Computers and the Humanities **36:** 381–410, 2002.

Statistical Morphological Disambiguation for Agglutinative Languages ⋆

DILEK Z. HAKKANI-TÜR[1]*, KEMAL OFLAZER[2] and GÖKHAN TÜR[3]
[1]*AT&T Labs-Research, 180 Park Avenue, Florham Park, NJ 07932, USA*
E-mail: dtur@research.att.com
[2]*Faculty of Engineering and Natural Sciences, Sabancı University, Orhanlı, Tuzla, 81474, Istanbul, Turkey*
E-mail: oflazer@sabanciuniv.edu
[3]*AT&T Labs-Research, 180 Park Avenue, Florham Park, NJ 07932, USA*
E-mail: gtur@research.att.com
(**author for correspondence)*

Abstract. We present statistical models for morphological disambiguation in agglutinative languages, with a specific application to Turkish. Turkish presents an interesting problem for statistical models as 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 inflectional groups and surface roots in trigram models. Among the four 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%.

Key words: agglutinative languages, morphological disambiguation, *n*-gram language models, statistical natural language processing, Turkish

1. Introduction

Many useful results have been obtained by applying statistical language modeling techniques to English (and similar languages) – in parsing, word sense disambiguation, part-of-speech tagging, speech recognition, etc. However, languages which display a substantially different behavior than English, like Turkish, Czech, Hungarian (in that, they have agglutinative or inflectional morphology and relatively free constituent order) have not been studied extensively using statistical approaches, compared to, for instance, English.

⋆ This work was done while the first and third authors were PhD students at Bilkent University, Ankara, Turkey.

1. masaca	10. masalık	18. masaya	26. masamız
2. masacasına	11. masanın	19. masaydı	27. masamsı
3. masacı	12. masası	20. masayı	28. masan
4. masalaş	13. masayken	21. masayız	29. masanız
5. masalan	14. masaykene	22. masayım	30. masadan
6. masalar	15. masayla	23. masada	31. masasal
7. masaları	16. masaymış	24. masadır	32. masasın
8. masasız	17. masaysa	25. masam	33. masasınız
9. masalı			

Figure 1. The list of words that can be obtained by suffixing **only one** morpheme to the noun *masa* (*table* in English).

Root: uyu- (*sleep* in English)

Some Word Formations	**English Translations**
uyuyorum	I am sleeping
uyuyorsun	you are sleeping
uyuyor	he/she/it is sleeping
uyuyoruz	we are sleeping
uyuyorsunuz	you are sleeping
uyuyorlar	they are sleeping
uyuduk	we slept
uyudukça	as long as (somebody) sleeps
uyumalıyız	we must sleep
uyumadan	without sleeping
uyuman	your sleeping
uyurken	while (somebody) is sleeping
uyuyunca	when (somebody) sleeps
uyutmak	to cause somebody to sleep
uyutturmak	to cause (somebody) to cause (another person) to sleep
uyutturturmak	to cause (somebody) to cause (some other person) to cause (another person) to sleep
...	

Figure 2. Examples of possible word formations with derivational and inflectional suffixes from a Turkish verb root.

In this paper, we address the problem of developing statistical models for Turkish and other similar agglutinative languages such as Hungarian and Finnish. Morphological disambiguation is the task of selecting the sequence of morphological parses corresponding to a sequence of words, from the set of possible parses for those words. Morphological disambiguation is a useful prior step for syntactic parsing, since it decreases the ambiguity of the sentence, and hence makes the computational problem smaller (Voutilainen, 1998). Text-to-speech synthesis systems and spelling correction modules can also benefit from a morphological disambiguator for context sensitive selection of correct pronunciation and prosody; and for selection of correct spellings, respectively.

Turkish presents an interesting problem for statistical models since the *potential* tag set size (i.e., the number of possible morphological parses) is very large because of the productive derivational morphology. Our approach handles this by breaking up the morphosyntactic tags into inflectional groups, each of which contains the inflectional features for each (intermediate) derived form. We developed and tested four statistical models which score the probability of each morhosyntactic tag by considering statistics over the individual inflectional groups and surface roots in trigram models.

In Section 2, we present relevant properties of Turkish. Then, in Section 3, we review the related work on part-of-speech (POS) tagging and morphological disambiguation. In Section 4, we describe our statistical models for morphological disambiguation. We conclude in Section 6, after presenting and discussing our results.

2. Turkish

In this section, we discuss the properties of Turkish that complicate the straightforward application of traditional statistical language processing approaches.

2.1. Syntactic properties of turkish

2.1.1. Morphology

Turkish has agglutinative morphology with productive inflectional and derivational suffixations (Oflazer, 1994). The number of word forms one can derive from a Turkish root form may be in the millions (Hankamer, 1989). The number of possible word forms that can be obtained from a NOUN, a VERB, and an ADJECTIVE root form by suffixing 1, 2, and 3 morphemes is listed in Table I. For example, it is possible to obtain 33 different surface words by suffixing only one morpheme to a noun. Figure 1 lists the 33 possible word forms that can be obtained from the noun *masa* (*table* in English) by suffixing only one morpheme.

The number of words in Turkish is theoretically infinite, since there is no syntactic limit on the number of derivational suffixes a word can take. For example, it is possible to embed multiple causatives in a single word (as in: somebody causes

Table I. The number of possible word formations obtained by suffixing 1, 2 and 3 morphemes to a NOUN, a VERB and an ADJECTIVE

Category	Number of overt morphemes		
	1	2	3
NOUN	33	490	4,825
VERB	46	895	11,313
ADJECTIVE	32	478	4,789

some other person to cause another person ... to do something). Figure 2 gives examples of some possible word formations from the root *uyu* (*sleep* in English). Multiple causatives are the final examples in this table.

2.1.2. Word Order

Turkish is a free constituent order language, in which constituents at certain phrase levels can change order rather freely according to the discourse context or text flow. The typical order of the constituents is subject-object-verb (SOV), however, other orders are also common, especially in discourse.

The morphology of Turkish enables morphological markings on most of the constituents to signal their grammatical roles without relying on their order. This does not mean that the word order is not important, sentences with different word orders reflect different pragmatic conditions, that is the topic, focus, and background information conveyed by those sentences differ (Erguvanlı, 1979).

Word order inside the noun phrases is more constrained, with specifiers preceding modifiers, but within each group, order (e.g., between cardinal and attributive modifiers) is mainly determined by which aspect is to be emphasized. For instance the Turkish equivalents of *two young men* and *young two men* are both possible: the former being the neutral case or the case where youth is emphasized, while the latter is the case where the cardinality is emphasized. A discussion of the function of word order in Turkish grammar and statistics on the variations of word order can be found in Erguvanlı (1979). Variations in the word order complicate statistical language processing, amplifying the data sparseness problem, since we need to collect statistics on all possible orders. Therefore, more training data is required in order to reliably capture the possible word order variations.

2.2. Issues for statistical processing of Turkish

Owing to the productive inflectional and derivational morphology of Turkish, the number of distinct word forms, i.e., the vocabulary size, is very large. For instance, Table II shows the size of the vocabulary (defined as the number of distinct tokens

Table II. Vocabulary sizes for two Turkish and English corpora

Corpus size	Turkish	English
1M words	106,547	33,398
10M words	417,775	97,734

Table III. The perplexity of Turkish and English corpora using word-based trigram language models

Language	Training data	Training set perplexity	Test set (1M words) perplexity
Turkish	1M words	66.13	1449.81
	10M words	94.08	1084.13
English	1M words	43.29	161.16
	10M words	44.38	108.52

encountered, including punctuation, digits, etc.) for 1 and 10 million word corpora of Turkish and English, collected from on-line newspapers. This large vocabulary is the reason 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 perplexity of word-based models is not an issue for morphological disambiguation, it gives a flavor of the difficulty of statistical modeling for Turkish.) Table III lists the training and test set perplexities of trigram language models trained on 1 and 10 million word corpora for Turkish and English. In order to determine the trigram probabilities, we use the SRILM – the SRI language modeling toolkit (Stolcke, 1999),[1] which uses the Maximum Likelihood Estimation technique, and smoothes the probabilities using Good-Turing method (Gale, 1994) combined with the back-off modeling (Katz, 1987). For each corpus, the first data column is the perplexity for the data the language model is trained on, and the second data column is the perplexity for previously unseen test data of 1 million words. As expected, the test set perplexity reduces as we increase the training data size, but it is still very large when compared with the perplexity of, for example, English, computed using the same amount of training and test data.

Another major reason for the high perplexity of Turkish is the high percentage of out-of-vocabulary words (words in the test data which do not occur in the training data); this also results from the productivity of the word formation process. This was also observed by Çarki *et al*. (2000).

The issue of large vocabulary brought in by productive inflectional and derivational morphology 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).[2] 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 with preceding constituents. 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 *masamdakiler* below, without losing any information: the category of the root (Noun), the final category of the word as a whole (Noun) or the intermediate category (Adj).[3]

masa-m-da+ki+ler

masa+**Noun**+A3sg+P1sg+Locˆ DB+**Adj**ˆ DB+**Noun**+Zero+A3pl+Pnon+Nom

those (things) on my table

Ignoring the fact that the root word is a noun may sever any relationships with an adjectival modifier modifying the root. Consider the following phrase

mavi masa-da+ki kalem

blue on-the-table pencil

bhe pencil on the blue table

The part-of speech of the word in the middle is Adj (which is our generic tag for anything (lexical or derived) that modifies a noun). The preceding word *mavi* modifies the *table* part of the word (which is a noun). But the word in the middle actually modifies the following word, a noun. If we do not include the right representation for the middle word, we can not capture the relationship with either the previous word or with the next word.

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.

Because of the morphological properties of Turkish, it is possible for a surface word to have two ambiguous parses with the same morphological features, but different roots. For example, two parses of the word *takası* are the same except their roots:

1. taka+Noun+A3sg+P3sg+Nom
2. takas+Noun+A3sg+P3sg+Nom

Therefore, we need to include the root in the morphosyntactic tag.

2.3. Examples of morphological ambiguity

In this section, we give some examples of morphological ambiguity in Turkish, using the word *izin*, to emphasize its difference compared to, say, tagging English:

1. Yerdeki **izin** temizlenmesi gerek.
   The trace on the floor should be cleaned.
2. Üzerinde parmak **izin** kalmış.
   Your finger **print** is left on (it).
3. İçeri girmek için **izin** alman gerekiyor.
   You need a **permission** to enter.

and the following are the corresponding morphological parses, respectively:

`izin`

1. `iz+Noun+A3sg+Pnon+Gen` (trace/print)
2. `iz+Noun+A3sg+P2sg+Nom` (trace/print)
3. `izin+Noun+A3sg+Pnon+Nom` (permission)

For further examples of morphological ambiguity, and a classification of frequent types of ambiguities, see Tür (1996).

2.4. INFLECTIONAL GROUPS

In order to alleviate the data sparseness problem we break down the full tags into smaller units. Breaking up a full tag list into smaller units has been described earlier by Hajič and Hladká (1998) in the context of tagging Czech, an inflectional language. The approach for Czech is mostly representational: There are slots for all possible features to be encoded across all parts-of-speech, hence all words have the same fixed size slot structure for their tags, and nouns use some of it and verbs use some of it, etc. Our break-up is motivated not necessarily by a representational issue but by the observation that we should deal with all this very productive derivational phenomena which gives rise to essentially a variable length tag structure.

We represent each word as a sequence of *inflectional groups*, (Gs hereafter), separated by `^DB`s denoting derivation boundaries, as described by Oflazer (1999). Thus, a morphological parse is represented in the following general form:

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

where $\texttt{G}_i$ 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 word *masamdakiler* with the morphological parse given in Section 2.2 would be represented with the noun reading of the root *masa* and the following 3 inflectional groups:

1. `Noun+A3sg+P1sg+Loc`
2. `Adj`
3. `Noun+Zero+A3pl+Pnon+Nom`

In order to simplify our models further, we use the following observation of dependency relationships in Turkish: When a word is considered to be a sequence of inflectional groups, syntactic relation links only emanate from the *last inflectional group* of a (dependent) word, and land on *one of the inflectional groups* of the (head) word on the right,[4] as shown in Figure 3 (Oflazer, 1999). Figure 4 shows

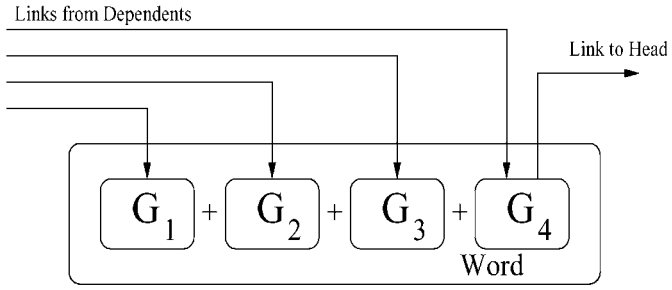

Figure 3. Inflectional groups in a word and the syntactic relation links.

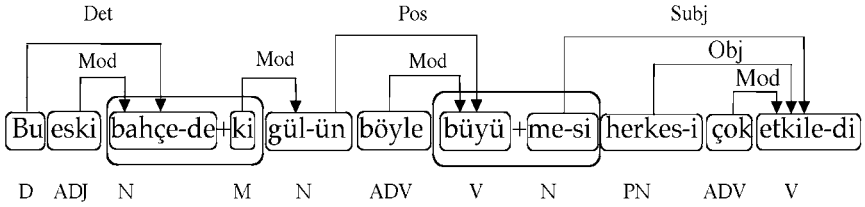

Figure 4. An example dependency tree for a Turkish sentence (*Such growth of the rose in this old garden impressed everybody a lot.* in English). The words are segmented along the inflectional group boundaries.

an example sentence with the dependency relations marked, taken from Oflazer (1999). In this example, the words are segmented along the inflectional group boundaries, marked with a '+' sign. The inflectional suffixes are marked with a preceding '–' sign.

2.5. STATISTICS ON INFLECTIONAL GROUPS

The number of possible units to be modeled is important for statistical processing. Table IV provides a comparison of the number distinct full morphosyntactic tags (ignoring the root words in this case) and inflectional groups, generatively possible and observed in a corpus of 1M words (considering all ambiguities).

As we already mentioned, the number of possible tags that are theoretically possible is infinite for Turkish. In a 1 million word corpus collected from Turkish daily newspapers, we observed 10,531 morphosyntactic tags ignoring the root words, which is very high compared to the size of the tagsets for other languages.[5] The number of possible inflectional groups as recognized by our morphological analyzer is 9,129. On the other hand, we have observed only 2,194 inflectional groups in our corpus. The number of observed inflectional groups is still very high, but smaller than the number of full tags (as expected).

Table IV. Numbers of tags and inflectional groups in a 1 million word corpus

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

On the average, in running text of about 850K tokens, there are 1.76 morphological parses/token and 1.38 inflectional groups/parse. 55% of the tokens have only one parse. Of all the parses:

- 72% have 1 inflectional group,
- 18% have 2 inflectional groups,
- 7% have 3 inflectional groups,
- 2% have 4 inflectional groups, and
- 1% have 5 or more inflectional groups.

There are also parses which have 5 or 6 inflectional groups. However, these are very rare as can be seen from the statistics.

3. Related Work

Our approach for statistical morphological disambiguation of Turkish is inspired by statistical POS tagging techniques. Therefore, we first mention the related work for POS tagging and then morphological disambiguation.

3.1. POS TAGGING

There has been a large number of studies in POS tagging using various techniques. POS tagging systems have used either a rule-based or a statistical approach.

In rule-based approaches, first, a dictionary is used to assign each word a list of potential part-of-speech tags. Then, 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 (Karllson *et al*., 1995).

In statistical approaches, a large, labeled corpus is used to train a probabilistic model which is then used to tag new text, assigning the most likely tag for a given word in a given context (e.g., Church, 1988; Garside, 1988; DeRose, 1988). It is also possible to train a statistical tagger using unlabeled data, with the expectation maximization algorithm (e.g., Cutting *et al*., 1992). Other notable approaches in POS tagging are Brill's transformation-based learning paradigm (Brill, 1995a), the memory-based tagging paradigm (Daelemans *et al*., 1996), and the maximum entropy-based approach (Ratnaparkhi, 1996). For a comprehensive overview of the related work on POS tagging, the reader is referred to van Halteren (1999).

3.2. MORPHOLOGICAL DISAMBIGUATION

Morphological disambiguation in inflectional 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. We assume that *all relevant lexical and morphological 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 of Hebrew. Hajič and Hladká (1998) have used maximum entropy modeling approach for morphological disambiguation of Czech, an inflectional language. Hajič (2000) extended this work to 5 other languages including English and Hungarian (an agglutinative language). Ezeiza *et al*. (1998) have combined stochastic and rule-based disambiguation methods for Basque, which also is an agglutinative language. Megyesi (1999) has adapted Brill's POS tagger with extended lexical templates to Hungarian.

Previous approaches to morphological disambiguation of Turkish text had employed constraint-based approaches (Oflazer and Kuruöz, 1994; Oflazer and Tür, 1996, 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.

4. Statistical Morphological Disambiguation

Morphological disambiguation is the problem of selecting the sequence of morphological parses (including the root), $T = t_1^n = t_1, t_2, \ldots, t_n$, corresponding to a sequence of words $W = w_1^n = w_1, w_2, \ldots, w_n$, from the set of possible parses for these words.

For example, the words of the Turkish noun phrase *evin terası* (the terrace of the house) have the parses given below:

evin	*terası*
1. evin+Noun+A3sg+Pnon+Nom	1. **teras+Noun+A3sg+P3sg+Nom**
2. ev+Noun+A3sg+P2sg+Nom	2. teras+Noun+A3sg+Pnon+Acc
3. **ev+Noun+A3sg+Pnon+Gen**	

The correct parse for each word is given in boldface. Among the possible parse combinations, only the first parse of the first word with the first parse of the second word, and the third parse of the first word with the first parse of the second word

make up a grammatical noun phrase. Among these combinations, only the root of the third parse of the first word occurs frequently with the root of the first parse of the second word.

4.1. THE BASIC FORM OF THE MODELS

Our approach is to model the distribution of morphological parses given the words, using a hidden Markov model (HMM), and then to seek the variable T, that maximizes the posterior probability, $P(T|W)$:

$$\underset{T}{\operatorname{argmax}}\, P(T|W) = \underset{T}{\operatorname{argmax}}\, \frac{P(T) \times P(W|T)}{P(W)} \tag{1}$$

$$= \underset{T}{\operatorname{argmax}}\, P(T) \times P(W|T) \tag{2}$$

The term $P(W)$ is a constant for all choices of T, and can thus be ignored when choosing the most probable T. Thus, Equation 1 can be simplified into Equation 2.

Following the approaches for POS tagging, we can simplify the problem of morphological disambiguation using following assumptions (Manning and Schütze, 1999):

- Words are independent of each other, given their tags, that is,

$$P(W|T) = \prod_{i=1}^{n} P(w_i|t_1^n), \tag{3}$$

  and,
- A word's identity depends only on its tag, and not on previous words or tags, that is,

$$P(w_i|t_1^n) = P(w_i|t_i). \tag{4}$$

We can then compute $P(W|T)$ as follows:

$$P(W|T) = \prod_{i=1}^{n} P(w_i|t_1^n) = \prod_{i=1}^{n} P(w_i|t_i). \tag{5}$$

We can compute $P(T)$ using the chain rule:

$$P(T) = \prod_{i=1}^{n} P(t_i|t_1^{i-1}) \tag{6}$$

and simplify Equation 6 further with the trigram tag model, so:

$$P(T) = \prod_{i=1}^{n} P(t_i|t_{i-2}, t_{i-1}). \tag{7}$$

Therefore, equation can be formulated as follows:

$$\underset{T}{\operatorname{argmax}}\, P(T|W) = \underset{T}{\operatorname{argmax}} \prod_{i=1}^{n} P(t_i|t_{i-2}, t_{i-1}) \times P(w_i|t_i) \tag{8}$$

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

This is the basic formulation of part-of-speech tagging for languages like English (Charniak *et al.*, 1993; Merialdo, 1994; Dermatas and Kokkinakis, 1995; Brants, 2000). It is also the basis of our baseline model where we use the full morphological analysis excluding the root word as the tag of the word in the conventional sense. We use the terms morphosyntactic tag, morphological analysis or parse interchangeably, to refer to individual distinct morphological parses of a token.

4.2. Simplifying the problem

In Turkish, given a morphological analysis including the root, there is only one surface form that can correspond to it, that is, there is no morphological generation ambiguity.[6] Therefore, we can assume that $P(w_i|t_i) = 1$ in the formulation above, since t_i includes the root form and all morphosyntactic features to uniquely determine the word form. In our case,

$$P(w_i|t_1^n) = P(w_i|t_i) = 1, \tag{9}$$

therefore, we can write:

$$P(W|T) = \prod_{i=1}^{n} P(w_i|t_1^n) = 1 \tag{10}$$

and the morphological disambiguation problem is simplified to:

$$\underset{T}{\operatorname{argmax}}\, P(T|W) = \underset{T}{\operatorname{argmax}}\, P(T). \tag{11}$$

Note that T represents only the possible sequences of parses that can correspond to W.

4.3. Morphological disambiguation of Turkish with n-gram language models

We use trigram language models for morphological disambiguation. The probability of a sequence of tags, $P(T)$ can be computed as follows according to the chain rule:

$$P(T) = P(t_n|t_1^{n-1}) \times P(t_{n-1}|t_1^{n-2}) \times \ldots \times P(t_2|t_1) \times P(t_1) \tag{12}$$

Simplifying Equation 12 further with a trigram tag model, we get:

$$\begin{aligned} P(T) &= P(t_n|t_{n-2}, t_{n-1}) \times P(t_{n-1}|t_{n-3}, t_{n-2}) \times \ldots \times \\ &\qquad P(t_3|t_1, t_2) \times P(t_2|t_1) \times P(t_1) \\ &= \prod_{i=1}^{n} P(t_i|t_{i-2}, t_{i-1}). \end{aligned} \tag{13}$$

The tag sequence that we are looking for is the tag sequence that has the maximum probability according to our trigram tag model (see Equation 11).

4.4. USING INFLECTIONAL GROUPS FOR MORPHOLOGICAL DISAMBIGUATION

The morphosyntactic tag model suffers from the data sparseness problem, since the number of possible tags is very large. Therefore, we decided to model smaller units which we obtain by splitting the full morphological parses across their derivational boundaries. If we consider morphological analyses as a sequence of root (r_i) and inflectional groups ($G_{i,x}$), each parse t_i can be represented as $(r_i, G_{i,1}, \ldots, G_{i,n_i})$, where n_i is the number of inflectional groups in the i^{th} word (Hakkani-Tür *et al.*, 2000).[7] r_i includes the part-of-speech category of the root word. This representation changes the problem as follows:

$$\begin{aligned} P(t_i|t_1^{i-1}) &= P(t_i|t_{i-2}, t_{i-1}) \\ &= P((r_i, G_{i,1} \ldots G_{i,n_i})|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}), \\ &\qquad (r_{i-1}, G_{i-1,1} \ldots G_{i-1,n_{i-1}})) \end{aligned} \tag{14}$$

We can use the chain rule to factor out the individual components:

$$\begin{aligned} P(t_i|t_1^{i-1}) &= P(r_i|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}), \\ &\qquad (r_{i-1}, G_{i-1,1} \ldots G_{i-1,n_{i-1}})) \times \\ &\quad P(G_{i,1}|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}), \\ &\qquad (r_{i-1}, G_{i-1,1} \ldots G_{i-1,n_{i-1}}), r_i) \times \\ &\quad \ldots \times \\ &\quad P(G_{i,n_i}|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}), \\ &\qquad (r_{i-1}, G_{i-1,1} \ldots G_{i-1,n_{i-1}}), \\ &\qquad r_i, G_{i,1}, \ldots, G_{i,n_i-1}) \end{aligned} \tag{15}$$

This formulation still suffers from the data sparseness problem, since the parameter space is very large. 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:

$$P(r_i|(r_{i-2}, G_{i-2,1}, \ldots, G_{i-2,n_{i-2}}), (r_{i-1}, G_{i-1,1}, \ldots, G_{i-1,n_{i-1}})) = P(r_i|r_{i-2}, r_{i-1}) \quad (16)$$

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

   (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 category of the root) of the previous words will be used to select the right root.

2. An interesting observation that we can make about Turkish is that when a word is considered as a sequence of inflectional groups, syntactic relations are only between the last inflectional group of a (dependent) word and with some (including the last) inflectional group of the (head) word on the right (Oflazer, 1999). Therefore, only the final inflectional groups of the preceding words can have a dependency relationship with the inflectional groups of the current word. Note that the selection of the root has some impact on what the next inflectional group in the word is, but the data sparseness problem will be severe if we model this relationship, so we assume that inflectional groups are determined by the syntactic context and not by the root.

Based on these assumptions, we define four models, all of which are based on word level trigrams. In the following subsections, we describe each of these models.

4.4.1. Model A

In Model A, we assume that the presence of the root of a word depends only on the roots of the previous two words, and the presence of inflectional groups in a word depends only on the final inflectional groups of the last two words. That is, in order to estimate the probability of an inflectional group in a word, we only look at the final inflectional groups of the previous two words, as shown below:

$$\begin{array}{lll} t_{i-2}: & \underline{r_{i-2}} & G_{i-2,1}\ G_{i-2,2}\ \ldots\ G_{i-2,n_{i-2}-1}\ \mathbf{G_{i-2,n_{i-2}}} \\ t_{i-1}: & \underline{r_{i-1}} & G_{i-1,1}\ G_{i-1,2}\ \ldots\ G_{i-1,n_{i-1}-1}\ \mathbf{G_{i-1,n_{i-1}}} \\ t_i: & \underline{r_i} & G_{i,1}\ G_{i,2}\ \ldots\ G_{i,k-1}\ \mathbf{G_{i,k}}\ G_{i,k+1}\ \ldots\ G_{i,n_i} \end{array}$$

This model ignores any morphotactical relation between an inflectional group and any previous inflectional group in the same word. Therefore, the probability of

an inflectional group is estimated as follows:

$$P(G_{i,k}|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}), (r_{i-1}, G_{i-1,1}, \ldots, G_{i-1,n_{i-1}}), r_i, G_{i,1}, \ldots, G_{i,k-1}) = P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}) \quad (17)$$

As a result, the probability of an analysis given the previous two analyses is estimated as follows:

$$P(t_i|t_{i-2}, t_{i-1}) = P(r_i|r_{i-2}, r_{i-1}) \times \prod_{k=1}^{n_i} P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}) \quad (18)$$

The first factor captures the relationship between the roots, as shown with underlining above, the second factor (which itself is the product of probabilities) models the relationship between the inflectional groups, as shown with boldfacing above.

4.4.2. Model B

In Model B, we use the assumption that the presence of the root of a word depends only on the roots of the previous two words, and the presence of inflectional groups in a word only depends on the final inflectional groups of the previous two words and the previous inflectional group in the same word, as shown below:

$$\begin{array}{lll} t_{i-2}: & \underline{r_{i-2}} & G_{i-2,1}\ G_{i-2,2}\ \ldots\ G_{i-2,n_{i-2}-1}\ \mathbf{G_{i-2,n_{i-2}}} \\ t_{i-1}: & \underline{r_{i-1}} & G_{i-1,1}\ G_{i-1,2}\ \ldots\ G_{i-1,n_{i-1}-1}\ \mathbf{G_{i-1,n_{i-1}}} \\ t_i: & \underline{r_i} & G_{i,1}\ G_{i,2}\ \ldots\ \mathbf{G_{i,k-1}}\ \mathbf{G_{i,k}}\ G_{i,k+1}\ \ldots\ G_{i,n_i} \end{array}$$

In this model, we consider morphotactical relations and assume that an inflectional group (except the first one) in a word form has some dependency on previous inflectional groups. Given that on the average a word has about 1.4 inflectional groups, inflectional group bigrams should be sufficient. The probability of an inflectional group is then estimated as follows:

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

Therefore, we compute the probability of a morphological analysis given the previous two analyses as follows:

$$P(t_i|t_{i-2}, t_{i-1}) = P(r_i|r_{i-2}, r_{i-1}) \times \prod_{k=1}^{n_i} P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, \mathbf{G_{i,k-1}}) \quad (19)$$

4.4.3. Model C

Model C uses the same assumptions with Model B, except that the dependence on the previous inflectional group in a word is assumed to be independent of the dependence on the final inflectional groups of the previous words. This allows the formulation to separate the contributions of the morphotactics and local syntax. That is,

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

and, we can first use the Bayes' rule and then the independence of the terms to divide the history in the conditional probability into two. As a result we do not need to collect 4-gram statistics but only bigram and trigram statistics:

$$\begin{aligned} &P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1}) \\ &= \frac{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1}|G_{i,k}) \times P(G_{i,k})}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})} \\ &= \frac{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}|G_{i,k}) \times P(G_{i,k-1}|G_{i,k}) \times P(G_{i,k})}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}) \times P(G_{i,k-1})} \\ &= \frac{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}|G_{i,k}) \times P(G_{i,k})}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}})} \times \\ &\quad \frac{P(G_{i,k-1}|G_{i,k}) \times P(G_{i,k})}{P(G_{i,k-1})} \times \frac{1}{P(G_{i,k})} \\ &= \frac{P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}) \times P(G_{i,k}|G_{i,k-1})}{P(G_{i,k})} \end{aligned} \tag{20}$$

Therefore,

$$\begin{aligned} P(t_i|t_{i-2}, t_{i-1}) \;=\; & P(r_i|r_{i-2}, r_{i-1}) \times \\ & \prod_{k=1}^{n_i} P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}) \times \\ & \frac{P(G_{i,k}|G_{i,k-1})}{P(G_{i,k})} \end{aligned} \tag{21}$$

4.4.4. Naive Bayes Model

Naive Bayes Model also uses the same assumptions with Model B, except that the dependence of the current inflectional group on any of the previous inflectional groups is naively assumed to be independent (as in naive Bayes classification approach).[8] Similar to Model C, we only have to collect bigram and trigram

statistics, instead of fourgram statistics, which would make the data sparseness problem more severe. That is,

$$\begin{aligned}
&P(G_{i,k}|(r_{i-2}, G_{i-2,1} \ldots G_{i-2,n_{i-2}}),\\
&\qquad (r_{i-1}, G_{i-1,1}, \ldots, G_{i-1,n_{i-1}}), r_i, G_{i,1}, \ldots, G_{i,k-1} =\\
&\quad P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})
\end{aligned}$$

and,

$$\begin{aligned}
&P(G_{i,k}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})\\
&\quad = P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1}|G_{i,k}) \times\\
&\qquad P(G_{i,k}) \times \frac{1}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})}\\
&\quad = P(G_{i-2,n_{i-2}}|G_{i,k}) \times P(G_{i-1,n_{i-1}}|G_{i,k}) \times P(G_{i,k-1}|G_{i,k}) \times\\
&\qquad P(G_{i,k}) \times \frac{1}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})}
\end{aligned} \tag{22}$$

Therefore,

$$\begin{aligned}
P(t_i|t_{i-2}, t_{i-1}) \;=\; & P(r_i|r_{i-2}, r_{i-1})\\
& \times \prod_{k=1}^{n_i} P(G_{i-2,n_{i-2}}|G_{i,k}) \times P(G_{i-1,n_{i-1}}|G_{i,k}) \times\\
& \qquad P(G_{i,k-1}|G_{i,k}) \times P(G_{i,k}) \times\\
& \qquad \frac{1}{P(G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,k-1})}
\end{aligned} \tag{23}$$

In order to simplify the notation in the description of our models, we have defined the following:

$$\begin{aligned}
P(r_1|r_{-1}, r_0) &= P(r_1)\\
P(r_2|r_0, r_1) &= P(r_2|r_1)\\
P(G_{1,k}|G_{-1,n_{-1}}, G_{0,n_0}) &= P(G_{1,k})\\
P(G_{2,l}|G_{0,n_0}, G_{1,n_1}) &= P(G_{2,l}|G_{1,n_1})\\
P(G_{i,1}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}}, G_{i,0}) &= P(G_{i,1}|G_{i-2,n_{i-2}}, G_{i-1,n_{i-1}})\\
P(G_{1,k}|G_{-1,n_{-1}}, G_{0,n_0}, G_{1,k-1}) &= P(G_{1,k}|G_{1,k-1})\\
P(G_{2,l}|G_{0,n_0}, G_{1,n_1}, G_{2,l-1}) &= P(G_{2,l}|G_{1,n_1}, G_{2,l-1})\\
P(G_{2,1}|G_{1,n_1}, G_{2,0}) &= P(G_{2,1}|G_{1,n_1})\\
P(G_{i,1}|G_{i,0}) &= P(G_{i,1})\\
P(G_{-1,n_{-1}}|G_{1,k}) &= 1\\
P(G_{0,n_0}|G_{1,k}) &= 1\\
P(G_{i,0}|G_{i,1}) &= 1\\
P(G_{-1,n_{-1}}, G_{0,n_0}, G_{1,k}) &= P(G_{1,k})\\
P(G_{1,0}) &= 1\\
P(G_{0,n_0}, G_{1,n_1}, G_{2,l}) &= P(G_{1,n_1}, G_{2,l})
\end{aligned}$$

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

4.5. IMPLEMENTATION OF THE MODELS

The models described above require two types of probabilities for the computation of the probabilities of the morphological parses: root probabilities and inflectional group probabilities. One way to construct the models is to form the root and inflectional group models that give us an estimate for the root and inflectional group trigram probabilities, and then merge these two models by computing the probabilities of all possible morphological parse sequences. However, if we do not use a threshold to limit the number of inflectional groups that can be in a parse, the number of these sequences is infinite. So, because of the derivational morphology, it is impossible to construct the complete model, that has the probabilities for all possible trigram sequences.

However, it is possible to construct a model according to the test data at run-time, by taking the product of the root and inflectional group probabilities. Such a model will not be complete, but we only compute the probabilities for the sequences that we need as we try to find the most probable tag sequence. Figure 5 shows the sequence of steps for combining the two models.

We first count the root and inflectional group sequences in the training data. Using these counts, and the SRILM, we form two trigram models that estimate the root and inflectional group probabilities.

We construct the combined models using the test data and the root and inflectional group models, at run-time, and use the Viterbi algorithm to find the most probable tag sequence. This step only requires the multiplication of the necessary root and inflectional group probabilities in order to compute the tag probabilities, and is not a part of the training process, as the root and inflectional group models were trained before we see the test data. We set the state output probabilities of our HMMs to one, and we use the root and inflectional group probabilities to compute the state transition probabilities. Once the most probable path is found, the parses on this path are output as the parses corresponding to the words in our sentence.

5. Experiments and Results

To evaluate our models, we first trained our models and then performed morphological disambiguation on our previously unseen test data, in order to make a fair test. In general, models learned from a sample of data have a tendency to expect future events to be like the events on which the model was trained, rather than allowing sufficiently for other possibilities. So, it is essential to test on different data (Manning and Schütze, 1999).

5.1. TRAINING AND TEST DATA

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/generator, developed by Oflazer (1994). We preprocessed the training and

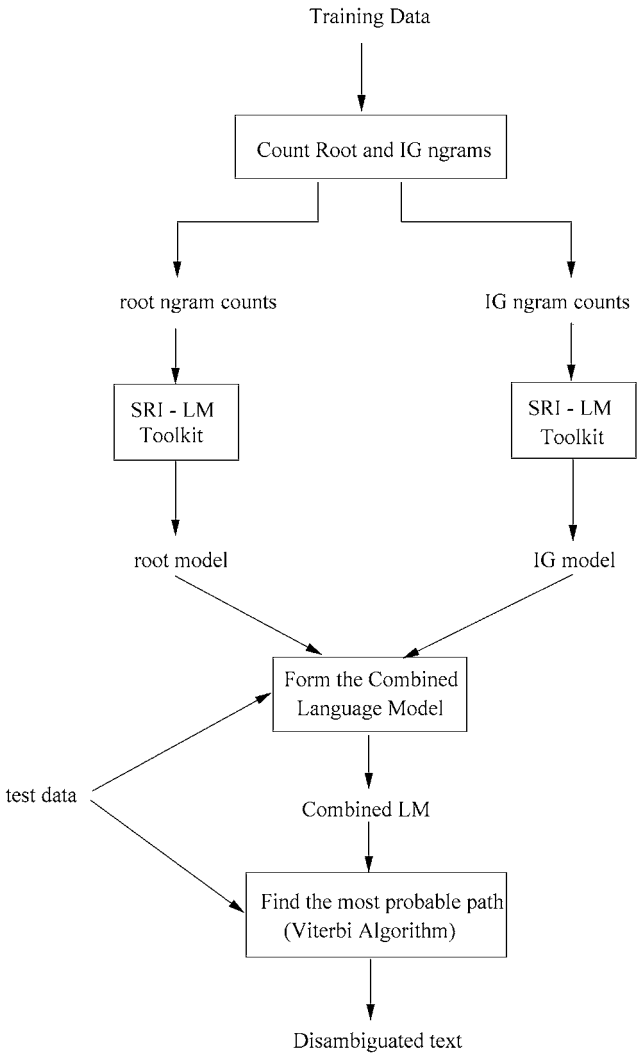

Figure 5. Implementation of the n-gram models.

test data, to reduce the morphological ambiguity. The steps of preprocessing are explained in the next section.

The training data consists of the unambiguous sequences (US) consisting of about 650K tokens in a corpus of 1 million tokens, and two manually disambiguated texts 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). Preprocessing also increases the size of the training data (that is, the unambiguous sequences).

The test data consists of 2763 tokens. The precision of the test data was 54% before preprocessing, and 65% after it. 935 (≈34%) of the tokens have more than one morphological analysis after preprocessing. Our preprocessing is very conservative, and does not reduce recall. The recall was 98.98% before preprocessing and 99.81% after it. The increase in the recall is due to the unknown word processing, which we explain in the next section. One point that should be emphasized here is that, we do the evaluation on the original manually disambiguated data.

The ambiguity of the training data was reduced from 1.75 to 1.55 and the ambiguity of the test data was reduced from 1.82 to 1.53 after preprocessing.

5.2. PREPROCESSING FOR AMBIGUITY REDUCTION

We preprocess the training and test data to reduce the morphological ambiguity, without reducing accuracy. Preprocessing also deals with the unknown words. The following are the steps of preprocessing:

1. We eliminate very rare root words that are ambiguous with a very frequent root word. An example is the word *bunlar* (*these* in English), which has the following two morphological parses:
   (a) `bun+Noun+A3pl+Pnon+Nom`
   (b) `bu+Pron+DemonsP+A3pl+Pnon+Nom`
   bun is an extremely rare root in Turkish, whereas *bu* is very frequent, so any parse with the root *bun* is eliminated.
2. We disambiguate the lexicalized and non-lexicalized collocations involving compound verbs. An example is the compound verb *yemek ye-*. For example in the sentence *Yemek yenecek* (in English *The dinner will be eaten*), the first word has the following two parses:
   (a) `yemek+Noun+A3sg+Pnon+Nom`
   (b) `ye+Verb+Pos^DB+Noun+Inf+A3sg+Pnon+Nom`
   and the second word has the following four parses:
   (a) `ye+Verb^DB+Verb+Pass+Pos+Fut+A3sg`
   (b) `ye+Verb^DB+Verb+Pass+Pos^DB+Adj+FutPart+Pnon`
   (c) `yen+Verb+Pos+Fut+A3sg`
   (d) `yen+Verb+Pos^DB+Adj+FutPart+Pnon`
   But, we know that when these words are seen consecutively, the correct parse for the first word is its first parse above, and the correct parse for the second word is the one that is derived from a Verb with root *ye*, that is, those that start with `ye+Verb`, (a, b) above.
3. We disambiguate postpositional phrases: Postpositions impose a constraint on the case of the preceding word; some subcategorize for 'Dative' noun objects, while others subcategorize for an 'Ablative', 'Nominative' etc., noun just preceding them. The subcategorization information can be inferred from the type of the postposition. For example, the word *sonra* has the following two parses:

(a) `sonra+Postp+PCAbl`
(b) `sonra+Adv`

If the preceding word is a noun in Ablative case, then the correct parse is the first one above. Likewise, the preceding noun is disambiguated if it has ablative case parses, ambiguous with other parses, eliminating the other parses.

We should emphasize that our preprocessing is very conservative. In our experimentation we have not seen any reduction in recall. But more aggressive preprocessing will certainly have an impact on recall.

The preprocessor analyzes unknown words with an unknown word processor. The unknown words are almost always foreign proper names, words adapted into the language and not in the lexicon, or very obscure technical words. These are sometimes inflected using Turkish word formation paradigms. The unknown word processor assumes that all unknown words have noun roots. It is constructed in the same way as the morphological analyzer, except it only has a noun root lexicon that recognizes any sequence of characters of length greater or equal to 1 from the Turkish surface alphabet as a potential root, provided the rest of the word (if any) can be parsed as a proper sequence of suffixes that can be attached to a noun root (Oflazer and Tür, 1996).

5.3. EVALUATION

As our evaluation metric, we used accuracy, which is the percentage of the correct parses among all selected parses:

$$accuracy = \frac{\text{number of correct parses}}{\text{number of selected parses}} \times 100$$

The number of selected parses is the number of tokens in our case, since our algorithm selects one parse among the set of possible parses for each token.

5.4. RESULTS

The accuracy results are given in Table V. For all cases, our models performed better than the baseline tag model. As expected, the baseline 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 B and C gave similar results, Model B suffered from data sparseness slightly more than Model C, as expected. Naive Bayes Model gave the worst results, supporting that our independence assumption was not correct.

Surprisingly, the bigram version of Model A (i.e., Equation (7), but with bigrams in root and inflectional group models), also performs quite well.

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

Table V. Accuracy results for different models

Model	Training Data		
	Unambiguous sequences (US)	US + 12,000 words	US + 24,000 words
Tag Model	86.75%	91.34%	91.34%
Model A	88.21%	93.52%	**93.95%**
Model A (Bigram)	89.06%	93.34%	93.56%
Model B	87.01%	92.43%	92.87%
Model C	87.19%	92.72%	92.94%
Model D	84.76%	88.49%	88.85%

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 (data column 1 in Table V). The manually disambiguated training sets have such ambiguities resolved, so the models trained using them perform much better.

In order to enhance the discussion of our results, we plot two set of learning curves before (Figure 6) and after (Figure 7) using the manually disambiguated data. When we incrementally add unambiguous sequences, the accuracies of our models increase first, and then converge. When we start using manually disambiguated data, the accuracies increase a lot at the beginning, and then again converge to upper limits. These curves also show the above point, that is, the unambiguous sequences consistently do not capture some cases, since they are almost always ambiguous, and therefore never occur in the unambiguous sequences. As a result, the models cannot learn to disambiguate these cases correctly, before they are trained with the manually disambiguated data.

We also performed Wilcoxon rank sum tests (Robbins and Ryzin, 1975), in order to see the statistical significance of difference among the models. Our best model, Model A, is significantly different than all the other models.[9] The difference between Model B and Model C is not very significant, as we expected.[10]

In order to see the contribution of the unambiguous sequences, we also trained our best model using only manually disambiguated data. We achieved an accuracy of 91.89%, which is 2.06% points lower than the accuracy with Model A, with all the training data.

If we consider just the morphological features and ignore any (lexical) semantic features (e.g., the proper noun marking) 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.[11] 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 espe-

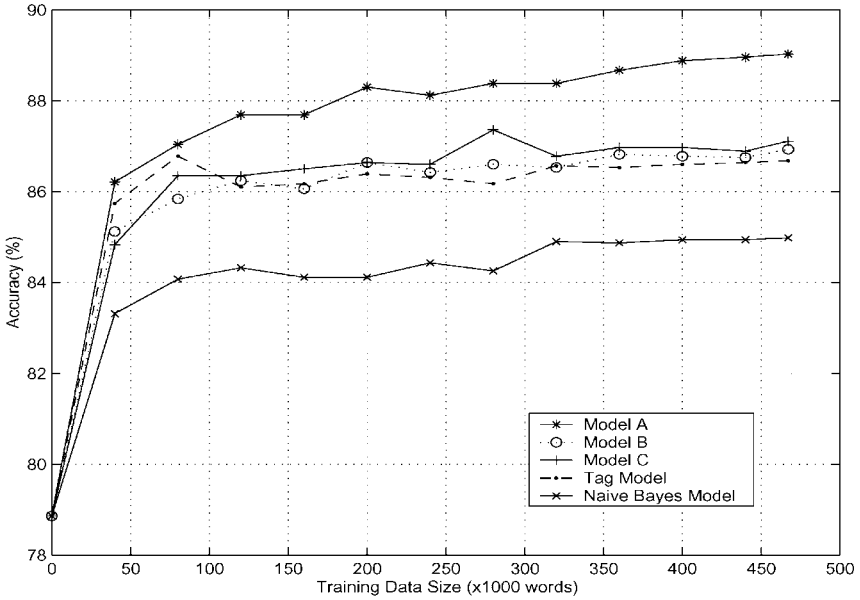

Figure 6. The learning curves of our models with only the unambiguous sequences.

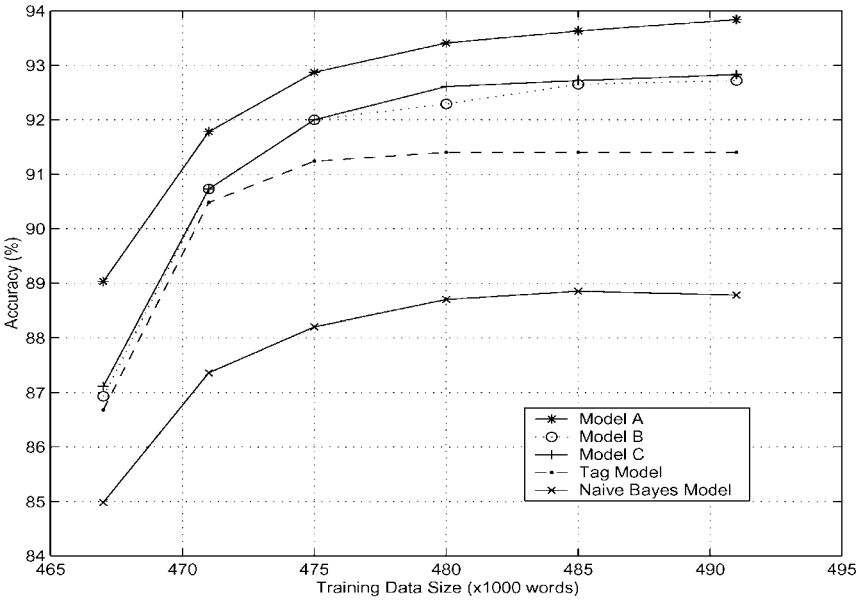

Figure 7. The learning curves of our models, using the manually disambiguated data as well as the unambiguous sequences.

cially 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.

5.4.1. Comparison with Previous Approaches

When we count as correct any errors involving semantic marker cases, we get an accuracy of 95.07% with the best case (cf. 93.95% of the Model A). 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 inflectional group 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%. Hajič (2000) reported an accuracy of 97.42% for Hungarian using full morphological tags, on about 100K training data (Orwell's novel "1984"). 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. It should be noted that, given the differences of these languages in their morphosyntactic properties, average ambiguities per token, percentage of ambiguous tokens, tag sets employed etc., these comparisons should not be interpreted as indicating the relative merits (or lack of it) of these approaches.

5.4.2. Contribution of the Root and Inflectional Group Models

Since our models assumed that we have independent models for disambiguating the root words, and the inflectional groups, we ran experiments to see the contribution of the individual models. Table VI summarizes the accuracy results of the individual models for the best case (Model A in Table V).

Using only the inflectional groups, we achieve only 0.74 percentage points improvement over the traditional approaches. Adding the root statistics to this model, we achieved 1.87 percentage points more accuracy improvement. These results emphasize the importance of the root model.

Table VI. The contribution of the individual models for the best case

Model	Accuracy
Inflectional Group Model	92.08%
Root Model	80.36%
Combined Model	93.95%

5.4.3. An Analysis of the Errors

In 15% of the errors, the last inflectional group of the word is incorrect but the root and the rest of the inflectional groups, if any, are correct. In 3% of the errors, the last inflectional group of the word is correct but the either the root or some of the previous inflectional groups are incorrect. In 82% of the errors, neither the last inflectional group nor any of the previous inflectional groups 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 inflectional group combination is not correctly determined.

5.4.4. Importance of Preprocessing

We preprocessed the training data in order to reduce the ambiguity, and increase the size of the unambiguous sequences. We also used the preprocessor for disambiguating the test data. In order to see the effect of preprocessing, we also tested the best model without preprocessing the test data. We achieved an accuracy of 93.59%, which is slightly (0.36%) less than the results with preprocessing. Preprocessing takes extra time, therefore this step can be omitted for the test data, as it does not change the final accuracy a lot.

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. In this way, we try to reduce the data sparseness problem. Among the four models that we have developed and tested for morphological disambiguation of Turkish, 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%. We also disambiguate the surface roots. Actually, we benefit a lot from using the surface roots in the morphological disambiguation problem.

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.

Acknowledgements

We would like to thank to 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. We would like to thank to the three anonymous reviewers for their comments and suggestions.

Appendix

A. Morphological Features

In this section, we provide a list of the morphological features' values used in the examples of this paper. The morphological analyzer uses a total of around 120 feature values encoding inflectional and derivational features.

`+A3sg`: 3sg number-person agreement.
`+A3pl`: 3pl number-person agreement.
`+P1sg`: 1sg possessive agreement.
`+P2sg`: 2sg possessive agreement.
`+P3sg`: 3sg possessive agreement.
`+Pnon`: No possessive marker.
`+Abl`: Ablative case marker.
`+Acc`: Accusative case marker.
`+Gen`: Genitive case marker.
`+Loc`: Locative case marker.
`+Nom`: Nominative case marker.
`+Zero`: Zero morpheme derivation.
`+Card`: Cardinal.
`+DemonsP`: Demonstrative pronoun.
`+PCAbl`: Postposition subcategorizes for an ablative object.
`+PCNom`: Postposition subcategorizes for an nominative object.
`+Pos`: Positive polarity for the verb.
`+Inf`: Infinitival.
`+Past`: Past tense.
`+Aor`: Aorist.
`+Fut`: Future Tense.
`+Prog1`: Progressive (with suffix +yor).
`+FutPart`: Future participle.
`+Caus`: Causative.
`+Pass`: Passive voice.

B. An Example Disambiguated Sentence

The following is an example sentence, disambiguated using Model 1. The surface form of each word in the sentence is given on the left hand side, and the selected parse is given on the right hand side.

```
<S>             <S>+BSTag
Milli           milli+Adj
Savunma         savun+Verb+Posˆ DB+Noun+Inf+A3sg+Pnon+Nom
Bakanlığı       bakanlık+Noun+A3sg+P3sg+Nom
dövizli         döviz+Noun+A3sg+Pnon+Nomˆ DB+Adj+With
askerlikten     asker+Noun+A3sg+Pnon+Nomˆ DB
                +Noun+Ness+A3sg+Pnon+Abl
yararlanmak     yarar+Noun+A3sg+Pnon+Nomˆ DB
                +Verb+Acquire+Posˆ DB+Noun+Inf+A3sg+Pnon+Nom
için            için+Postp+PCNom
yurt            yurt+Noun+A3sg+Pnon+Nom
dışında         dış+Noun+A3sg+P3sg+Loc
bulunma         bulun+Verb+Posˆ DB+Noun+Inf+A3sg+Pnon+Nom
süresini        süre+Noun+A3sg+P3sg+Acc
3               3+Num+Card
yıldan          yıl+Noun+A3sg+Pnon+Abl
1               1+Num+Card
yıla            yıl+Noun+A3sg+Pnon+Dat
indirirken      in+Verbˆ DB+Verb+Caus+Pos+Aorˆ DB
                +Adv+While
,               ,+Punc
dövizli         döviz+Noun+A3sg+Pnon+Nomˆ DB+Adj+With
askerlik        asker+Noun+A3sg+Pnon+Nomˆ DB
                +Noun+Ness+A3sg+Pnon+Nom
rakamını        rakam+Noun+A3sg+P3sg+Acc
iki             iki+Num+Card
katına          kat+Noun+A3sg+P3sg+Dat
çıkartmayı      çık+Verbˆ DB+Verb+Causˆ DB+Verb+Caus+Posˆ DB
                +Noun+Inf+A3sg+Pnon+Acc
planlıyor       planla+Verb+Pos+Prog1+A3sg
</S>            </S>+ESTag
```

Notes

[1] SRILM is a toolkit for building and applying statistical language models (LMs), primarily for use in speech recognition, statistical tagging and segmentation.

[2] Some researchers have used large tag sets to refine granularity, but they are still small compared to Turkish. More information on the issues of tagset design is given by Elworthy (1995).

[3] In the first line, preceding +'s mark derivational suffixes and preceding -'s mark inflectional suffixes. A list of morphological features, other than the POS categories, used in this paper are given in the Appendix A. ^DB's mark derivational boundaries.
[4] The perplexity of a unigram tag model trained using the full tags of our whole data of around 490K tokens, and tested on 2763 tokens (as explained in Section 5.1) is 436.1.
[5] Note that in very small set of infrequent cases, the head may be to the left. We ignore these cases.
[6] 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 very rarely seen in written texts, and can thus be ignored.
[7] In our training and test data, the number of inflectional groups in a word form is on the average 1.4, therefore, n_i is usually 1 or 2. We have seen, occasionally, word forms with 5 or 6 inflectional groups.
[8] This approach was suggested by Jason Eisner of University of Rochester. A similar approach was also used for Czech (Hajič and Hladká, 1998).
[9] With a p value of 4.04×10^{-4} from the baseline tag model, 3.86×10^{-7} from Model B, 3.02×10^{-9} from Model C, and 1.58×10^{-5} from the Naive Bayes Model.
[10] With a p value of 0.30.
[11] In fact, any word form is a potential first name or a last name.

References

Brants, T. "TnT – A Statistical Part-of-speech Tagger". In *Proceedings of the Sixth Applied Natural Language Processing Conference (ANLP-2000)*. Seattle, WA, 2000.

Brill, E. "Transformation-based Error-driven learning and Natural Language Processing: A Case Study in Part-of-speech Tagging". *Computational Linguistics*, 21(4) (1995a), pp. 543–566.

Brill, E. "Unsupervised Learning of Disambiguation Rules for Part of Speech Tagging". *Proceedings of the Third Workshop on Very Large Corpora*, Cambridge, MA, 1995b.

Çarkı, K., P. Geutner and T. Schultz. "Turkish LVCSR: Towards Better Speech recognition for Agglutinative Languages". *ICASSP 2000: IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, 2000.

Charniak, E., C. Hendrickson, N. Jacobson and M. Perkowitz. "Equations for Part-of-speech Tagging". *Proceedings of the Eleventh National Conference on Artificial Intelligence*, AAAI Press/MIT Press, Menlo Park, CA, 1993, pp. 784–789.

Church, K.W. "A Stochastic Parts Program and a Noun Phrase Parser for Unrestricted Text". *Proceedings of the Second Conference on Applied Natural Language Processing*, Austin, Texas, 1988.

Cutting, D., J. Kupiec, J. Pedersen and P. Sibun. "A Practical Part-of-speech Tagger". *Proceedings of the Third Conference of Applied Natural Language Processing*, Trento, Italy, 1992.

Daelemans, W., J. Zavrel, P. Nerck and S. Gillis. "Mbt: A Memory-based Part of Speech Tagger-generator". In *Proceedings of the Fourth Workshop on Very Large Corpora*. Eds. E. Ejerhead and I. Dagan, 1996, pp. 14–27.

Dermatas, E. and G. Kokkinakis. "Automatic Stochastic Tagging of Natural Language Texts". *Computational Linguistics*, 21(2) (1995), pp. 137–163.

DeRose, S.J. "Grammatical Category Disambiguation by Statistical Optimization". *Computational Linguistics*, 14 (1988), pp. 31–39.

Elworthy, D. "Tagset Design and Inflected Languages". *From Texts to Tags: Issues in Multilingual Language Analysis, Proceedings of the ACL SIGDAT Workshop*, University College, Belfield, Dublin, Ireland, 1995, pp. 1–9.

Erguvanh, E.E. *The Function of Word Order in Turkish*. Ph.D. Dissertation, University of California, Los Angeles, 1979.

Ezeiza, N., I. Alegria, J.M. Arriola, R. Urizar and I. Aduriz. "Combining Stochastic and Rule-based Methods for Disambiguation in Agglutinative Languages". *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, Montreal, Quebec, Canada, 1998, pp. 379–384.

Gale, W.A. "Good-turing Smoothing without Tears". Technical Report, Bell Labs. The corresponding postscript file can bi found at hhtp://cm.bell-labs.com/cm/ms/departments/sia/doc/94.5.ps, 1994.

Garside, R. *The Computational Analysis of English: A Corpus-based Approach*. Eds. R. Garside, G. Sampson and G. Leech, Longman, London, chapter The CLAWS word-tagging system, 1998, pp. 30–41.

Hajič, J. and B. Hladká. "Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset". *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (COLING/ACL'98)*, Montreal, Canada, 1998, pp. 483–490.

Hajič, J. "Morphological Tagging: Data vs. Dictionaries". *Proceedings of the Applied Natural Language Processing and the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL)*, Seattle, 2000.

Hakkani-Tür, D.Z., K. Oflazer and G. Tür. "Statistical Morphological Disambiguation for Agglutinative Languages". *Proceedings of the 18th International Conference on Computational Linguistics (COLING-2000)*, 2000.

Hankamer, J. *Lexical Representation and Process*. Ed. W. Marslen-Wilson, The MIT, Press, chapter Morphological Parsing and the Lexicon, 1989.

Karlsson, F., A. Voutilainen, J. Heikkilä and A. Anttila. *Constraint Grammar-A Language-independent System for Parsing Unrestricted Text*, Mouton de Gruyter, 1995.

Katz. "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer". *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume assp35:3, 1987, pp. 400–401.

Levinger, M., U. Ornan and A. Itai. "Learning Morpho-lexical Probabilities from an Untagged Corpus with an Application to Hebrew". *Computational Linguistics* 21(3) (1995), pp. 383–404.

Manning, C.D. and H. Schutze. *Foundations of Statistical Natural Processing*, The MIT Press, 1999.

Megyesi, B. "Improving Brill's POS Tagger for an Agglutinative Language". In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*. Eds. F. Pascale and Z. Joe, College Park, Maryland, USA, 1999, pp. 275–284.

Merialdo, B. "Tagging English Text with a Proabilistic Model". *Computational Linguistics*, 20(2) (1994), pp. 155–172.

Oflazer, K. and I. Kuruöz. "Tagging and Morphological Disambiguation of Turkish Text". *Proceedings of the 4th Applied Natural Language Processing Conference*, ACL, 1994, pp. 144–149.

Oflazer, K. and G. Tür. "Combining Hand-Crafted Rules and Unsupervised Learning in Constraint-based Morphological Disambiguation". In *Proceedings of the ACL-SIGDAT Conference on Empirical Methods in Natural Language Processing*. Eds. E. Brill and K. Church, 1996.

Oflazer, K. and G. Tür. "Morphological Disambiguation by Voting Constraints". *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL'97/EACL'97)*, Madrid, Spain, 1997.

Oflazer, K., D.Z. Hakkani-Tür and G. Tür. "Design for a Turkish Treebank". *Proceedings of Workshop on Linguistically Interpreted Corpora, at EACL'99*, Bergen, Norway, 1999.

Oflazer, K. "Two-level Description of Turkish Morphology". *Literary and Linguistic Computing*, 9(2) (1994), pp. 137–148.

Oflazer, K. "Dependency Parsing with a Extended Finite State Approach". *Proceedings of the 37th Annual Meeting of the Association of Computational Linguistics*, College Park, Maryland, 1999.

Ratnaparkhi, A. "A Maximum Entropy Model for Part-of speech Tagging". *Proceedings of the Empirical Methods in Natural Language Processing Conference*, University of Pennsylvania, 1996.

Robbins, H., and J.V. Ryzin. *Introduction of Statistics*, SRA, Science Research Associates, Inc., 1975.

Stolcke, Andreas. SRILM – the SRI Language Modeling Toolkit. http://www.speech.dri.com/~projects/srilm/, 1999.

Tür, G. "Using Multiple Sources of Information for Constraint-based Morphological Disambiguation". Master's thesis, Department of Computer Engineering and Information Science, Bilkent University, Ankara, Turkey, 1996.

van Kalteren, H. (ed.). *Syntactic Wordclass Tagging*. Text, Speech and Language Technology. Kluwer Academic Publishers, 1999.

Voutilainen, A. "Does Tagging Help Parsing? A Case Study on Finite State Parsing". In *Proceedings of the International Workshop on Finite State Methods in Natural Language Processing (FSMNLP'98)*. Eds. L. Karttunen and K. Oflazer, Bilkent University, Ankara, Turkey, 1998, pp. 25–36.