Statistical Morphological Disambiguation for Agglutinative Languages

Dilek Z. Hakkani-Tür, Kemal Oflazer, Gökhan Tür
Department of Computer Engineering,
Bilkent University,
Ankara, 06533, TURKEY
{hakkani,ko,tur}@cs.bilkent.edu.tr

Abstract

In this paper, we present statistical models for morphological disambiguation in Turkish. Turkish presents an interesting problem for statistical models since the *potential* tag set size is very large because of the productive derivational morphology. We propose to handle this by breaking up the morhosyntactic tags into inflectional groups, each of which contains the inflectional features for each (intermediate) derived form. Our statistical models score the probability of each morhosyntactic tag by considering statistics over the individual inflection groups in a trigram model. Among the three models that we have developed and tested, the simplest model ignoring the local morphotactics within words performs the best. Our best trigram model performs with 93.95% accuracy on our test data getting all the morhosyntactic and semantic features correct. If we are just interested in syntactically relevant features and ignore a very small set of semantic features, then the accuracy increases to 95.07%.

1 Introduction

Recent advances in computer hardware and availability of very large corpora have made the application of statistical techniques to natural language processing a feasible and a very appealing research area. Many useful results have been obtained by applying these techniques to English (and similar languages) – in parsing, word sense disambiguation, part-of-speech (POS) tagging, speech recognition, etc. However, languages like Turkish, Czech, Hungarian and Finnish, display a substantially different behavior than English. Unlike English, these languages have agglutinative or inflective morphology and relatively free constituent order. Such languages have received little previous attention in statistical processing.

In this paper, we present our work on modeling Turkish using statistical methods, and present results on morphological disambiguation. The methods developed here are certainly applicable to other agglutinative languages, especially those involving productive derivational phenomena. The paper is organized as follows: After a brief overview of related previous work, we summarize relevant aspects of Turkish and present details of various statistical models for morphological disambiguation for Turkish. We then present results and analyses from our experiments.

2 Related Work

There has been a large number of studies in tagging and morphological disambiguation using various techniques. POS tagging systems have used either a statistical or a rule-based approach. In the statistical approach, a large corpus has been used to train a probabilistic model which then has been used to tag new text, assigning the most likely tag for a given word in a given context (e.g., Church (1988), Cutting et al. (1992)). In the rule-based approach, a large number of hand-crafted linguistic constraints are used to eliminate impossible tags or morphological parses for a given word in a given context (Karlsson et al., 1995). Brill (1995a) has presented a transformation-based learning approach, which induces disambiguation rules from tagged corpora.

Morphological disambiguation in inflecting or agglutinative languages with complex morphology involves more than determining the major or minor parts-of-speech of the lexical items. Typically, morphology marks a number of inflectional or derivational features and this involves ambiguity. For instance, a given word may be chopped up in different ways into morphemes, a given morpheme may mark different features depending on the morphotactics, or lexicalized variants of derived words may interact with productively derived versions (see Oflazer and Tür (1997) for the different kinds of morphological ambiguities in Turkish.) We assume that *all syntactically relevant features* of word forms have to be determined correctly for morphological disambiguation.

In this context, there have been some interesting previous studies for different languages. Levinger et al. (1995) have reported on an approach that learns morpholexical probabilities from an untagged corpus and have used the resulting information in

morphological disambiguation in Hebrew. Hajič and Hladká (1998) have used maximum entropy modeling approach for morphological disambiguation in Czech. Ezeiza et al. (1998) have combined stochastic and rule-based disambiguation methods for Basque. Megyesi (1999) has adapted Brill's POS tagger with extended lexical templates to Hungarian.

Previous approaches to morphological disambiguation of Turkish text had employed a constraint-based approach (Oflazer and Kuruöz, 1994; Oflazer and Tür, 1996; Oflazer and Tür, 1997). Although results obtained earlier in these approaches were reasonable, the fact that the constraint rules were hand crafted posed a rather serious impediment to the generality and improvement of these systems.

3 Turkish

Turkish is a free constituent order language. The order of the constituents may change freely according to the discourse context and the syntactic role of the constituents is indicated by their case marking. Turkish has agglutinative morphology with productive inflectional and derivational suffixations. The number of word forms one can derive from a Turkish root form may be in the millions (Hankamer, 1989). Hence, the number of distinct word forms, i.e., the vocabulary size, can be very large. For instance, Table 1 shows the size of the vocabulary for 1 and 10 million word corpora of Turkish, collected from online newspapers. This large vocabulary is the reason

Corpus size	Vocabulary size
1M words	106,547
10M words	417,775

Table 1: Vocabulary sizes for two Turkish corpora.

for a serious data sparseness problem and also significantly increases the number of parameters to be estimated even for a bigram language model. The size of the vocabulary also causes the perplexity to be large (although this is not an issue in morphological disambiguation). Table 2 lists the training and test set perplexities of trigram language models trained on 1 and 10 million word corpora for Turkish. For each corpus, the first column is the perplexity for the data the language model is trained on, and the second column is the perplexity for previously unseen test data of 1 million words. Another major reason for the high perplexity of Turkish is the high percentage of out-of-vocabulary words (words in the test data which did not occur in the training data); this results from the productivity of the word formation process.

Training Data	Training Set Perplexity	Test Set (1M words) Perplexity
1M words	66.13	1449.81
10M words	94.08	1084.13

Table 2: The perplexity of Turkish corpora using word-based trigram language models.

The issue of large vocabulary brought in by productive inflectional and derivational processes also makes tagset design an important issue. In languages like English, the number of POS tags that can be assigned to the words in a text is rather limited (less than 100, though some researchers have used large tag sets to refine granularity, but they are still small compared to Turkish.) But, such a finite tagset approach for languages like Turkish may lead to an inevitable loss of information. The reason for this is that the morphological features of intermediate derivations can contain markers for syntactic relationships. Thus, leaving out this information within a fixed-tagset scheme may prevent crucial syntactic information from being represented (Oflazer et al., 1999). For example, it is not clear what POS tag should be assigned to the word *sağlamlaştırmak* (below), without losing any information, the category of the root (Adjective), the final category of the word as a whole (Noun) or one of the intermediate categories (Verb).[1]

sağlam+laş+tır+mak
sağlam+Adj^DB+Verb+Become^DB
+Verb+Caus+Pos^DB+Noun+Inf+A3sg+Pnon+Nom
to cause (something) to become strong / to strengthen/fortify (something)

Ignoring the fact that the root word is an adjective may sever any relationships with an adverbial modifier modifying the root. Thus instead of a simple POS tag, *we use the full morphological analyses of the words, represented as a combination of features (including any derivational markers) as their morphosyntactic tags.* For instance in the example above, we would use everything including the root form as the morphosyntactic tag.

In order to alleviate the data sparseness problem we break down the full tags. We represent each word as a sequence of *inflectional groups* (IGs hereafter), separated by ^DBs denoting derivation boundaries, as described by Oflazer (1999). Thus a morphological parse would be represented in the following general form:

[1] The morphological features other than the POSs are: +Become: become verb, +Caus: causative verb, +Pos: Positive polarity, +Inf: marker that derives an infinitive form from a verb, +A3sg: 3sg number-person agreement, +Pnon: No possessive agreement, and +Nom: Nominative case. ^DB's mark derivational boundaries.

	Possible	Observed
Full Tags (No roots)	∞	10,531
Inflectional Groups	9,129	2,194

Table 3: Numbers of Tags and IGs

$$\texttt{root+IG}_1\texttt{\^{}DB+IG}_2\texttt{\^{}DB+}\cdots\texttt{\^{}DB+IG}_n$$

where $\texttt{IG}_i$ denotes relevant inflectional features of the inflectional groups, including the part-of-speech for the root or any of the derived forms.

For example, the infinitive form *sağlamlaştırmak* given above would be represented with the adjective reading of the root *sağlam* and the following 4 IGs:

1. `Adj`
2. `Verb+Become`
3. `Verb+Caus+Pos`
4. `Noun+Inf+A3sg+Pnon+Nom`

Table 3 provides a comparison of the number distinct full morphosyntactic tags (ignoring the root words in this case) and IGs, generatively possible and observed in a corpus of 1M words (considering all ambiguities). One can see that the number observed full tags ignoring the root words is very high, significantly higher than quoted for Czech by Hajič and Hladká (1998).

4 Statistical Morphological Disambiguation

Morphological disambiguation is the problem of finding the corresponding sequence of morphological parses (including the root), $T = t_1^n = t_1, t_2, ..., t_n$, given a sequence of words $W = w_1^n = w_1, w_2, ..., w_n$. Our approach is to model the distribution of morphological parses given the words, using a hidden Markov model, and then to seek the variable T, that maximizes $P(T|W)$:

$$\begin{aligned} \operatorname*{argmax}_T P(T|W) &= \operatorname*{argmax}_T \frac{P(T) \times P(W|T)}{P(W)} \quad (1) \\ &= \operatorname*{argmax}_T P(T) \times P(W|T) \quad (2) \end{aligned}$$

The term $P(W)$ is a constant for all choices of T, and can thus be ignored when choosing the most probable T. We can further simplify the problem using the assumption that words are independent of each other given their tags. In Turkish we can use the additional simplification that $P(w_i|t_i) = 1$ since t_i includes the root form and all morphosyntactic features to uniquely determine the word form.[2] Since in our case $P(w_i|t_1^n) = P(w_i|t_i) = 1$, we can write:

$$P(W|T) = \prod_{i=1}^{n} P(w_i|t_1^n) = 1$$

and

$$\operatorname*{argmax}_T P(T|W) = \operatorname*{argmax}_T P(T) \quad (3)$$

Now,

$$\begin{aligned} P(T) &= P(t_n|t_1^{n-1}) \times P(t_{n-1}|t_1^{n-2}) \times \ldots \\ &\quad \times P(t_2|t_1) \times P(t_1) \end{aligned}$$

Simplifying further with the trigram tag model, we get:

$$\begin{aligned} P(T) &= P(t_n|t_{n-2}, t_{n-1}) \times \\ &\quad P(t_{n-1}|t_{n-3}, t_{n-2}) \times \ldots \\ &\quad P(t_3|t_1, t_2) \times P(t_2|t_1) \times P(t_1) \\ &= \prod_{i=1}^{n} P(t_i|t_{i-2}, t_{i-1}) \quad (4) \end{aligned}$$

where we define $P(t_1|t_{-1}, t_0) = P(t_1)$, $P(t_2|t_0, t_1) = P(t_2|t_1)$ to simplify the notation.

If we consider morphological analyses as a sequence of root and IGs, each parse t_i can be represented as $(r_i, IG_{i,1}, \ldots, IG_{i,n_i})$, where n_i is the number of IGs in the i^{th} word.[3] This representation changes the problem as shown in Figure 1 where the chain rule has been used to factor out the individual components.

This formulation still suffers from the data sparseness problem. To alleviate this, we make the following simplifying assumptions:

1. A root word depends only on the roots of the previous words, and is independent of the inflectional and derivational productions on them:

$$\begin{aligned} P(r_i|(r_{i-2}, IG_{i-2,1}, \ldots, IG_{i-2,n_{i-2}}), \\ (r_{i-1}, IG_{i-1,1}, \ldots, IG_{i-1,n_{i-1}})) = \\ P(r_i|r_{i-2}, r_{i-1}) \quad (5) \end{aligned}$$

   The intention here is that this will be useful in the disambiguation of the root word when a given form has morphological parses with different root words. So, for instance, for disambiguating the surface form *adam* with the following two parses:

[2] That is, we assume that there is no morphological generation ambiguity. This is almost always true. There are a few word forms like *gelirkene* and *nerde*, which have the same morphological parses with the word forms *gelirken* and *nerede*, respectively but are pronounced (and written) slightly differently. These are rarely seen in written texts, and can thus be ignored.

[3] In our training and test data, the number of IGs in a word form is on the average 1.6, therefore, n_i is usually 1 or 2. We have seen, occasionally, word forms with 5 or 6 inflectional groups.

$$
\begin{aligned}
P(t_i|t_1^{i-1}) &= P(t_i|t_{i-2},t_{i-1}) \\
&= P((r_i, IG_{i,1} \ldots IG_{i,n_i})|(r_{i-2}, IG_{i-2,1} \ldots IG_{i-2,n_{i-2}}),(r_{i-1}, IG_{i-1,1} \ldots IG_{i-1,n_{i-1}})) \\
&= P(r_i|(r_{i-2}, IG_{i-2,1} \ldots IG_{i-2,n_{i-2}}),(r_{i-1}, IG_{i-1,1} \ldots IG_{i-1,n_{i-1}})) \times \\
&\quad P(IG_{i,1}|(r_{i-2}, IG_{i-2,1} \ldots IG_{i-2,n_{i-2}}),(r_{i-1}, IG_{i-1,1} \ldots IG_{i-1,n_{i-1}}), r_i) \times \\
&\quad \ldots \times \\
&\quad P(IG_{i,n_i}|(r_{i-2}, IG_{i-2,1} \ldots IG_{i-2,n_{i-2}}),(r_{i-1}, IG_{i-1,1} \ldots IG_{i-1,n_{i-1}}), r_i, IG_{i,1}, \ldots, IG_{i,n_i-1})
\end{aligned}
$$

Figure 1: Equation for morphological disambiguation when tags are decomposed into inflectional groups.

(a) `adam+Noun+A3sg+Pnon+Nom` (*man*)

(b) `ada+Noun+A3sg+P1sg+Nom` (*my island*)

in the noun phrase *kırmızı kazaklı adam* (*the man with a red sweater*), only the roots (along with the part-of-speech of the root) of the previous words will be used to select the right root. Note that the selection of the root has some impact on what the next IG in the word is, but we assume that IGs are determined by the syntactic context and not by the root.

2. An interesting observation that we can make about Turkish is that when a word is considered as a sequence of IGs, syntactic relations are between the last IG of a (dependent) word and with some (including the last) IG of the (head) word on the right (with minor exceptions) (Oflazer, 1999).

Based on these assumptions and the equation in Figure 1, we define three models, all of which are based on word level trigrams:

1. **Model 1:** The presence of IGs in a word only depends on the final IGs of the previous words. This model ignores any morphotactical relation between an IG and any previous IG in the same word.
2. **Model 2:** The presence of IGs in a word only depends on the final IGs of the previous words and the previous IG in the same word. In this model, we consider morphotactical relations and assume that an IG (except the first one) in a word form has some dependency on the previous IG. Given that on the average a word has about 1.6 IGs, IG bigrams should be sufficient.
3. **Model 3:** This is the same as Model 2, except that the dependence with the previous IG in a word is assumed to be independent of the dependence on the final IGs of the previous words. This allows the formulation to separate the contributions of the morphotactics and syntax.

The equations for these models are shown in Figure 2. We also have built a baseline model based on the standard definition of the tagging problem in Equation 2. For the baseline, we have assumed that the part of the morphological analysis after the root word is the tag in the conventional sense (and the assumption that $P(w_i|t_i) = 1$ no longer holds).

5 Experiments and Results

To evaluate our models, we first trained our models and then tried to morphologically disambiguate our test data. For statistical modeling we used SRILM – the SRI language modeling toolkit (Stolcke, 1999).

Both the test data and training data were collected from the web resources of a Turkish daily newspaper. The tokens were analyzed using the morphological analyzer, developed by Oflazer (1994). The ambiguity of the training data was then reduced from 1.75 to 1.55 using a preprocessor, that disambiguates lexicalized and non-lexicalized collocations and removes certain obviously impossible parses, and tries to analyze unknown words with an unknown word processor. The training data consists of the unambiguous sequences (US) consisting of about 650K tokens in a corpus of 1 million tokens, and two sets of manually disambiguated corpora of 12,000 and 20,000 tokens. The idea of using unambiguous sequences is similar to Brill's work on unsupervised learning of disambiguation rules for POS tagging (1995b).

The test data consists of 2763 tokens, 935 (≈34%) of which have more than one morphological analysis after preprocessing. The ambiguity of the test data was reduced from 1.74 to 1.53 after preprocessing.

As our evaluation metric, we used accuracy defined as follows:

$$accuracy = \frac{\#\ of\ correct\ parses}{\#\ of\ tokens} \times 100$$

The accuracy results are given in Table 4. For all cases, our models performed better than baseline tag model. As expected, the tag model suffered considerably from data sparseness. Using all of our training data, we achieved an accuracy of 93.95%, which is 2.57% points better than the tag model trained using the same amount of data. Models 2 and 3 gave

In all three models we assume that roots and IGs are independent.

Model 1: This model assumes that an IG in a word depends on the last IGs of the two previous words.

$$P(IG_{i,k}|(r_{i-2}, IG_{i-2,1}...IG_{i-2,n_{i-2}}), (r_{i-1}, IG_{i-1,1}, \ldots, IG_{i-1,n_{i-1}}), r_i, IG_{i,1}, \ldots, IG_{i,k-1}) = P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}})$$

Therefore,

$$P(t_i|t_{i-2}, t_{i-1}) = P(r_i|r_{i-2}, r_{i-1}) \times \prod_{k=1}^{n_i} P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}) \quad (6)$$

Model 2: The model assumes that in addition to the dependencies in Model 1, an IG also depends on the **previous IG** in the same word.

$$P(IG_{i,k}|(r_{i-2}, IG_{i-2,1}...IG_{i-2,n_{i-2}}), (r_{i-1}, IG_{i-1,1}, \ldots, IG_{i-1,n_{i-1}}), r_i, IG_{i,1}, \ldots, IG_{i,k-1}) = P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}, \mathbf{IG_{i,k-1}})$$

Therefore,

$$P(t_i|t_{i-2}, t_{i-1}) = P(r_i|r_{i-2}, r_{i-1}) \times \prod_{k=1}^{n_i} P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}, \mathbf{IG_{i,k-1}}) \quad (7)$$

Model 3: This is same as Model 2, except the morphotactic and syntactic dependencies are considered to be independent.

$$P(IG_{i,k}|(r_{i-2}, IG_{i-2,1}...IG_{i-2,n_{i-2}}), (r_{i-1}, IG_{i-1,1}, \ldots, IG_{i-1,n_{i-1}}), r_i, IG_{i,1}, \ldots, IG_{i,k-1}) = P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}, IG_{i,k-1})$$

$$P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}, IG_{i,k-1}) = P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}) \times \frac{P(IG_{i,k}|IG_{i,k-1})}{P(IG_{i,k})}$$

Therefore,

$$P(t_i|t_{i-2}, t_{i-1}) = P(r_i|r_{i-2}, r_{i-1}) \times \prod_{k=1}^{n_i} \left(P(IG_{i,k}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}) \times \frac{P(IG_{i,k}|IG_{i,k-1})}{P(IG_{i,k})} \right) \quad (8)$$

In order to simplify the notation, we have defined the following:

$$P(r_1|r_{-1}, r_0) = P(r_1) \qquad P(IG_{1,k}|IG_{-1,n_{-1}}, IG_{0,n_0}) = P(IG_{1,k})$$
$$P(r_2|r_0, r_1) = P(r_2|r_1) \qquad P(IG_{2,l}|IG_{0,n_0}, IG_{1,n_1}) = P(IG_{2,l}|IG_{1,n_1})$$

$$P(IG_{i,1}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}}, IG_{i,0}) = P(IG_{i,1}|IG_{i-2,n_{i-2}}, IG_{i-1,n_{i-1}})$$
$$P(IG_{1,k}|IG_{-1,n_{-1}}, IG_{0,n_0}, IG_{1,k-1}) = P(IG_{1,k}|IG_{1,k-1})$$
$$P(IG_{2,l}|IG_{0,n_0}, IG_{1,n_1}, IG_{2,l-1}) = P(IG_{2,l}|IG_{1,n_1}, IG_{2,l-1})$$

$$P(IG_{2,1}|IG_{1,n_1}, IG_{2,0}) = P(IG_{2,1}|IG_{1,n_1})$$
$$P(IG_{i,1}|IG_{i,0}) = P(IG_{i,1})$$

for $k = 1, 2, ..., n_1$, $l = 1, 2, ..., n_2$, and $i = 1, 2, ..., n$.

Figure 2: Equations for Models 1, 2, and 3.

Training Data	Tag Model (Baseline)	Model 1	Model 1 (Bigram)	Model 2	Model 3
Unambiguous sequences (US)	86.75%	88.21%	89.06%	87.01%	87.19%
US + 12,000 words	91.34%	93.52%	93.34%	92.43%	92.72%
US + 32,000 words	91.34%	93.95%	93.56%	92.87%	92.94%

Table 4: Accuracy results for different models.

similar results, Model 2 suffered from data sparseness slightly more than Model 3, as expected.

Surprisingly, the bigram version of Model 1 (i.e., Equation (7), but with bigrams in root and IG models), also performs quite well. If we consider just the syntactically relevant morphological features and ignore any semantic features that we mark in morphology, the accuracy increases a bit further. These stem from two properties of Turkish: Most Turkish root words also have a proper noun reading, when written with the first letter capitalized.[4] We count it as an error if the tagger does not get the correct proper noun marking, for a proper noun. But this is usually impossible especially at the beginning of sentences where the tagger can not exploit capitalization and has to back-off to a lower-order model. In almost all of such cases, all syntactically relevant morphosyntactic features except the proper noun marking are actually correct. Another important case is the pronoun *o*, which has both personal pronoun (s/he) and demonstrative pronoun readings (it) (in addition to a syntactically distinct determiner reading (that)). Resolution of this is always by semantic considerations. When we count as correct any errors involving such semantic marker cases, we get an accuracy of 95.07% with the best case (cf. 93.91% of the Model 1). This is slightly better than the precision figures that is reported earlier on morphological disambiguation of Turkish using constraint-based techniques (Oflazer and Tür, 1997). Our results are slightly better than the results on Czech of Hajič and Hladká (1998). Megyesi (1999) reports a 95.53% accuracy on Hungarian (a language whose features relevant to this task are very close to those of Turkish), with just the POS tags being correct. In our model this corresponds to the root and the POS tag of the last IG being correct and the accuracy of our best model with this assumption is 96.07%. When POS tags and subtags are considered, the reported accuracy for Hungarian is 91.94% while the corresponding accuracy in our case is 95.07%. We can also note that the results presented by Ezeiza et al. (1998) for Basque are better than ours. The main reason for this is that they employ a much more sophisticated (compared to our preprocessor) constraint-grammar based system which improves precision without reducing recall. Statistical techniques applied after this disambiguation yield a better accuracy compared to starting from a more ambiguous initial state.

[4] In fact, any word form is a potential first name or a last name.

Since our models assumed that we have independent models for disambiguating the root words, and the IGs, we ran experiments to see the contribution of the individual models. Table 5 summarizes the accuracy results of the individual models for the best case (Model 1 in Table 4.)

Model	Accuracy
IG Model	92.08%
Root Model	80.36%
Combined Model	93.95%

Table 5: The contribution of the individual models for the best case.

There are quite a number of classes of words which are always ambiguous and the preprocessing that we have employed in creating the unambiguous sequences can never resolve these cases. Thus statistical models trained using only the unambiguous sequences as the training data do not handle these ambiguous cases at all. This is why the accuracy results with only unambiguous sequences are significantly lower (row 1 in Table 4). The manually disambiguated training sets have such ambiguities resolved, so those models perform much better.

An analysis of the errors indicates the following: In 15% of the errors, the last IG of the word is incorrect but the root and the rest of the IGs, if any, are correct. In 3% of the errors, the last IG of the word is correct but the either the root or some of the previous IGs are incorrect. In 82% of the errors, neither the last IG nor any of the previous IGs are correct. Along a different dimension, in about 51% of the errors, the root and its part-of-speech are not determined correctly, while in 84% of the errors, the root and the first IG combination is not correctly determined.

6 Conclusions

We have presented an approach to statistical modeling for agglutinative languages, especially those having productive derivational phenomena. Our approach essentially involves breaking up the full morphological analysis across derivational boundaries and treating the components as subtags, and then determining the correct sequence of tags via statistical techniques. This, to our knowledge, is the first detailed attempt in statistical modeling of agglutinative languages and can certainly be applied to other such languages like Hungarian and Finnish with productive derivational morphology.

7 Acknowledgments

We thank Andreas Stolcke of SRI STAR Lab for providing us with the language modeling toolkit and for very helpful discussions on this work. Liz Shriberg of SRI STAR Labs, and Bilge Say of Middle East Technical University Informatics Institute, provided helpful insights and comments.

References

Eric Brill. 1995a. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–566, December.

Eric Brill. 1995b. Unsupervised learning of disambiguation rules for part of speech tagging. In *Proceedings of the Third Workshop on Very Large Corpora*, Cambridge, MA, June.

Kenneth W. Church. 1988. A stochastic parts program and a noun phrase parser for unrestricted text. In *Proceedings of the Second Conference on Applied Natural Language Processing*, Austin, Texas.

Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, Trento, Italy.

N. Ezeiza, I. Alegria, J. M. Arriola, R. Urizar, and I. Aduriz. 1998. Combining stochastic and rule-based methods for disambiguation in agglutinative languages. In *Proceedings of the* 36^{th} *Annual Meeting of the Association for Computational Linguistics and* 17^{th} *International Conference on Computational Linguistics*, pages 379–384, Montreal, Quebec, Canada, August.

Jan Hajič and Barbora Hladká. 1998. Tagging inflective languages: Prediction of morphological categories for a rich, structured tagset. In *Proceedings of COLING/ACL'98*, pages 483–490, Montreal, Canada, August.

Jorge Hankamer, 1989. *Lexical Representation and Process*, chapter Morphological Parsing and the Lexicon. The MIT Press.

Fred Karlsson, Atro Voutilainen, Juha Heikkilä, and Arto Anttila. 1995. *Constraint Grammar-A Language–Independent System for Parsing Unrestricted Text*. Mouton de Gruyter.

Moshe Levinger, Uzzi Ornan, and Alon Itai. 1995. Learning morpho-lexical probabilities from an untagged corpus with an application to Hebrew. *Computational Linguistics*, 21(3):383–404, September.

Beáta Megyesi. 1999. Improving Brill's POS tagger for an agglutinative language. In Pascale Fung and Joe Zhou, editors, *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 275–284, College Park, Maryland, USA, June.

Kemal Oflazer and İlker Kuruöz. 1994. Tagging and morphological disambiguation of Turkish text. In *Proceedings of the* 4^{th} *Applied Natural Language Processing Conference*, pages 144–149. ACL, October.

Kemal Oflazer and Gökhan Tür. 1996. Combining hand-crafted rules and unsupervised learning in constraint-based morphological disambiguation. In Eric Brill and Kenneth Church, editors, *Proceedings of the ACL-SIGDAT Conference on Empirical Methods in Natural Language Processing*.

Kemal Oflazer and Gökhan Tür. 1997. Morphological disambiguation by voting constraints. In *Proceedings of the* 35^{th} *Annual Meeting of the Association for Computational Linguistics (ACL'97/EACL'97)*, Madrid, Spain, July.

Kemal Oflazer, Dilek Z. Hakkani-Tür, and Gökhan Tür. 1999. Design for a Turkish treebank. In *Proceedings of Workshop on Linguistically Interpreted Corpora, at EACL'99*, Bergen, Norway, June.

Kemal Oflazer. 1994. Two-level description of Turkish morphology. *Literary and Linguistic Computing*, 9(2):137–148.

Kemal Oflazer. 1999. Dependency parsing with an extended finite state approach. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, College Park, Maryland, June.

Andreas Stolcke. 1999. SRILM—the SRI language modeling toolkit. `http://www.speech.sri.com/projects/srilm/`.