# Using a Large Set of EAGLES-compliant Morpho-Syntactic Descriptors as a Tagset for Probabilistic Tagging

**Dan Tufiş**

RACAI-Romanian Academy
13, "13 Septembrie", Ro-74311, Bucharest 5, Romania
tufis@valhalla.racai.ro

**Abstract**

The paper presents one way of reconciling data sparseness with the requirement of high accuracy tagging in terms of fine-grained tagsets. For lexicon encoding, EAGLES elaborated a set of recommendations aimed at covering multilingual requirements and therefore resulted in a large number of features and possible values. Such an encoding, used for tagging purposes, would lead to very large tagsets. For instance, our EAGLES-compliant lexicon required a set of about 1000 morpho-syntactic description codes (MSDs) which after considering some systematic syncretic phenomena, was reduced to a set of 614 MSDs. Building reliable language models (LMs) for this tagset would require unrealistically large training data (hand annotated/validated). Our solution was to design a hidden reduced tagset and use it in building various LMs. The underlying tagger uses these LMs to tag a new text in as many variants as LMs are available. The tag differences between these variants are processed by a combiner which chooses the most likely tags. In the end, the tagged text is subject to a conversion process that maps the tags from the reduced tagset onto the more informative tags from the large tagset. We describe this processing chain and provide a detailed evaluation of the results.

## Large tagsets and tiered tagging

The paper discusses experiments and results concerned with tagging highly inflectional languages, based on multiple register diversified language models (LMs). The case study language is Romanian, for the tagset of which we adopted the internationally accepted set of EAGLES guidelines for morpho-syntactic encoding of lexica. The Romanian lexicon, EAGLES compliant, was built within the MULTEXT-EAST Copernicus Joint Project and the description of its almost half a million wordforms used a set of 614 morpho-syntactic description (MSD) codes. A full description of the encoding scheme we used is given in (Erjavec & Monachini, 1995). Multilingual content analyses of the MULTEXT-EAST lexica and corpora are presented in (Tufiş et al, 1998; Dimitrova et al, 1998).

In order to cope with the inherent problems raised by such a large tagset in statistical morpho-syntactical tagging, we designed a reduced tagset (Ctag-set), which is used for an intermediary tagging, hidden to the beneficiary (human or software) of the disambiguated text. The training corpora were hand annotated in terms of the large MDS tagset. The MSD annotated corpora were automatically converted into Ctag-set annotated ones. The language models that drive the disambiguation of a text were obtained from training the tagger on these Ctag-set annotated corpora. A new text is tagged in terms of the Ctag-set followed by a second processing phase, lexicon driven, which replaces the tags in the hidden tagset with the more informative tags from the large tagset. We call this process *tiered tagging*.

Thus, with a small price in tagging accuracy (as compared to the direct reduced tagset approach), and practically no price in computational resources, it is possible to tag a text with a large tagset by using language models built for reduced tagsets and reasonably large training data.

The Ctag-set for Romanian consists of 92 tags, plus 10 punctuation tags. The relation between the MSD-set and the Ctag-set is encoded as a mapping table that specifies for each MSD the corresponding Ctag and for each Ctag the set of MSDs that are mapped onto it. The post-processor that deterministically replaces one Ctag by one or more MSDs, is essentially a database look-up procedure. It takes as input parameters a lexical token plus the Ctag the token was assigned to in the previous phase and returns one or more (2, rarely 3) MSDs. The operation is equivalent to computing the intersection of the ambiguity class of the lexical token and the set of all the MSDs that maps onto the tag assigned to the token in case. As the first set of this intersection depends on the lexical token, it is obvious that a Ctag is not always mapped onto the same MSD or set of MSDs. The tokens that this replacement makes ambiguous are more often than not the difficult cases in statistical disambiguation. Very simple contextual (unification) rules differentiate the interpretations of the few still ambiguous items. These rules investigate (depending on the ambiguity case) left, right or both contexts within a limited distance (in our experiment the maximum span is 4) for a disambiguating tag or wordform. The success rate of this second phase is almost 99%. Below is given such a rule, out of the 14 rules we currently use. The rule disambiguates between possessive pronouns and possessive determiners (a notoriously difficult case for statistical disambiguation):

```
Ps|Ds{Ds.αβδ:(-1 Ncαβδy)|(-1 Af.αβδy)|(-1 Mo.αβδy)|
                (-2 Af.αβδn & –1 Ts)|(-2 Ncαβδn &-1 Ts)|
                (-2 Np and –1 Ts)|(-2 D..αβδ and -1 Ts)
        Ps. αβδ: true}
```

In the rule above, α,β and δ stand for shared feature values, taking care of agreement in *gender*, *number* and *case* respectively, while '.' stands for the '*any*' value of the attribute in the corresponding position. For instance, in Ds.αβδ and Ps.αβδ the dot stands for any value of the *person* attribute, but in Af.αβδy, the dot stands for any value of the "*degree of comparison*" attribute of the adjective. The (non-recursive) unification mechanism underlying the rule interpreter ensures the attribute value binding. The reading of the rule is as follows:

*When there exist an ambiguity between a possessive pronoun (Ps.αβδ) and a possessive determiner (Ds.αβδ)*
*if*
*the previous word is tagged as a definite Noun, or a definite Adjective, or a definite Numeral (ordinal), or when the previous two words are tagged as indefinite noun followed by a possessive article or proper noun followed by a possessive article determiner followed by a possessive article*
*then*
*choose* ***the determiner interpretation***
*else if none of above holds*
*choose* ***the pronoun interpretation***.

## Corpus tagset design

The design of an Ctag-set from an MSD-tagset is based on a trial & error procedure. A key property required for the Ctag-set is what we call *MSD-tagset recoverability*, formally described by expression (1). We use the following notations: $W_i$ represents a word from the lexicon (Lex), $T_i$ represents a tag from the reduced tagset (Ctag-set) assigned to $W_i$, $MSD_k$ represents a tag from the MSD-tagset, $AMB(W_k)$ represents the ambiguity class of the word $W_k$ in terms of MSDs (as encoded in Lex), MAP is an application that maps each $T_i$ onto a subset of MSD-set and |X| represents the number of elements of the set X.

(1) $\forall Ti \in$ Ctag-set, $MAP(Ti)= \{MSD1 \ldots MSDk\} \subset$ MSD-tagset,
$\forall W_k \in$ Lex & $AMB(W_k)=\{MSD_{k1} \ldots MSD_{kn}\} \subset$ MSD-tagset $\Rightarrow$

$$\left|MAP(Ti) \bigcap AMB(Wk)\right| = \begin{cases} 1 & \text{for more than 90\% cases} \\ >1 & \text{for less than 10\% cases} \end{cases}$$

The *initial* phase of the Ctag-set design is described by the following procedure:

*a) extract all ambiguity classes from the lexicon*
*b) normalize all MSD ambiguity classes*
*c)* ***for each*** *ambiguity class* $AC_i$
*preserve only intra-categorical ambiguities:* $ICA_{ij}$
*d)* ***for each*** $ICA_i$ ***repeat***
***for each*** $MSD_{ij}$ ***repeat***
***for each*** *attribute* $A_k$ *in* $MSD_{ij}$ ***repeat***
***if*** *eliminating* $A_k$ *would not reduce any ICA's card*
***then*** *remove* $A_k$ *from all tags and update ICAs*
***else if*** *eliminating* $A_k$ *would reduce the card for no more than 10% of ICAs*
***then*** *mark* $A_k$ *as removable* ***endif endif***
***endfor***
***endfor***
***endfor***
*e) for all* $A_k$ *marked as removable, compute the maximal set of attributes that minimally reduces the cardinal of all ICAs (not unique solution)*
*f) for each Ctag-set obtained in the step e) evaluate the performance*

In the algorithm sketched above, step b) was introduced to deal with the feature values syncretism while step c) was motivated by the empirical observation that, for highly inflectional languages, inter-categorical misclassifications are much less frequent then intra-categorical ones. The following MSD ambiguity class displays only inter-categorical ambiguities, therefore no ICA will be created (LC represents the number of distinct wordforms in the lexicon that are associated with this ambiguity class):

$MSD\text{-}AC_i$*=(Afpms-n Ncms-n Vmp--sm) LC =715*

On the other hand, the ambiguity class $MSD\text{-}AC_j$, containing intra-categorical ambiguities would generate two ICAs:

$MSD\text{-}AC_j$*=(Ncfp-n Ncfson Vmis3s Vmm-2s Vmnp)LC=33*
$ICA_{j1}$*=(Ncfp-n Ncfson),* $ICA_{j2}$*=(Vmis3s Vmm-2s Vmnp).*

The step d) implements the MSD-tagset recoverability property previously discussed. The last step of the algorithm transforms the annotation of the training corpus according to each candidate Ctag-set, builds the classifiers, tags the test corpora and evaluates the accuracy. For the examples above, one could eventually get: $Ctag\text{-}AC_i$*=(ASN NSN VP)* and respectively $Ctag\text{-}AC_j$ *= (NPN NSON V3 V2 VN).*

The procedure above does not necessarily lead the designer to an "optimal" Ctag-set. The main reason is that some eliminated attributes, although fully recoverable, if preserved, might act as contextual restrictors for the ambiguity classes in the neighborhood. We figured out such cases by the introspective analysis of the confusion sets, automatically extracted in the training phase (see the credibility profile discussed in the next section) and reintroduced some previously eliminated attributes.

## Combining multiple classifiers

Before the final tagset mapping, optionally, one could use a combined classifier to improve the quality of the Ctag tagging of the text. The basic assumption in trying to combine different classifiers, even of comparable accuracy, is that *they do not make identical errors* (Brill & Wu, 1998; Adda et al 1998). This assumption was confirmed by all the experiments we are aware of.

To make thinks clearer, we should specify that what we call a (basic) classifier is a trained tagger, that is the tagging engine plus the learnt LM. If any of these two components is changed, we speak about another classifier. Thus, if one has at his/her disposal K tagging engines and N training corpora, he/she could construct K*N (basic) classifiers.

Given that each classifier has its own view on the processed text TX (encoded in its underlying LM), it is very unlikely for the k versions of TX to be identical. However, as compared to the *truth* (a human judged annotation), the probability for an arbitrary token from TX to be assigned the correct interpretation in at least one of

the k versions of TX is very high (in general, more than 99%). Let us call the hypothetical guesser of this correct tag an *oracle* (as it is called in (Brill & Wu, 1998)). Implementing an oracle, i.e. automatically deciding which of the k interpretations is the correct one, is a very difficult problem. However, the oracle concept, as defined above, is very useful since its accuracy gives an estimation of the upper bound of correctness that can be achieved by a given classifier combination. The oracle's errors are represented by those cases where no classifier came out with the correct tag.

The experiment described in (v.Halteren et al, 1998) is based on the tagged LOB corpus and uses four different taggers: a 3gram HMM tagger (Steetskamp, 1995), a memory-based tagger (Daelemans et all, 1997), a rule-based tagger (Brill, 1995) and a ME-based tagger (Radnaparkhi, 1996). The oracle's accuracy is estimated at 99.22%. Several decision-making strategies are proposed, out of which the *pair-wise voting* strategy outscored all the individual classifiers (97.92%).

An almost identical point of view and similar results are reported in (Brill & Wu, 1998). Their experiment is based on the Wall Street Journal corpus and uses a HMM 3gram tagger, a rule-based tagger (Brill, 1995) and a ME-based tagger (Radnaparkhi, 1996). In that case, the estimated accuracy of the oracle is 98.59% (apparently, WSJ contains more annotation errors than LOB), and using the *pick-up tagger* combination method, an overall accuracy of 97.2% was obtained.

Our methodology, even though similar at first sight to the ones discussed above, is actually different: instead of using several taggers and the same training data, it uses one tagger (a 3gram-HMM) and trains it on several register-corpora. A new text is independently tagged with each classifier, in as many versions as classifiers are available. These slightly different versions are further combined into the final tagged text. We claim that our approach, which could be based on any particular tagger (ideally, the best one), is more linguistically motivated, as any differences showing up in the output of the individual classifiers are justified only by the linguistic data used in the training. This fact could be used for a rough estimation of the text type/genre/register.

We made experiments with two 3-gram HMM taggers QTAG (Tufiş & Mason, 1998) and TnT (Brants, 1998). Each tagger was separately trained on 4 distinct register-diversified corpora, constructing 4 language models for each tagger. We constructed in this way 8 basic classifiers (2 taggers * 4 training corpora). A new text (unseen, from an unknown register) was independently tagged with each individual classifier. Our combiner interpolated the results provided by the different basic classifiers, with an always better accuracy than that of any basic classifier. On average, the combined classifier made 10.95% fewer errors when compared with the best performing basic classifier and 28,96% fewer errors when compared with the worst performing individual classifier. We checked the statistical significance of the improvement (using McNemar's test) and for all possible pairs, the combined classifier was confirmed to have a different expected behavior from any basic classifier (better).

An interesting finding was that out of the possible classifiers, the best combinations were obtained when the basic classifiers were constructed with the same tagger. The average accuracy of the combined classifier tagger was about 98.5%. We made experiments with various combiners (simple majority, weighted majority voting etc). The best performing one is called *CREDIBILITY*. This combiner is driven by a set of *credibility profiles* (one for each classifier). The $k^{th}$ credibility profile is automatically constructed from evaluating the $k^{th}$ classifier on the text resulted by concatenating all the training corpora. A credibility profile specifies for each tag $T_i$ the probability estimates of recall $R^k(T_i)$ and precision $Pr^k(T_i)$, as well as a confusion set. If $H_{Ti}$ is the number of tags $T_i$ used in the training corpus, $M_{Ti}$ is the number of tags $T_i$ assigned in the machine-tagged corpus, $M_{TiR}$ is the number of rightly assigned $T_i$ tags (out of $M_{Ti}$) then, we define $Pr^k(T_i)$ as $M_{TiR}/M_{Ti}$ and $R^k(T_i)$ as $M_{TiR}/H_{Ti.}$ The confusion set for a tag $T_i$ consists of pairs $<T_j\ P_c^k(T_j|T_i)>$, with $T_j$ a tag that is confused with $T_i$, and $P_c^k(T_j|T_i)$ the probability estimate for such a confusion. The next table displays the entries for adjective tags in such a profile:

| TAG | RECALL | PRECISION | CONFUSION SET |
|---|---|---|---|
| A | 96.02 | 92.51 | R:7.48 |
| AN | 99.73 | 99.73 | NN:0.26 |
| APN | 98.31 | 99.28 | ASN:0.04 ASON:0.08 NPN:0.43 PI:0.01 V2:0.03 V3:0.09 |
| APOY | 100 | 94.23 | NPOY:5.76 |
| APRY | 100 | 97.97 | NPN:0.5 NPRY:1.51 |
| ASN | 96.76 | 95.71 | AN:0.01 M:0.05 NN:0.01 NSN:0.6 NSRN:0.13 PPPD:0.03 R:3.26 S:0.03 V3:0.03 VG:0.01 VP:0.1 |
| ASON | 98.7 | 92.37 | APN:7.45 V3:0.17 |
| ASOY | 100 | 96.12 | NSOY:3.87 |
| ASRY | 99.42 | 97.01 | NSRY:2.99 |
| ASVY | 100 | 90.91 | NSVY:9.09 |

Table 1: Adjectival Entries in a Credibility Profile

The relation (2) describes the *CREDIBILITY* combiner:

$$(2)\ \arg\max_k C^k(T_i) = Pr^k(T_i) - \sum_j Pc^k(T_j \mid Ti) * \beta(Tj)$$

where: $C^k(T_i)$ is the credibility that the $k^{th}$ classifier is right on $T_i$ assignment and $\beta(T_j)$ is 1 or 0 depending whether $T_j$ is assigned or not by a competing classifier. Thus, the $k^{th}$ classifier's credibility with respect to the assignment of a tag $T_i$ represents its precision for the tag $T_i$, decreased with the probability of $T_i$ being confused for a tag $T_j$ proposed by a competing classifier. The winning tag is the one proposed by the classifier with the highest credibility.

## Lexicon, Training Data and Test Data

The Table 2 presents the data content of the main Romanian lexicon that was used for the corpus analysis. An entry is a triple <wordform lemma MSD>. A full account on the lexicon encoding strategies and a thorough analysis on Romanian lexicon and corpus can be found in (Tufiş et al 1997).

| Entries | Items | Lemmas | MSDs |
|---|---|---|---|
| 428042 | 351992 | 41324 | 614 |

Table 2: Lexicon overview

We constructed three training-corpora for different registers (fiction, philosophy and journalism) based on Orwell's *"1984"*, Plato's *"The Republic"* and several issues of the *"România Liberă"* and *"Adevărul"* newspapers. These three corpora (Table 3), cover more than 80% of the MSDs defined in the lexicon, with the remaining ones very implausible to be seen in usual texts (e.g. vocative cliticized adjectives). The catenation of the three basic corpora is further referred to as *Global*.

| Corpus | Occurrences | Items | MSDs |
|---|---|---|---|
| 1984 | 118357 | 15081 | 410 |
| Republic | 135341 | 11002 | 389 |
| News | 98194 | 16672 | 396 |
| Global | 361892 | 29588 | 501 |

Table 3: Training corpora overview

For testing purposes, we hand-tagged about 60.000 additional words from different texts in the three registers:

- "*1994*" is a follow-up for Orwell's famous novel,
- *"barnes"* is a monograph on Aristotle's work and
- *"ziarNou"* is a collection of articles from other newspapers than those included into the News corpora.

In estimating the difficulty of the disambiguation task, one usual measure is the degree of ambiguity. There are three typical ways to compute the ambiguity of a text:

- $AMB_1$ simply counts the number of tags assigned to the items of the text before disambiguation and divides this number to the total number of tokens;
- $AMB_2$ considers only the ambiguous tokens
- *Amb_items* represents the percentage of items that are assigned more than one tag.

Obviously, these ambiguity measures are related:

$$Amb\_items=(AMB_1-1)/(AMB_2-1).$$

The figures in Table 4 represent the three way computed ambiguities for the basic training texts. As one can see, on average, every second word is ambiguous. In reality, for Romanian the average distance between two ambiguous words is longer than in English, but a Romanian ambiguous word, usually, carries more possible interpretations than an ambiguous English one.

| Corpus | Amb_items | $AMB_1$ | $AMB_2$ |
|---|---|---|---|
| 1984 | 38.41% | 1,68 | 2,77 |
| Republic | 42.01% | 1,71 | 2,69 |
| News | 38.17% | 1,71 | 2,86 |

Table 4: Ambiguity in the MSD-annotated corpora

## Evaluation

We mentioned before that the tagger used for the combined language model approach (CLAM) was a 3gram HMM tagger. Initially, this was a slightly modified version of O. Mason's QTAG tagger (available, licence-based, from the author). It uses a local optimization strategy, a sliding 3-word window with the word of interest in the 1st, 2nd and 3rd position respectively. In order to verify that the designed Ctag-set was not over-tuned and biased by the peculiarities of QTAG, we have repeated the same experiments with another 3gram tagger, namely TnT due to T. Brants (available licence-based, from the author). TnT uses the same input/output format but, unlike QTAG, it uses global optimization in tagging (Viterbi algorithm). The evaluation showed different results for the two taggers, but, in both cases, our basic claim was confirmed: TT-CLAM methodology ensured high accuracy in tagging with a large tagset.

The Table 5 shows the results of the evaluation process for 8 single classifiers and 2 combined classifiers (CLAM), built based on the two taggers (QTAG and TnT) and four training corpora (Orwell's "1984", Plato's "The Republic", News and Global). They were run on three texts unseen before, representing chunks of approximately 20,000 words from the previously mentioned 3 test corpora ("1994", "barnes" and "ziarNou").

The column *Text/LM* specifies the chunk of the test text that was tagged by using the classifier based on the *LM* training corpus. The S*ize* column specifies the number of lexical items in the chunk. The $Amb_1$ describes the average ambiguity of the tagged text. The *#unk* column specifies the number of unknown items in the test text. The *Accuracy* and *#errors* columns describe, for each tagger used in building a classifier, the percentage of correct tag assignments (number of

correctly assigned tags versus the number of tags) and the absolute number of errors, respectively. *Accuracy* was computed as 100*(1-#errors/Size). For instance, "1994_20/Republic" is a chunk, containing 20110 items, extracted from "1994" test corpus, tagged with a classifier built by training QTAG and TnT on the "Republic" corpus. The QTAG-based classifier made 425 errors while the TnT-based classifier made 427 errors, thus the accuracy of the two classifiers was 97.88% and 97.87% respectively. As one can see, the CLAM classifier was always the best performer, irrespective of the used tagger and the test text. Table 5 also shows that the classifiers based on the *Global* LM were almost always the second-best ones (except for the text *ziarNou_20*).

This supports the known fact that more training data improves the tagging performance but also sustains our conjecture: *dividing a balanced training corpus into register-specific training corpora and using a combined LM classifier may increase the tagging performance.*

It is worth noting that after error analysis, we found out that the non-Romanian word "qua" occurred in "barnes_20" 81 times, used as a close-class category (conjunction). Similarly, the text "ziarNou" contained 43 occurrences of the item "lu" (used, for stylistic purposes, as a slang form of the pronoun "lui"). The figures in Table 5 show a better performance for the classifiers constructed with TnT. However, when the two anomalous words were normalized ("*qua*" and "*lu*" were defined as aliases for the conjunction "ca" and the pronoun/article "lui", respectively), besides a dramatic accuracy improvement of all classifiers (more than 1,5%), we noticed that the QTAG-based classifiers performed approximately as well as the TnT-based ones. However, the TnT-based CLAM remained superior to the QTAG-based CLAM.

| | | | | QTAG | | TnT | |
|---|---|---|---|---|---|---|---|
| Text/LM | Size | $Amb_1$ | #unk | Accuracy | # errors | Accuracy | # errors |
| 1994_20/CLAM | 20110 | 1.59 | 26 | **98.42** | 318 | **98.45** | 313 |
| 1994_20/Global | | | | 98.32 | 338 | 98.20 | 361 |
| 1994_20/1984 | | | | 98.23 | 356 | 98.08 | 385 |
| 1994_20/Republic | | | | 97.88 | 425 | 97.87 | 427 |
| 1994_20/News | | | | 97.76 | 450 | 97.84 | 433 |
| barnes_20/CLAM | 20120 | 1.64 | 158 | **97.06** | 590 | **97.45** | 512 |
| barnes_20/Global | | | | 96.92 | 620 | 97.15 | 572 |
| barnes_20/News | | | | 96.89 | 624 | 96.96 | 610 |
| barnes_20/1984 | | | | 96.62 | 680 | 96.95 | 613 |
| barnes_20/Republic | | | | 96.56 | 692 | 96.92 | 619 |
| ziarNou_20/CLAM | 20035 | 1.67 | 248 | **97.36** | 527 | **98.33** | 336 |
| ziarNou_20/Global | | | | 97.18 | 564 | 98.30 | 342 |
| ziarNou_20/News | | | | 97.34 | 533 | 98.18 | 365 |
| ziarNou_20/Republic | | | | 96.73 | 655 | 97.94 | 414 |
| ziarNou_20/1984 | | | | 96.50 | 701 | 97.87 | 427 |

Table 5: Evaluation with unknown items in the test data

In a second experiment, we introduced all the previously unknown items in the lexicon, all the possible interpretations being assigned equal lexical probabilities. Table 6 displays the results of this second experiment. Without unknown items, the differences in the accuracy of tagging the same text with the same LM could be explained by the optimization technique used by the two taggers. The experiment supported the idea that global optimization is in general better than the local one, unless too many unknown words are present in the input text. In order to estimate the effect of a wrong guess on the overall tagging accuracy with respect to the global/local optimization strategy, we computed a correlation factor $\mu = 1-N_{PL}/N_{FL}$. $N_{PL}$ represents the number of tags wrongly assigned in a text containing unknown words (partial lexicon) and $N_{FL}$ stands for the number of tags wrongly assigned in the same text, with no unknown words (full lexicon). One should like the value of $\mu$ as low as possible: 0 means either a perfect guesser or a strictly non-propagating error. The table 7 shows the experimental values for the $\mu$ factor. The *OK* column shows, for each classifier and each optimization strategy, the number of correctly tagged tokens out of the total number of unknown ones (*#unk*). Although the percentage of unknown token recovery is better for the GLOBAL strategy than for LOCAL strategy with all classifiers, $\mu_{GLOBAL}$ is always greater than $\mu_{LOCAL,}$ implying that the effect of a wrong guess is reduced in local optimization approach. Put it otherwise, when using a global optimization strategy in tagging, the quality of a very good guesser is of utmost importance.

Another worth making remark concerns the accuracy of the part-of-speech-only disambiguation. Not surprisingly, when evaluating the correctness of POS-only assignment, the accuracy of any classifier (single or combined) was very high (see Table 8).

| | | | | QTAG | | TnT | |
|---|---|---|---|---|---|---|---|
| Text/LM | Size | $Amb_1$ | #unk | Accuracy | # errors | Accuracy | # errors |
| 1994_20/CLAM | 20110 | 1.59 | 0 | **98.71** | 260 | **98.74** | 254 |
| 1994_20/Global | | | | 98.69 | 264 | 98.56 | 289 |
| 1994_20/1984 | | | | 98.54 | 290 | 98.45 | 310 |
| 1994_20/News | | | | 98.15 | 374 | 98.36 | 329 |
| 1994_20/Republic | | | | 98.29 | 342 | 98.30 | 341 |
| barnes_20/CLAM | 20120 | 1.64 | 0 | **99.00** | 203 | **99.11** | 180 |
| barnes_20/Global | | | | 98.64 | 275 | 98.93 | 215 |
| barnes_20/News | | | | 98.57 | 289 | 98.80 | 241 |
| barnes_20/Republic | | | | 98.31 | 340 | 98.73 | 254 |
| barnes_20/1984 | | | | 98.43 | 316 | 98.67 | 266 |
| ziarNou_20/CLAM | 20035 | 1.67 | 0 | **98.88** | 225 | **99.20** | 160 |
| ziarNou_20/News | | | | 98.30 | 341 | 99.14 | 172 |
| ziarNou_20/Global | | | | 98.13 | 376 | 99.05 | 192 |
| ziarNou_20/Republic | | | | 97.77 | 447 | 98.93 | 213 |
| ziarNou_20/1984 | | | | 97.57 | 488 | 98.78 | 244 |

Table 6: Evaluation without unknown items in the test data

| | | GLOBAL | | | | | LOCAL | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Text/LM | # unk | OK | % | $N_{PL}$ | $N_{FL}$ | μ | OK | % | $N_{PL}$ | $N_{FL}$ | μ |
| 1994_20/1984 | 26 | 18 | **0.69** | 385 | 310 | 0,195 | 16 | 0,61 | 356 | 290 | **0.18** |
| 1994_20/Rep | | 18 | **0.69** | 427 | 341 | 0,20 | 17 | 0,65 | 425 | 342 | **0.19** |
| 1994_20/News | | 15 | **0.57** | 433 | 329 | 0,24 | 16 | 0,61 | 450 | 374 | **0.16** |
| barnes_20/1984 | 158 | 104 | **0.65** | 613 | 266 | 0,56 | 66 | 0,41 | 680 | 316 | **0.53** |
| barnes_20/Rep | | 94 | **0.59** | 619 | 254 | 0,59 | 65 | 0,41 | 692 | 340 | **0.50** |
| barnes_20/News | | 101 | **0.63** | 610 | 241 | 0,60 | 70 | 0,44 | 624 | 289 | **0.53** |
| ziarNou_20/1984 | 248 | 191 | **0.77** | 427 | 244 | 0,42 | 130 | 0,52 | 701 | 488 | **0.30** |
| ziarNou_20/Rep | | 181 | **0.72** | 414 | 213 | 0,48 | 131 | 0,52 | 655 | 447 | **0.31** |
| ZiarNou_20/News | | 196 | **0.79** | 336 | 172 | 0,48 | 141 | 0,56 | 533 | 341 | **0.36** |

Table 7: The effect of unknown words and error propagation (non-normalised test data)

| Text/LM | POS Accuracy | # errors |
|---|---|---|
| 1994_20/CLAM | **99.50** | 102 |
| 1994_20/Global | 99.37 | 128 |
| 1994_20/1984 | 99.11 | 179 |
| 1994_20/News | 99.11 | 179 |
| 1994_20/Republic | 99.02 | 199 |
| barnes_20/CLAM | **99.61** | 79 |
| barnes_20/Global | 99.50 | 102 |
| barnes_20/News | 99.22 | 158 |
| barnes_20/1984 | 99.17 | 169 |
| barnes_20/Republic | 99.16 | 171 |
| ZiarNou_20/CLAM | **99.66** | 69 |
| ZiarNou_20/Global | 99.65 | 72 |
| ZiarNou_20/News | 99.46 | 110 |
| ZiarNou_20/Republic | 99.37 | 127 |
| ZiarNou_20/1984 | 99.30 | 141 |

Table 8: Evaluation for POS-only accuracy tagging with unknown items in the test data

This result is much better than what one might obtain if only the part-of-speech information would be retained in the tagset (e.g. 14 tags). In our experiments, the accuracy for POS-tagging with the only 14 tags corresponding to the classes in EAGLES was never better than 93%.

This shows that a proper hidden tagset is needed not only when a large tagset is of interest, but also when a very coarse one is sufficient for a given task. While in the first case (more difficult), the information in the hidden layer of tags is enhanced, in the second case (very simple) the unnecessary information in the hidden layer of tags is left-out by a filter.

## Implementation

The TT-CLAM methodology, described in this paper, is underlying the implementation of a plug-in UNIX environment for tagging, called LINGUASTAT. The tagging process is implemented in a client/server architecture, with the classifiers running in parallel on various servers.

In Figure 1 there is a screen snapshot from a working session with LINGUASTAT. The *Setup* menu (the bottom-right corner window in Figure 1) shows the main components identified by the following tabs: *mtSeg* (a Multext segmenter variant), *qTagger* (QTAG), *Tagging* (TnT), *recover* (the module which maps the Ctags onto MSDs), *ruler* (the rule-based disambiguation), *Long pipe* (a configuration utility for defining a processing pipe), *Combined…* (the LM combined classifier) and *statistics* (several evaluation functions and tests).

LINGUASTAT has also a compare and edit facility (the upper window and the small window below it in Figure 1). The text tagged by various classifiers is displayed in separate windows. The differences among the classifiers are highlighted, so that, by stepping through them (pressing the *Next* button), a human post-editor may validate and correct the final tagging. This way, the extension/creation of a training corpus is sped up at least by an order of magnitude. In (Tufiş, 1998) we showed that the percentage of wrong agreements between classifiers is extremely low (0,59%-0,83%), so that inspecting only the highlighted disagreements (2,85%-3,3%) is quite safe.

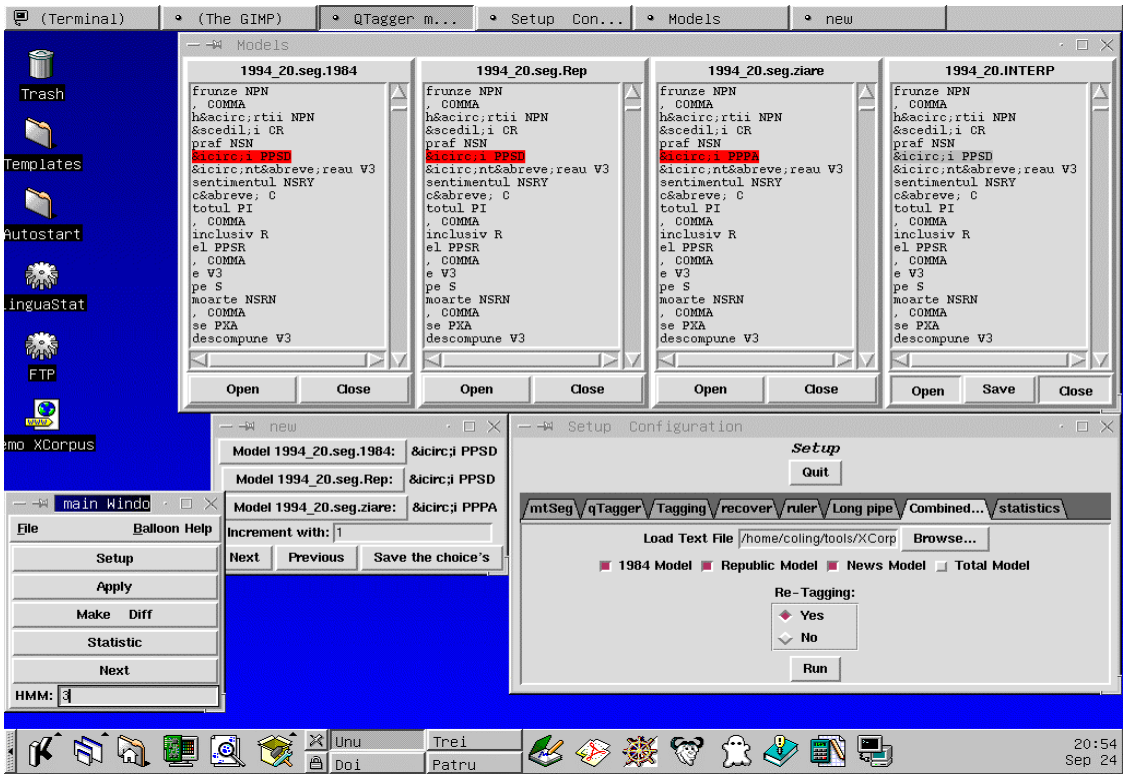


Figure 1: A snapshot from a LINGUASTAT session

Based on the technology described in this paper we implemented a public application for highly accurate (close to 99%) automatic insertion of diacritics into Romanian texts (Tufiş & Chiţu, 1998). The accuracy of the diacritics insertion is superior to the performance of the tagging process since many intra-categorical errors done by the underlying tagger, are harmless for the right decision on inserting or not a diacritical sign.

The language resources and the software are available for research purposes and are distributed by TELRI association through TRACTOR user service.

## Conclusions

The evaluation results show several important things:

- tiered tagging (tagging a text by a hidden tagset layer followed by the recovery of the information in the initial tagset) allows for successful use of statistical methods in the processing of highly inflectional languages (in our case Romanian);
- the methodology we presented is not language dependent. The results described in (Varadi, 1999) and (Tufiş et al, 2000) support this claim with experiments on Hungarian, a language very different from Romanian.
- combining register-diversified LMs, significantly increases the accuracy of the tagging process. Splitting a balanced training corpus into specialized register training corpora is worth considering: even a simple combiner as MAJORITY ensures a better result than using only the LM of the initial balanced corpus;

- the high level of correct agreements and the negligible percentage of false agreements can help in fast and cheap development of large training corpora. The human expert annotator can concentrate quite safely on the disagreement cases; with an average 3% (in our experiments) of tag disagreement, the hand validation of large training corpora is a manageable task. It is worth observing that, in our experiments, we have not observed any instance of disagreement between the 4 classifiers where the right tag was not proposed by at least one of them.

## Acknowledgements

The work reported here built on the main results of the Multext-East European project (COP106/1995) and was partly funded by the Romanian Academy grant (GAR#188/1997).

## References

Adda, G., Mariani, J., Lecompte, J., Paroubek, P., Rajman, M. "The GRACE French Part-of-Speech Tagging Evaluation Task".*In Proceedings of First International Conference on Language Resources and Evaluation*, Granada, Spain, 1998, 433-441.

Berger, A.L., Della Pietra, S.A., Della Pietra, V.J. "A Maximum Entropy Approach to Natural Language Processing" In *Computational Linguistics*, vol. 22, no. 1, March 1996, 39-72.

Brants, Th. "TnT- A Statistical Part-of-Speech Tagger" Instalation and User Guide, University of Saarland, Computational Linguistics, March 1998.

Brill, E., Wu, J. "Classifier Combination for Improved Lexical Disambiguation" *In Proceedings of COLING-ACL'98* Montreal, Canada, 1998, pp. 191-195.

Brill, E. "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging." *Computational Linguistics*, 21(4), 1995, pp. 543-565.

Daelemans, W., Zavrel, J., Berck, P., Gillis, S, "A Memory-Based Part-of-Speech Tagger Generator. In *Proceedings of 4th Workshop on Very Large Corpora*, Copenhagen, Denmark, 1997.

Dermatas, E., Kokkinakis, G. "Automatic Stochastic Tagging of Natural Language Texts." *Computational Linguistics*, 21(2), 1995. 321-350.

Dietterich, T. "Machine Learning Research: Four Current Directions", *In AI Magazine*, Winter, 1997, pp. 97-136.

Dimitrova,L., Erjavec,T., Ide, N., Kaalep, H., Petkevic,V., D. Tufiş: "Multext-East: Parallel and Comparable Corpora and Lexicons for Six Central and Eastern European Languages", *In Proceedings of COLING-ACL'98*, Montreal, Canada, 315-319.

Erjavec, T. and Monachini, M. (eds.). "Specifications and Notation for Lexicon Encoding". COP Project 106 Multext-East,WP1 - Task 1.1 Deliverable D1.1 F (Final Report ), 17 December 1997.

v. Halteren, H., Zavrel, J., Daelemans, W. "Improving Data Driven Wordclass Tagging by System Combination" *In Proceedings of COLING-ACL'98*, Montreal, Canada, 1998, pp. 491-497.

Padró, L., Márquez, L. "On the Evaluation and Comparison of Taggers: the Effect of Noise in Testing Corpora." *In Proceedings of COLING-ACL'98.* Montreal, Canada, 1998, 997-1002.

Rathaparkhi, A. "A Maximum Entropy Part of Speech Tagger." In *Proceedings of EMNLP'96*, Philadelphia, Pennsylvania, 1996.

Steetskamp, R. "An implementation of a probabilistic tagger" Master's Thesis, TOSCA Research Group, University of Nijmegen, 1995.

Tufiş, D. "Tiered Tagging and Combined Language Models Classifiers". In Jelinek, F., Nöth, E. (eds.) "*Text, Speech and Dialogue*", Lecture Notes in Artificial Intelligence 1692, Springer, 1999, pp. 28-33.

Tufiş, D., Chiţu, A. "Automatic Insertion of Diacritics in Romanian Texts". In Proceedings of the 5th Workshop on Computational Lexicography COMPLEX, Pecs, Hungary, 1999, 195-194.

Tufiş, D., Barbu, A.M., Pătraşcu, V., Rotariu, G., Popescu, C. "Corpora and Corpus-Based Morpho-Lexical Processing" in Dan Tufiş, Poul Andersen (eds.) *Recent Advances in Romanian Language Technology*, Editura Academiei, 1997, pp.35-56 (also available at http://www.racai.ro/books).

Tufiş, D. "Tiered Tagging". *Research Report no. 32*, June 1998, RACAI, Bucharest, 72p (in Romanian).

Tufiş, D., Mason, O. "Tagging Romanian Texts: a Case Study for QTAG, a Language Independent Probabilistic Tagger" *In Proceedings of First International Conference on Language Resources and Evaluation*, Granada, Spain, 1998, pp. 589-596.

Tufiş, D., Ide, N., Erjavec, T. "Standardised Specifications, Development and Assessment of Large Morpho-Lexical Resources for Six Central and Eastern European Languages". In *Proceedings of First International Conference on Language Resources and Evaluation*, Granada, Spain, 1998, pp. 233-240.

Tufiş, D., Dienes, P., Oravecz, C., Váradi T. "Principled Hidden Tagset Design for Tiered Tagging of Hungarian". In *Proceedings of Second International Conference on Language Resources and Evaluation*, Athens, Greece, 2000, *this volume*.

Váradi, T. "Morpho-syntactic Ambiguity and Tagset Design for Hungarian". In *Proceeding of the EACL Workshop on Annotated Corpora"*, Bergen, Norway, 1999.