end-for
  for each feature j

$$w_k := w_k - \frac{1}{LB} \sum_{(\mathbf{x}_i, y_i) \in Hit} \delta(x_{t,k}, x_{i,k}) + \sum_{c \neq y_t} \frac{p_y}{(1 - p_c)LB} \sum_{(\mathbf{x}_i, y_i) \in M_c} \delta(x_{t,k}, x_{i,k})$$

  end-for
end-for
return the set of weights w_k
end RELIEFF

FIGURE 2. Pseudo code of the RELIEFF algorithm

Finally, our variant RELIEFF-IG consists exactly of RELIEFF in which the distance measure used to calculate nearest hits/misses does not treat all attributes equally, but it weights them with a pre-calculated score using the Quinlan's Information gain measure over the whole set of available examples. We empirically tested that this variation increases the reliability of selecting nearest neighbours with respect to really important attributes and that the resulting trees outperform RELIEFF-based trees.

2. Relaxation Labelling for POS tagging

In this section the relaxation algorithm is described from a general point of view. Its application to POS tagging is straightforwardly performed, considering each word as a variable and each of its possible POS tags as a label.

2.1. Definitions. Let $V = \{v_1, v_2, \ldots, v_N\}$ be a set of variables (words). Let $T_i = \{t_1^i, t_2^i, \ldots, t_{m_i}^i\}$ be the set of possible labels (POS tags) for variable v_i (where m_i is the number of different labels that are possible for v_i). Let C be a set of constraints between the labels of the variables. Each constraint is a *compatibility value* for a combination of pairs variable–label:

0.53 $[(v_1, A)(v_3, B)]$	binary constraint (e.g. bi-gram)
0.29 $[(v_1, A)(v_3, B)(v_6, C)]$	ternary constraint (e.g. tri-gram)

The first constraint states that the combination of variable v_1 having label A, and variable v_3 having label B, has a compatibility value of 0.53. Similarly, the second constraint states the compatibility value for the three pairs variable–value it contains. Constraints can be of any order, so we can define the compatibility value for combinations of any number of variables.

The aim of the algorithm is to find a *weighted labeling* such that *global consistency* is maximized.

A *weighted labeling* is a weight assignment for each possible label of each variable: $P = (p^1, p^2, \ldots, p^N)$ where each p^i is a vector containing a weight for each possible label of v_i, that is: $p^i = (p_1^i, p_2^i, \ldots, p_{m_i}^i)$.

Since relaxation is an iterative process, the weights vary in time. We will note the weight for label j of variable i at time step n as $p_j^i(n)$, or simply p_j^i when the time step is not relevant.

Maximizing *global consistency* is defined as maximizing for each variable v_i, $(1 \leq i \leq N)$, the average support for that variable, which is defined as the weighted sum of the support received by each of its possible labels, that is: $\sum_{j=1}^{m_i} p_j^i \times S_{ij}$, where S_{ij} is the support received by that pair from the context.

The support for a pair variable–label (S_{ij}) expresses *how compatible* is the assignment of label j to variable i with the labels of neighbouring variables, according to the constraint set.

Although several support functions may be used, we chose the following one, which defines the support as the sum of the influence of every constraint on a label.

$$S_{ij} = \sum_{r \in R_{ij}} Inf(r),$$

where R_{ij} and $Inf(r)$ are defined as follows:

- R_{ij} is the set of constraints on label j for variable i, i.e. the constraints formed by any combination of variable–label pairs that includes the pair: (v_i, t_j^i).
- $Inf(r) = C_r \times p_{k_1}^{r_1}(m) \times \ldots \times p_{k_d}^{r_d}(m)$, is the product of the current weights for the labels appearing in the constraint except (v_i, t_j^i) (representing *how applicable* the constraint is in the current context) multiplied by C_r which is the constraint compatibility value (stating how compatible the pair is with the context).

Although the C_r compatibility values for each constraint may be computed in different ways, recent experiments [**Pad96, Pad98**] point out that the best results for POS tagging purposes are obtained when computing compatibilities as the *mutual information* between the tag and the context. Mutual information measures how informative is a discrete random variable with respect to another, and is computed as the expectation of the expression in (10) for every possible pair

of values [**CT91**]. Since we are interested on events rather than on distributions, we will use the corresponding expression for the outcomes A and B rather than its expectation [**KS97**].

$$MI(A,B) = log\frac{P(A,B)}{P(A)\cdot P(B)} \tag{10}$$

If A and B are independent events, the conditional probability of A given B will be equal to the marginal probability of A and the measurement will be zero. If the conditional probability is larger, it means than the two events tend to appear together more often than they would by chance, and the measurement yields a positive number. Inversely, if the conditional occurrence is scarcer than chance, the measurement is negative. Although Mutual information is a simple and useful way to assign *compatibility* values to our constraints, a promising possibility still to be explored is assigning them by Maximum Entropy Estimation [**Ros94, Rat97b, Ris97**].

2.2. The Algorithm. The pseudo-code for the relaxation algorithm can be found in table 3. It consists of the following steps:

1. Start in a random labeling P_0. In our case, we select a better-informed starting point, which are the lexical probabilities for each word tag.
2. For each variable, compute the support that each label receives from the current weights from other variable labels (i.e. see how compatible is the current weight assignment with the current weight assignments of the other variables, given the set of constraints).
3. Update the weight of each variable label according to the support obtained by each of them (that is, increase weight for labels with high support — greater than zero—, and decrease weight for those with low support —less than zero—). The chosen updating function in our case was:

$$p_j^i(m+1) = \frac{p_j^i(m) \times (1 + S_{ij})}{\sum_{k=1}^{k_i} p_k^i(m) \times (1 + S_{ik})}$$

4. Iterate the process until a convergence criterion is met. The usual criterion is to wait for no more changes from one iteration to the next.

The support computing and weight changing must be performed in parallel, to avoid that changing a weight for a label would affect the support computation of the others.

We could summarize this algorithm by saying that at each time-step, a variable changes its label weights depending on how compatible is that label with the labels of the other variables at that time-step. If the constraints are consistent, this process converges to a state where each variable has weight 1 for one of its labels and weight 0 for all the others.

The performed *global consistency* maximization is a vector optimization. It does not maximize —as one might think— the sum of the supports of all variables, but it finds a weighted labeling such that any other choice would not increase the support for *any* variable, given —of course— that such a labeling exists. If such a labeling does not exist, the algorithm will end in a local maximum.

$P := P_0$
repeat
  for each $v_i \in$ variables
    for each t_j possible_label_for v_i
      $S_{ij} := \sum_{r \in R_{ij}} Inf(r)$
    end-for
    for each t_j possible_label_for v_i
      $p_j^i(m+1) := \frac{p_j^i(m) \times (1 + S_{ij})}{\sum_{k=1}^{k_i} p_k^i(m) \times (1 + S_{ik})}$
    end-for
  end-for
until no_more_changes

FIGURE 3. Pseudo-code of the relaxation labelling algorithm

Note that this *global consistency* idea makes the algorithm robust: The problem of having mutually incompatible constraints (there is no combination of label assignment which satisfies all the constraints) is solved because relaxation does not necessarily find an exclusive combination of labels —i.e. a unique label for each variable—, but a weight for each possible label such that constraints are satisfied to the maximum possible degree. This is especially useful in our case, since constraints will be automatically acquired, and different knowledge sources will be combined, so constraints might not be fully consistent.

The advantages of the algorithm are:

- Its highly local character (each variable can compute its new label weights given only the state at previous time-step). This makes the algorithm highly parallelizable (we could have a processor to compute the new label weights for each variable, or even a processor to compute the weight for each label of each variable).
- Its expressiveness, since we state the problem in terms of constraints between variable labels. In our case, this enables us to use binary (bi-gram) or ternary (tri-gram) constraints, as well as more sophisticated constraints (decision tree branches or hand-written constraints).
- Its flexibility, we do not have to check absolute consistency of constraints.
- Its robustness, since it can give an answer to problems without an exact solution (incompatible constraints, insufficient data, ...)
- Its ability to find local-optima solutions to np problems in a non-exponential time (only if we have an upper bound for the number of iterations, i.e. convergence is fast or the algorithm is stopped after a fixed number of iterations).

The drawbacks of the algorithm are:

- Its cost. N being the number of variables, v the average number of possible labels per variable, c the average number of constraints per label, and I the average number of iterations until convergence, the average cost is $N \times v \times$

$c \times I$, that is, it depends linearly on N, but for a problem with many labels and constraints, or if convergence is not quickly achieved, the multiplying terms might be much bigger than N. In our application to POS tagging, the bottleneck is the number of constraints, which may be several thousand. The average number of tags per ambiguous word is about 2.5, and an average sentence contains about 10 ambiguous words.

- Since it acts as an approximation of gradient descent algorithms, it has their typical convergence problems: Found optima are local, and convergence is not guaranteed, since the chosen step might be too large for the function to optimize.
- In general relaxation labeling applications, constraints would be written manually, since they are the modeling of the problem. This is good for easy–to–model domains or reduced constraint–set problems, but in the case of POS tagging, constraints are too many and too complicated to be easily written by hand.
- The difficulty of stating by hand what the *compatibility value* is for each constraint. If we deal with combinatorial problems with an exact solution (e.g. traveling salesman), the constraints will be either fully compatible (e.g. stating that it is possible to go to any city from any other), fully incompatible (e.g. stating that it is not possible to be twice in the same city), or will have a value straightforwardly derived from the distance between cities. But if we try to model more sophisticated or less exact problems (such as POS tagging), we will have to establish a way of assigning graded compatibility values to constraints.
- The difficulty of choosing the most suitable support and updating functions for each particular problem.

APPENDIX C

Tag Sets

1. WSJ Corpus Tagset

Figure 1 contains a description of the Penn Treebank tagset, used for tagging the WSJ corpus. For a complete description of the corpus see the paper by Mercus et al. [MMS93].

CC Coordinating conjunction	PRP Personal pronoun	WP$ Possessive *wh*-pronoun
CD Cardinal number	PP$ Possessive pronoun	WRB *wh*-adverb
DT Determiner	RB Adverb	# Pound sign
EX Existential *there*	RBR Adverb, comparative	$ Dollar sign
FW Foreign word	RBS Adverb, superlative	. End of sentence
IN Preposition	RP Particle	, Comma
JJ Adjective	SYM Symbol	: Colon, semi-colon
JJR Adjective, comparative	TO *to*	(Left bracket character
JJS Adjective, superlative	UH Interjection	) Right bracket character
LS List item marker	VB Verb, base form	" Straight double quote
MD Modal	VBD Verb, past tense	` Left open single quote
NN Noun, singular	VBG Verb, gerund	`` Left open double quote
NNP Proper noun, singular	VBN Verb, past participle	' Right close single quote
NNS Noun, plural	VBP Verb, non-3rd ps. sing. present	'' Right close double quote
NNPS Proper noun, plural	VBZ Verb, 3rd ps. sing. present	
PDT Predeterminer	WDT *wh*-determiner	
POS Possessive ending	WP *wh*-pronoun	

FIGURE 1. The Penn Treebank tagset

2. LEXESP Corpus Tagset

The tagset used to annotate the LEXESP corpus consists of a set of PAROLE compliant labels that were specially developed for Spanish and Catalan. In this labels, the first symbol codifies main syntactic categories, the second symbol codifies sub-categories, and the rest codify other features such as gender, number, person, tense, etc.

The full tagset is too large to be used in a statistically-based tagger, thus a reduced version was constructed to perform POS tagging. The reduced tagset limits the information to the two first symbols of each label (three in the case of verbs). It is described below.

- **Adjective:** AQ.
  (Q=qualifier).
- **Adverb:** RG.
  (G=general).
- **Article:** TD, TI, TP.
  (D=definite; I=indefinite; P=personal).
- **Determiner:** DD, DP, DT, DE, DI.
  (D=demonstrative; P=possessive; T=interrogative; E=exclamative; I=indefinite).
- **Noun:** NC, NP.
  (C=common; P=proper).
- **Verb:** VMI, VMS, VMM, VMC, VMN, VMG, VMP, VAI, VAS, VAM, VAC, VAN, VAG, VAP.
  (2nd symbol: M=main; A=auxiliary. 3rd symbol: I=indicative; S=subjunctive; M=imperative; C=conditional; N=infinitive; G=gerund; P=participle).
- **Pronoun:** PP, PD, PX, PI, PT, PR.
  (P=personal; D=demonstrative; X=possessive; I=indefinite; T=interrogative; R=relative).
- **Conjunction:** CC, CS.
  (C=coordinate; S=subordinate).
- **Numerals:** MC, MO.
  (C=cardinal; O=ordinal).
- **Interjection:** I.
- **Abbreviation:** Y.
- **Punctuation marks:** {., ,, ;, :, -, ¡, ', ', '', '', !, ¿, ?, (,), , ...}
- **Residuals:** X.

APPENDIX D

An Example Tree

We list in this appendix the full size tree referenced as an example in section 2.3 of chapter 3, which is the decision tree acquired from 8,012 examples of the preposition–adverb (IN-RB) ambiguity class.

The notation employed is lisp–like and it has the be read in the following way. For instance, the fragment:

```
...
((VBZ VBP) tag(+1) (34 137) 3 0
   (child-node-1)
   (child-node-2)
   (child-node-3)
)
...
```

would represent an internal node that comes from its parent through an edge labelled with "VBZ VBP" (which means either of two tags). The tree contains a split using the 'tag(+1)' attribute which branches to three child nodes with the corresponding values (not printed). Finally, '(34 137)' are the class counts of examples associated to the tree (meaning 34 instances labelled RB, and 137 instances labelled IN). Observe that leaf nodes contain a 0 in the second position of the list.

Finally, note that the branch mentioned in that examples has been emphasized between brackets (<<< and >>>).

```
(Root word(0) (809 7203) 19 0
  ((since after) 0 (19 948) 0 2)
  ((once) tag(+1) (91 17) 7 0
     ((JJ DT) tag(-1) (28 5) 6 0
        ((RB CC JJS DT VB) 0 (13 0) 0 10)
        ((IN NNS) 0 (0 2) 0 2)
        ((VBD) 0 (4 0) 0 10)
        ((,) 0 (9 0) 0 10)
        ((NN) 0 (0 3) 0 2)
        ((VBZ VBN) 0 (2 0) 0 10))
     ((PRP) 0 (0 9) 0 2)
     ((JJR . IN) 0 (7 0) 0 10)
     ((VBD) 0 (23 0) 0 10)
     ((NNP) tag(-1) (4 3) 3 0
        ((,) 0 (2 0) 0 10)
        ((NNP JJ CC) 0 (0 3) 0 2)
        ((NN JJS) 0 (2 0) 0 10))
     ((TO CC CD JJS ) WRB , MD) 0 (8 0) 0 10)
     ((VBN RB) 0 (21 0) 0 10))
```

```
((Once) tag(+2) (22 7) 4 0
   ((DT ,) 0 (5 0) 0 10)
   ((VBZ VBP NNS) tag(-2) (2 6) 3 0
      ((:) 0 (0 1) 0 2)
      ((<null>) 0 (0 4) 0 2)
      ((.) 0 (2 1) 0 10))
   ((NN) 0 (11 0) 0 10)
   ((JJ CC VBD) tag(-2) (4 1) 2 0
      ((<null>) 0 (4 0) 0 10)
      ((.) 0 (0 1) 0 2)))
((notwithstanding) tag(-1) (1 3) 3 0
   ((,) 0 (0 2) 0 2)
   ((NN) 0 (1 0) 0 10)
   ((NNP) 0 (0 1) 0 2))
((aboard) 0 (0 5) 0 2)
((between) 0 (0 348) 0 2)
((beside) 0 (0 2) 0 2)
((Alongside About) tag(-2) (17 26) 2 0
   ((DT .) 0 (2 0) 0 10)
   ((<null>) tag(+1) (15 26) 4 0
      ((DT) tag(+2) (1 5) 2 0
         ((JJ) 0 (0 3) 0 2)
         ((NN) 0 (1 2) 0 2))
      (($) 0 (2 2) 0 10)
      ((NN JJ) 0 (0 2) 0 2)
      ((CD) tag(+2) (12 17) 2 0
         ((NN) 0 (6 5) 0 10)
         ((VBP NNP JJ CD NNS) 0 (6 12) 0 2))))
((though) tag(+1) (34 137) 7 0
   ((`` JJS CD PRP$ EX) 0 (0 5) 0 2)
   ((NNP NNS RB JJ IN) 0 (1 43) 0 2)
   ((NN) 0 (0 4) 0 2)
   ((,) tag(-1) (27 2) 3 0
      ((DT RB) 0 (2 0) 0 10)
      ((,) 0 (25 1) 0 10)
      ((NN) 0 (0 1) 0 2))
   ((PRP DT) 0 (0 81) 0 2)
   ((.) 0 (6 0) 0 10)
   ((JJR) 0 (0 2) 0 2))
((`til) 0 (1 1) 0 10)
((After) 0 (2 107) 0 2)
((Before) 0 (1 25) 0 2)
<<<((As as) tag(+1) (534 2719) 14 0 >>>
   ((DT) 0 (0 801) 0 2)
   ((CD NNP NNS PRP) 0 (1 703) 0 2)
   ((`` NN) 0 (3 366) 0 2)
   ((VB FW EX) 0 (0 6) 0 2)
   ((VBN VBP) tag(-1) (10 64) 8 0
      ((VBZ RP CC VB VBG) 0 (5 0) 0 10)
      ((PRP) 0 (1 1) 0 10)
      ((JJ VBN RBR NNPS) 0 (0 4) 0 2)
      ((RB :) tag(-2) (1 7) 3 0
         ((RB) 0 (0 4) 0 2)
```

```
      ((VBD) 0 (1 0) 0 10)
      ((NNP JJ NNS) 0 (0 3) 0 2))
   ((,) 0 (1 41) 0 2)
   ((NN) 0 (0 4) 0 2)
   ((VBD) 0 (2 0) 0 10)
   ((NNS) 0 (0 7) 0 2))
((JJ) tag(+2) (236 240) 10 0
   ((TO) tag(-3) (8 2) 3 0
      ((VBZ PRP NNP) 0 (6 0) 0 10)
      ((NNS IN) 0 (0 2) 0 2)
      ((JJ RB) 0 (2 0) 0 10))
   ((CD , MD VBD) tag(-1) (4 16) 5 0
      ((RB) tag(-2) (2 7) 3 0
         ((, JJ . VBG) 0 (0 4) 0 2)
         ((RB) 0 (0 3) 0 2)
         ((VBP VBZ) 0 (2 0) 0 10))
      ((,) 0 (1 1) 0 10)
      ((NNP NN VBN) 0 (0 7) 0 2)
      ((NNS) 0 (0 1) 0 2)
      ((IN) 0 (1 0) 0 10))
   (($ () 0 (2 0) 0 10)
   ((VBN CC RB) tag(-3) (6 7) 4 0
      ((IN NNS) 0 (0 4) 0 2)
      ((PRP NN) 0 (4 0) 0 10)
      ((CC : VBD) 0 (0 3) 0 2)
      ((MD VBG) 0 (2 0) 0 10))
   ((VBP) 0 (0 2) 0 2)
   ((IN DT) tag(-1) (195 16) 10 0
      ((JJ) 0 (0 4) 0 2)
      ((CC VBZ) 0 (19 0) 0 10)
      ((IN) 0 (52 0) 0 10)
      ((VB) 0 (33 0) 0 10)
      ((:) 0 (2 0) 0 10)
      ((NNS VBG) tag(-3) (11 1) 3 0
         ((MD) 0 (2 0) 0 10)
         ((VBZ TO PRP$ CD DT VB VBD NNS VBG) 0 (9 0) 0 10)
         ((IN) 0 (0 1) 0 2))
      ((RB VBD) 0 (50 1) 0 10)
      ((`` RP) 0 (2 0) 0 10)
      ((VBN ,) tag(-3) (3 9) 3 0
         ((JJR NN NNS) 0 (3 0) 0 10)
         ((IN) 0 (0 2) 0 2)
         ((MD RB CC VBP VBD , NNP) 0 (0 7) 0 2))
      ((TO VBP NN) 0 (23 1) 0 10))
   ((JJ NN) tag(-1) (9 82) 8 0
      ((VBZ VBP VBG) 0 (3 0) 0 10)
      ((JJ) 0 (0 17) 0 2)
      ((NN NNS) 0 (0 22) 0 2)
      ((CC VB) tag(-3) (3 3) 2 0
         ((VBZ <null> VBN) 0 (0 3) 0 2)
         ((, NN VBG) 0 (3 0) 0 10))
      ((IN CD) tag(-2) (1 3) 2 0
         ((NNP JJ NN) 0 (0 3) 0 2)
```

```
            ((NNS) 0 (1 0) 0 10))
         ((RB ,) 0 (1 21) 0 2)
         ((VBD) 0 (0 2) 0 2)
         ((VBN NNP) tag(-3) (1 14) 3 0
            ((RB JJ VB NNP ,) 0 (0 5) 0 2)
            ((MD) 0 (1 1) 0 10)
            ((IN NN) 0 (0 8) 0 2)))
      ((NNP) tag(-2) (1 8) 3 0
         ((VB) 0 (1 0) 0 10)
         ((RB NN JJ) 0 (0 3) 0 2)
         ((, IN) 0 (0 5) 0 2))
      ((.) tag(-2) (8 15) 4 0
         ((VB NN) 0 (0 5) 0 2)
         ((VBZ PRP) 0 (5 0) 0 10)
         ((CD VBP) 0 (2 0) 0 10)
         ((POS RB VBN DT VBD , VBG) tag(-1) (1 10) 4 0
            ((VB) 0 (1 0) 0 10)
            ((NN) 0 (0 3) 0 2)
            ((RB) 0 (0 5) 0 2)
            ((: NNS) 0 (0 2) 0 2)))
      ((NNS) 0 (3 92) 0 2))
   <<<((RB) tag(+2) (269 40) 8 0 >>>
      ((( ) PRP JJ) tag(-1) (10 3) 4 0
         ((RB VBN IN) 0 (3 0) 0 10)
         ((VBD) 0 (0 2) 0 2)
         ((JJ) 0 (0 1) 0 2)
         ((, NN NNS) 0 (7 0) 0 10))
      ((,) tag(-1) (2 15) 4 0
         ((VBP VBD) 0 (0 2) 0 2)
         ((JJ RB , '' NNS) 0 (0 12) 0 2)
         ((NN) 0 (1 1) 0 10)
         ((VBN) 0 (1 0) 0 10))
      ((VBN :) tag(-1) (11 2) 4 0
         ((: VBZ NNS) 0 (3 0) 0 10)
         ((NN) 0 (3 0) 0 10)
         ((,) 0 (5 0) 0 10)
         ((JJ RB) 0 (0 2) 0 2))
      ((TO .) tag(-1) (25 4) 6 0
         ((VB RB NNP , VBN) 0 (12 0) 0 10)
         ((NN) 0 (8 0) 0 10)
         ((JJ) 0 (0 3) 0 2)
         ((RBR) 0 (0 1) 0 2)
         ((VBG) 0 (1 0) 0 10)
         ((NNS) 0 (4 0) 0 10))
      ((JJR RB NN) tag(-3) (4 4) 4 0
         ((DT RB) 0 (0 2) 0 2)
         ((NN) 0 (0 2) 0 2)
         ((PRP IN) 0 (2 0) 0 10)
         ((NNS) 0 (2 0) 0 10))
      ((VBG) 0 (0 2) 0 2)
   <<<((IN) 0 (215 3) 0 10)>>>
      ((`` NNS DT VBD) tag(-3) (2 7) 4 0
         ((VBP) 0 (1 1) 0 10)
```

```
          ((VBZ) 0 (1 0) 0 10)
          ((VB '') 0 (0 2) 0 2)
          ((NN NNS) 0 (0 4) 0 2)))
      ((WP WRB VBZ) tag(-3) (2 17) 5 0
         ((PRP) 0 (1 0) 0 10)
         ((NN) 0 (0 4) 0 2)
         (($ RBR DT .) 0 (0 4) 0 2)
         ((IN) 0 (1 1) 0 10)
         ((JJ RB NNS) 0 (0 8) 0 2))
      ((# , WDT :) 0 (0 4) 0 2)
      ((PRP$) tag(-3) (2 34) 5 0
         ((DT NN , JJ NNS) 0 (0 11) 0 2)
         ((NNP) 0 (0 11) 0 2)
         ((RB CC) 0 (2 0) 0 10)
         ((VBP RP VBZ VBD PRP$ VBN VBG RBS) 0 (0 8) 0 2)
         ((IN) 0 (0 4) 0 2))
      ((JJR) tag(-3) (3 5) 3 0
         ((RB VBN NNS) 0 (0 3) 0 2)
         ((JJ) 0 (0 2) 0 2)
         ((DT MD NN) 0 (3 0) 0 10))
      ((TO IN) 0 (4 94) 0 2)
      ((VBD NNPS) 0 (0 14) 0 2)
      (($ VBG) 0 (0 90) 0 2))
   ((Along before below) tag(+1) (45 452) 9 0
      ((VBN NNS RB) tag(-3) (4 15) 3 0
         (($ DT . NNS) 0 (0 9) 0 2)
         ((TO JJ '') 0 (3 0) 0 10)
         ((<null> VBN IN VB NNP) tag(-2) (1 6) 3 0
            ((TO NN . NNS) 0 (0 4) 0 2)
            ((JJ) 0 (0 2) 0 2)
            ((PRP$) 0 (1 0) 0 10)))
      ((CD DT VBG) 0 (7 248) 0 2)
      (($ CC) 0 (0 12) 0 2)
      ((IN) tag(-3) (1 7) 3 0
         ((VBP) 0 (1 0) 0 10)
         ((VBZ VBN) 0 (0 2) 0 2)
         ((<null>) 0 (0 5) 0 2))
      ((WP JJS PDT '') 0 (0 4) 0 2)
      ((VBP , . : EX) tag(-3) (27 12) 4 0
         ((VBN PRP VBZ) tag(+2) (3 3) 2 0
            (('') 0 (2 0) 0 10)
            ((<null> CC) tag(-2) (1 3) 2 0
               ((VBZ PRP) 0 (0 2) 0 2)
               ((VBN) 0 (1 1) 0 10)))
         ((NN VBD) tag(+2) (9 1) 4 0
            ((<null>) 0 (4 0) 0 10)
            ((DT '' IN) 0 (3 0) 0 10)
            ((CC) 0 (2 0) 0 10)
            ((VBG) 0 (0 1) 0 2))
         ((CD VBP , MD JJ RB IN EX) tag(-1) (15 5) 4 0
            ((JJ RB NNS VBG) tag(+2) (9 3) 2 0
               ((DT) 0 (6 3) 0 10)
               ((<null>) 0 (3 0) 0 10))
```

```
          ((PRP) 0 (0 1) 0 2)
          ((VBN) 0 (1 1) 0 10)
          ((NN IN) 0 (5 0) 0 10))
       ((CC NNS DT) 0 (0 3) 0 2))
    ((PRP NN) 0 (0 87) 0 2)
    ((VBD) 0 (3 3) 0 10)
    ((NNP JJ PRP$) 0 (3 64) 0 2))
((In) 0 (0 1253) 0 2)
((because) tag(+1) (38 683) 10 0
    ((NNS PRP$) 0 (0 45) 0 2)
    ((JJS CD ) WRB VBG) tag(-2) (2 9) 3 0
       ((VB ,) 0 (0 5) 0 2)
       ((MD RB) 0 (2 0) 0 10)
       ((NN PRP$ CC VBG) 0 (0 4) 0 2))
    ((PRP) 0 (0 180) 0 2)
    ((JJR RB) 0 (0 7) 0 2)
    ((IN) tag(-1) (30 224) 13 0
       ((VB) tag(-3) (4 2) 3 0
          ((NN) 0 (1 1) 0 10)
          ((VBD) 0 (0 1) 0 2)
          ((MD RB NNS) 0 (3 0) 0 10))
       ((``) 0 (1 1) 0 10)
       ((( '') 0 (0 4) 0 2)
       ((NN) 0 (0 54) 0 2)
       ((RB) 0 (2 51) 0 2)
       ((NNP) 0 (0 10) 0 2)
       ((,) 0 (0 28) 0 2)
       ((JJR VBP PRP JJ VBD FW VBG) tag(-2) (6 21) 5 0
          ((VBN NN VBG) 0 (3 0) 0 10)
          ((`` VBP WDT NNS) tag(-3) (1 5) 3 0
             ((.) 0 (1 0) 0 10)
             ((NNS) 0 (0 2) 0 2)
             ((NN VBD VBG) 0 (0 3) 0 2))
          ((IN VB) 0 (0 4) 0 2)
          ((VBD) 0 (1 1) 0 10)
          ((RB) tag(+2) (1 11) 4 0
             ((DT) 0 (0 5) 0 2)
             ((`` JJ NN) 0 (0 3) 0 2)
             ((NNP) 0 (1 0) 0 10)
             ((NNS) 0 (0 3) 0 2)))
       ((CC) tag(-3) (2 3) 3 0
          ((CD IN) 0 (2 0) 0 10)
          ((<null>) 0 (0 2) 0 2)
          ((NNS) 0 (0 1) 0 2))
       ((VBN) tag(-3) (8 3) 2 0
          ((PRP TO VBD RB NN , VBN) tag(-2) (8 1) 2 0
             ((VB) 0 (2 0) 0 10)
             ((VBZ VBD RB CC) 0 (6 1) 0 10))
          ((VBZ WDT) 0 (0 2) 0 2))
       ((IN) 0 (0 7) 0 2)
       ((NNS) tag(+2) (6 30) 5 0
          ((DT NNS) tag(-3) (2 15) 3 0
             ((TO IN DT NNP NN) 0 (0 11) 0 2)
```

```
            ((MD PRP$) 0 (2 0) 0 10)
            ((JJ VB NNS WDT) 0 (0 4) 0 2))
         ((NN) tag(-2) (4 2) 3 0
            ((DT JJ) 0 (0 2) 0 2)
            ((VB CD) 0 (2 0) 0 10)
            ((NN) 0 (2 0) 0 10))
         ((JJ) 0 (0 8) 0 2)
         ((`` VBN JJR) 0 (0 3) 0 2)
         ((PRP$) 0 (0 2) 0 2))
      ((CD) tag(+2) (1 10) 3 0
         ((JJ) 0 (0 2) 0 2)
         ((NNP NN VBN) tag(-3) (1 4) 2 0
            (($) 0 (0 2) 0 2)
            ((NN) 0 (1 2) 0 2))
         ((NNS) 0 (0 4) 0 2)))
   ((DT) 0 (0 110) 0 2)
   ((`` JJ EX) 0 (2 43) 0 2)
   ((NN) 0 (1 24) 0 2)
   ((NNP) 0 (0 38) 0 2)
   ((,) tag(-2) (3 3) 3 0
      ((VBP JJ NNP) 0 (3 0) 0 10)
      ((NN) 0 (0 2) 0 2)
      ((VBD) 0 (0 1) 0 2)))
((beyond Besides) 0 (1 55) 0 2)
((under) 0 (1 405) 0 2))
```

APPENDIX E

Research Projects and other Links

The research reported in this dissertation has been developed inside the framework of mainly four research projects. They are briefly described below.

1. ACQUILEX-I(II) Projects

The ACQUILEX projects were funded by the European Commission under the Basic Research initiative The goal of the first project was to explore the utility of constructing a multilingual lexical knowledge base from machine-readable versions of conventional dictionaries. The second project extended this goal by exploring the utility of machine readable textual corpora as a source of lexical information not coded in conventional dictionaries, and by adding dictionary publishing partners to exploit the lexical database and corpus extraction software developed by the projects for conventional lexicography. The ACQUILEX-II project finished in September 1995. Partners in this project were: University of Cambridge, University of Amsterdam, Instituto di Linguistica Computazionale of CNR, the joint NL research group from the Catalan Polytechnical University (UPC) and University of Barcelona (UB), and the publishing partners: Cambridge University Press (UK), Biblograf (Spain), and Van Dale Lexicografie (The Netherlands).

ACQUILEX homepage: http://www.cl.cam.ac.uk/Research/NL/acquilex

2. EuroWordNet Project

EuroWordNet (LE-2 4003 & LE-4 8328) is a resources and development project supported by the Human Language Technology sector of the Telematics Applications Programme.

It aims to develop a generic multilingual database with WordNets for several European languages –English, Dutch, Italian and Spanish– with 30,000 senses each one. Those WordNets will be linked through the English WordNet, so each English synonym will be associated with its equivalent in the other languages. Partners in this project are: University of Amsterdam, University of Sheffield, Instituto di Linguistica Computazionale of CNR, Novell Belgium and the joint NL research group from the Catalan Polytechnical University (UPC), the University of Barcelona (UB) and the Spanish Open University (UNED).

EuroWordNet homepage: http://www.let.uva.nl/~ewn
Wordnet homepage: http://www.cogsci.princeton.edu/~wn

3. ITEM Project

ITEM is a project funded by Spanish Research Department (CICYT) consisting basically of integrating different existing NLP tools and resources in a unique environment, in order to enable and ease the construction of multilingual information extraction and retrieval systems.

It includes tools for NLP of Catalan, Basque and Spanish. The integrated tools cover basic NL tasks (tokenizers, morphological analyzers, taggers, parsers, etc.) as well as higher level tasks oriented to information extraction. The integration environment also contains several lexical resources such as corpus, machine-readable dictionaries (MRDs), lexicons, taxonomies, grammars, etc. Tools and resources are documented, available and transportable.

Partners in this project are the Computational Linguistics Group from the University of Barcelona (UB), the NLP research group from the Polytechnical University of Catalonia (UPC) the NLP group from the Basque Country University (EHU), and the NLP group from the Spanish Open University (UNED).

ITEM homepage: http://sensei.ieec.uned.es/item

4. LEXESP Project

The LEXESP Project is a multi-disciplinary effort headed by the Psychology Department of the University of Barcelona in collaboration with the Psychology Department of the University of Oviedo. It aims to create a large database of language usage in order to enable and encourage research activities in a wide range of fields, from linguistics to medicine, through psychology and artificial intelligence, among others. One of the main issues of this database of linguistic resources is the LEXESP corpus, which contains 5.5 Mw of written material, including general news, sports news, literature, scientific articles, etc., and which aims to be a balanced and general sample of modern Spanish language usage.

The corpus will be morphologically analyzed and disambiguated, and syntactically parsed. The provided tools and resources include a broad coverage morphological analyzer, two taggers for Spanish, and a chart parser.

LEXESP homepage: http://www.ub.es/pbasic/recerca.htm

Additional information can be found at the homepage of the Natural Language Research Group, of the Catalan Polytechnical University (Dep. LSI). This page allow the user to access to on-line demonstrations of the aforementioned tools. (they include analyzers for Catalan language also).

NLRG homepage: http://www.lsi.upc.es/~acquilex/nlrg.html