Minimally supervised induction of morphology through bitexts

by

Taesun Moon, B.A.

REPORT

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF ARTS

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2008

Minimally supervised induction of morphology through bitexts

APPROVED BY

SUPERVISING COMMITTEE:

Katrin Erk, Supervisor

Jason Baldridge

Acknowledgments

I wish to thank my advisor, Katrin Erk, for all her insightful comments and feedback and helping me understand that there is nothing to be ashamed of with finishing a paper in a timely fashion. Thanks are also due Jason Baldridge, who provided input that was no less insightful but gave me the greatest heartwarming gift when I learned to do the dumb thing first. My final thanks go to the inmates of the German House who taught me how to brew my own beer and to the level of potency that is desired. I still drink the same amount of rotgut, but I can now at least say that it's my rotgut.

Minimally supervised induction of morphology through bitexts

Taesun Moon, M.A.
The University of Texas at Austin, 2008

Supervisor: Katrin Erk

A knowledge of morphology can be useful for many natural language processing systems. Thus, much effort has been expended in developing accurate computational tools for morphology that lemmatize, segment and generate new forms. The most powerful and accurate of these have been manually encoded, such endeavors being without exception expensive and time-consuming. There have been consequently many attempts to reduce this cost in the development of morphological systems through the development of unsupervised or minimally supervised algorithms and learning methods for acquisition of morphology. These efforts have yet to produce a tool that approaches the performance of manually encoded systems.

Here, I present a strategy for dealing with morphological clustering and segmentation in a minimally supervised manner but one that will be more linguistically informed than previous unsupervised approaches. That is, this study will attempt to induce clusters of words from an unannotated text that are inflectional variants of each other. Then a set of inflectional suffixes by part-of-speech will be induced from these clusters. This level of detail is made possible by a method known as alignment and transfer (AT), among other names, an approach that uses aligned bitexts to transfer linguistic resources developed for one language–the source language–to another language–the target. This approach has a further advantage in that it allows a reduction in the amount of training data without a significant degradation in performance making it useful in applications targeted at data collected from endangered languages. In the current study, however, I use English as the source and German as the target for ease of evaluation and for certain typlogical properties of German. The two main tasks, that of clustering and segmentation, are approached as sequential tasks with the clustering informing the segmentation to allow for greater accuracy in morphological analysis.

While the performance of these methods does not exceed the current roster of unsupervised or minimally supervised approaches to morphology acquisition, it attempts to integrate more learning methods than previous studies. Furthermore, it attempts to learn inflectional morphology as opposed to derivational morphology, which is a crucial distinction in linguistics.

Table of Contents

Acknowledgments **iv**

Abstract **v**

List of Tables **ix**

List of Figures **x**

Chapter 1. Introduction **1**

Chapter 2. Alignment and Transfer **4**

2.1 Motivation for using alignment and transfer 6

Chapter 3. Clustering **9**

3.1 Relevant Studies 10

3.2 Pilot clustering 14

3.2.1 Lemmatization candidate trimming 15

3.2.2 Candidate set induction 18

3.3 AT-based two-constraint clustering 21

3.3.1 Word alignment 22

3.3.2 String similarity: longest common subsequence measure. 22

3.3.3 Clustering. 23

3.3.4 Thresholds for clusters. 24

3.3.5 POS tagging the target and induced clusters 25

Chapter 4. Data and Experiments **27**

4.1 Data 27

4.2 Word alignment 28

4.3 Clustering experiments 28

4.3.1 Pilot clustering experiments 28
4.3.2 Two-constraint clustering experiments 29

Chapter 5. Results and discussion 31
5.1 Evaluation of transitivity function 31
5.2 Pilot clustering 32
5.2.1 TIGER Treebank Corpus 32
5.2.2 Results 33
5.3 Two-constraint clustering 34
5.3.1 Evaluation metric 35
5.3.2 Baseline 36
5.3.3 Evaluation on CELEX 38

Chapter 6. Segmentation 44
6.1 Related work 44
6.2 Directed acyclic graph segmentation 45
6.3 Data 47
6.4 Segmentation results 48

Chapter 7. Conclusion 50

Vita 61

List of Tables

5.1 Scores by tokens and POS tag 33
5.2 Scores by types and POS tag 33
5.3 Singleton cluster baseline. 37
5.4 Prefix based baseline. 38
5.5 Performance at $\theta_{sim} = 0.88, \theta_{al} = 0.0001$ 42
5.6 Number of word types to cluster ratio by POS category. 43

6.1 Affixes induced by segmentation of clusters with DAG and tabulating those which occur most frequently. 49

List of Figures

3.1 Overview of alignment based two-constraint clustering. 9

3.2 Algorithm for computing edit distance threshold 17

4.1 Extraction of Levenshtein edit distance threshold 29

5.1 A list of German candidate verb and noun lemmas and their inflected forms extracted automatically through alignment and transitive linkage. List of candidate inflections is unordered either in terms of frequency or in terms of dictionary precedence. 32

5.2 Learning curve of recall, precision and f-score as the value of θ_{sim} is increased in increments of 0.05. $\theta_{al} = 0.0001$ with 8K corpus. 39

5.3 Learning curve of recall, precision and f-score as the value of θ_{sim} is increased in increments of 0.05. $\theta_{al} = 0.0001$ with 64K corpus. 40

5.4 Learning curve of recall, precision and f-score as the value of θ_{al} is geometrically increased in multiples of $\sqrt{10}$. $\theta_{sim} = 0.88$ with 8k corpus. 41

5.5 Learning curve of recall, precision and f-score as the value of θ_{al} is geometrically increased in multiples of $\sqrt{10}$. $\theta_{sim} = 0.88$ with 64k corpus. 42

6.1 8 nodes and 11 arcs. The paths for *ein* and *einem* have intervening nodes and are distributed across more nodes than is minimally possible 46

6.2 6 nodes and 8 arcs. *ein* and *einem* have no intervening nodes and the minimum number of nodes necessary to represent the cluster is used 46

6.3 The *e* in *ver* and *ge* are shared 47

6.4 *ver* and *ge* are split across different nodes 47

Chapter 1

Introduction

A knowledge of morphology can be useful for many natural language processing systems, e.g. for reducing the dimensionality of word alignments in machine translation (Al-Onaizan et al., 1999; Hajič, Hric, and Kuboň, 2000; Nießen and Ney, 2001; Goldwater and McClosky, 2005) or retrieval targets in information retrieval (Kraaij and Pohlmann, 1996; Krovetz, 2000; Larkey, Ballesteros, and Connell, 2002; Airio, 2006). In the case of language modeling for non-isolating languages, it is critical that the system have knowledge of the morphology to produce natural sounding string sequences (Kornai, 1996; Lee et al., 2003; Hacioglu et al., 2003).

The most accurate models of morphology that lemmatize, segment and generate new forms have been manually encoded (Koskenniemi, 1983; Karttunen, Kaplan, and Zaenen, 1992; Trost, 1990). However, manually encoding such knowledge is an expensive, time-consuming task and there have been many attempts to learn morphology in an unsupervised manner (Jacquemin, 1997; Brent, 1999; Schone and Jurafsky, 2000; Goldsmith, 2001; Creutz and Lagus, 2007) or with a reduced level of supervision (Yarowsky and Wicentowski, 2000; Yarowsky, Ngai, and Wicentowski, 2001), which often fall short of the robustness or detail of manually encoded systems.

Here, I present a strategy for dealing with morphological clustering and segmentation in a minimally supervised manner but one that will be more linguistically informed than previous unsupervised approaches. That is, this study will attempt to induce clusters of words from an unannotated text that are inflectional variants of each other. Then a set of inflectional suffixes by part-of-speech will be induced from these clusters. This level of detail is made possible by a method known as alignment and transfer (AT), among other names, an approach that uses aligned bitexts to transfer linguistic resources developed for one language–the source language–to another language–the target. This approach has a further advantage in that it allows a reduction in the amount of training data without a significant degradation in performance making it useful in applications targeted at data collected from endangered languages. In the current study, however, I use English as the source and German as the target for ease of evaluation and for certain typlogical properties of German. The two main tasks, that of clustering and segmentation, are approached as sequential tasks with the clustering informing the segmentation to allow for greater accuracy in morphological analysis.

For training the models, I use the English and German portions of Europarl (Koehn, Och, and Marcu, 2003). Evaluation data is provided by the TIGER Treebank corpus (Brants and Hansen, 2002) and CELEX (Baayen, Piepenbrock, and van H., 1993).

Since AT constitutes the core of my approach, I provide an overview of

AT in section 2 as well as the core motivation for using AT. Section 3 lays out two lexical clustering models, where one served as a preliminary experiment that informed the other, drastically revised approach that constitutes the core of the current clustering method. The experiments with these models will be presented in section 4 and the results will be presented in section 5 In section 6, I discuss my segmentation model that uses information from the clustering stage to induce linguistically informed affixes. Relevant studies will be discussed in each of the sections discussing the models. I conclude with a discussion of my approach.

Chapter 2

Alignment and Transfer

Alignment and transfer (AT) is an approach that uses aligned parallel corpora to transfer the linguistic resources developed for one language–the source–to another language–the target. Some other common names in the literature for this method are: cross-language projection, cross-language transfer, cross-language annotation transfer, etc. It has been used for a diverse range of tasks such as deriving the syntactic structure of a target language (Wu, 1997), extracting paraphrases (Pang, Knight, and Marcu, 2003; Bannard and Callison-Burch, 2005), extracting bilingual knowledge(Shin, Han, and Choi, 1996), or semantic disambiguation (Diab, 2000). Among these, one group of approaches has focused on inducing basic NLP tools such as POS taggers, noun chunkers, and morphology analyzers for a given target language (Yarowsky and Wicentowski, 2000; Yarowsky, Ngai, and Wicentowski, 2001; Yarowsky and Ngai, 2001; Drábek and Yarowsky, 2005; Ozdowska, 2006).

Yarowsky, Ngai, and Wicentowski (2001) is of particular interest to the current study since it not only analyzes the morphology of a target language, but it analyzes its *inflectional* morphology. Furthermore, it also generates new inflected forms that were unobserved in the training data and thereby enhances the coverage of its model. The shortcoming of this approach is that it is not unsupervised and some knowledge of the target language is necessary; in particular: knowledge of some candidate stems, the regular suffixes, the vowels and consonants, and a weighted matrix laying out the phonological distance between the characters in the language. The last is used to refine the minimal edit distance algorithm (Navarro, 2001) used in their study. Besides this knowledge, a core algorithm in the induction of the inflectional morphology is one which uses the relational transitivity of the word alignments between source and target to infer conditional probabilities for candidate stems given some word form. It is based on the intuition that if one lexeme and another lexeme in the target language have been aligned with more lemmas in the source language than with some other lemma, the more likely it is that the two words can be grouped together under some meaningful cluster. The function itself is a sum of the conditional alignment probabilities expanded by a Bayesian chain rule marginalized over the lemmata in the source language:

$$P(T_{lemma}|T_{infl}) = \sum_{i} P(T_{lemma}|S_{lemma_i})P(S_{lemma_i}|T_{infl}) \qquad (2.1)$$

where T_{lemma} is a candidate lemma for the target language taken from a predefined list, T_{infl} is a lexeme in the target and S_{lemma_i} is a lemma in the source.

With just (2.1) and a fixed set of stems, they post a precision of 0.992 and a recall of 0.994 for the 12M word French Hansards. However, it should be noted that the induction was performed for only verbs in the target language. Also, they had implemented a POS tagger for the target induced through similar minimally supervised means before inducing a morphological analyzer. With further refinements, they post a precision of 0.99 and retrieval of 1.00. Yarowsky and Wicentowski (2000) is recommended for further details on this minimally supervised morphology induction scheme.

2.1 Motivation for using alignment and transfer

Unsupervised approaches for morphology induction can be categorized into two groups. One approach does not use any contextual information and relies purely upon the information derived from the orthography–in other words, the distributional properties of the characters in relation to the word types–to segment the words observed in a text (Jacquemin, 1997; Brent, 1999; Goldsmith, 2001; Creutz and Lagus, 2007). The other approach takes a more sophisticated view of morphology by assuming that some form of semantic relatedness must be considered as well as surface string similarity (Schone and Jurafsky, 2000; Schone and Jurafsky, 2001; Baroni, Matiasek, and Trost, 2002; Freitag, 2005). The basis of this semantic relatedness is a statistical analysis of the distribution of the words in an unannotated text.

Unfortunately, distributional properties alone cannot generate clusters that go beyond any fuzzy notion of semantic relatedness and face the same limitations as other bag-of-word approaches (Jones and Mewhort, 2007). The morphological clusters induced by these approaches are unable to distinguish words that are related through syntactic inflection and semantic derivation. Considering that this is a core distinction in standard approaches to morphology, it is a grave oversight. AT can overcome these limitations, not because of the superiority of the underlying model, but because it can leverage any resources that might exist for the source language and induce fine-grained structural information about the target language.

The absolute size of the data involved can also be reduced through AT. Though the original motivation for AT was to reduce the costs involved in resource production, AT has rarely been applied to an even further impoverished subgroup of under-resourced languages, that of underdocumented, endangered languages. In an informal survey of colleagues who work on underdocumented languages, the amount of data collected was concentrated mostly in the thousands to the low tens of thousands with one outlier that had accumulated 700,000 words, albeit with significant contributions from the indigenous community. Contrast this with monolingual, distribution-based approaches which were based on corpora ranging from 1.2 million to 28 million words. In light of this fact, I propose that one further dimension must be added to the discussion of under-resourced languages in the AT literature in addition to the amount of annotated data and NLP tools: the amount of raw data that has been amassed for a language.

Therefore, with an explicit aim to help the documentary community, I provide the final motivation for this approach by latching on to an aspect of data collected in the field on underdocumented languages in that they are often interlinearized with a translation or gloss in a language that has a healthy amount of linguistic resources. Furthermore, in some situations, an underdocumented language will have partial or whole translations of the Bible, broadening the landscape in which this approach may be applied. In many cases, the metalanguage in which the language is documented will be a language with considerable NLP resources such as English or Spanish, and therefore I assume, like Yarowsky et al., that a rich set of computational and linguistic resources are available for the source language.

Given the difficulty of evaluating such an approach, I use German as the target language for ease of evaluation since it constitutes part of CELEX. Also, I use the 28M word Europarl corpus as our training set but limit the size of our data to simulate data collected from the field. In addition to these logistic conveniences, German possesses some interesting typological characteristics that are advantageous to us from a linguistic perspective. It displays diverse morphological patterns that encapsulate much of the regular morphology observed in human languages. It has a regular circumfix in addition to its regular suffixes; its separable prefixes exhibit regular behavior when finite; it has a considerable amount of vowel gradation (which could be considered analogous to infixation). Also, given its liberal use of noun compounding, it could even potentially be employed as a testbed for agglutinative patterns.

Chapter 3

Clustering

Previous studies have shown that improvements in performance can been gained for morphological segmentation when some form of semantics is considered (Schone and Jurafsky, 2000; Freitag, 2005). This vague notion of semantics is induced by conducting a statistical analysis over the distribution of the words in a monolingual text. I present an alternative approach based on AT that can induce more informative clusters that conform to linguistic notions of inflectional morphology and that label these clusters by part-of-speech that will used for the segmentation step.

In the following sections, I discuss in more detail relevant approaches, a preliminary method that helped refine the current core clustering approach,

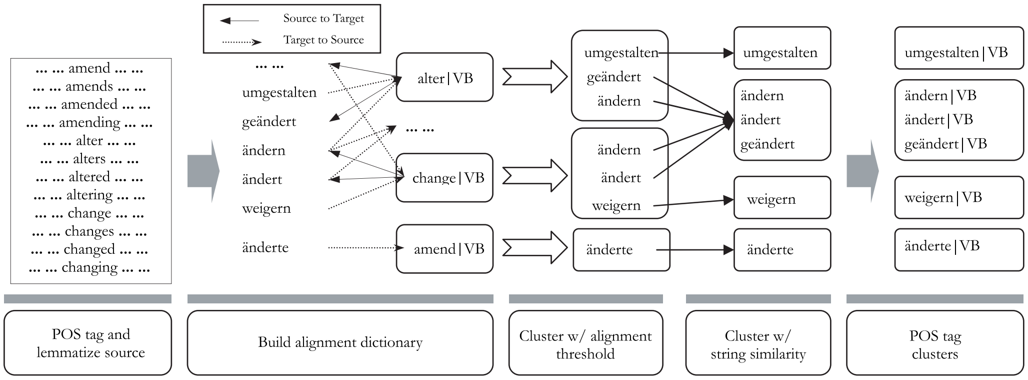

Figure 3.1: Overview of alignment based two-constraint clustering.

and my AT-based two-constraint clustering. In brief, my core method (Figure 3.1) induces clusters based on inflectional paradigms by using constraints from word alignments between bitexts and a string similarity measure between words or candidate clusters. Then, through a HMM POS-tagger I induce for the target language, I tag the clusters obtained.

3.1 Relevant Studies

In the literature, morphological clustering is a lexical clustering task that attempts to group words together based on some form of morphological affinity (Jacquemin, 1997; Schone and Jurafsky, 2000; Schone and Jurafsky, 2001; Baroni, Matiasek, and Trost, 2002; Freitag, 2005). In these studies, no distinction is made between inflectional morphology and derivational morphology, and the term *conflation set* is often used to refer to the clusters generated, as there is no corresponding term in standard studies of morphology from a non-computational perspective. In traditional approaches, the central dichotomy in morphological processes is between inflectional morphology and derivational morphology. This lack of distinction in computational approaches seems to have not had an effect on performance when applied to information retrieval (Krovetz, 2000; Larkey, Ballesteros, and Connell, 2002).[1] When morphological processing is applied to MT, however, a strict distinction is made so that only processing for inflectional variation is done (Hajič, Hric, and Kuboň, 2000; Nießen and Ney, 2001; Goldwater and McClosky, 2005) for any highly

[1] see Kraaij and Pohlmann (1996) or Airio (2006) for different results

inflecting languages that might be involved in the task.

In spite of its usefulness, lemmatization–the task of grouping together words that are mutual variants in an inflectional paradigm–is very challenging to approach as an unsupervised task. Viewed as a clustering problem, it is not plausible, as is done with a popular clustering algorithm such as the K-means, to posit a fixed number of clusters beforehand. Basing distance between cluster members solely on some measure of string similarity ignores the many irregularities that will exist in any morphologically complex language; to say nothing of the problems that exist for words which might be similar in terms of orthography but are unrelated in terms of either semantics or syntax. Therefore, to induce clusters of words belonging to the same inflectional paradigm, it is necessary to apply a clustering constraint based on both semantics and syntax in conjunction with a string similarity measure. It need not be said that there is no known way of inducing such a syntactic/semantic constraint in an unsupervised manner that is sufficiently robust to serve as the basis for other tasks.

As a result of such challenges, current methods in morphological clustering disregard distinctions between inflectional and derivational morphology, but an approach exists which improves clustering performance within this loose definition of morphological relatedness. In its broad outlines, it is a two component process that uses the distribution of words on the one hand and surface features, i.e. the string representation itself, on the other to refine the clusters. Quite possibly an exhaustive overview of such studies is listed below:

First used in Schone and Jurafsky (2000) [2], the two-constraint clustering model induces a loose set of clusters based on word distribution and then further refines the clusters through some means dependent on the surface strings. Specifically, Schone and Jurafsky (2000) uses latent semantic analysis (Deerwester et al., 1990) to build a vector space of the words in an unannotated document in English, then using a ranked set of suffixes derived from a forward trie (Fredkin, 1960) and a ranked set of prefixes from a backward trie, it further refines the the vector space to build pairs of potential morphological variants. Using a cutoff theshold of 0.7 in the vector space they induce clusters with an f-score of 84.3 when evaluated against CELEX. They use the TREC corpus comprising some 8 million words for the training.

Schone and Jurafsky (2001) extends this model to consider transitive links between words that may not be directly related in the vector space model but might be considered related through transitivity. They also extend their model to consider circumfixation. They use newswire texts for English (6.7M words), German (2.3M words) and Dutch (6.7M words) to induce clusters and evaluate against CELEX. Evaluated for suffixes alone, they obtain f-scores of 88.1, 92.3, 85.5 for English, German and Dutch respectively.

Baroni, Matiasek, and Trost (2002) uses pointwise mutual information to induce the first set of distribution based clusters and refines these clusters with the Levenshtein edit distance measure. They run their experiments using

[2] The idea itself was preceded in Xu and Croft (1998). This was, however, a semi-supervised approach.

the Brown corpus for English and the APA corpus for German which constitute 1.2M words and 28M words respectively. To evaluate, they manually build a standard of some 5000 word pairs using the XEROX morphology analyzer for each language and compare the top ranked pairs induced by their model with this standard. By this measure, precision ranged from 97% for the standard with 500 pairs to 50% for the standard containing all 5000 pairs.

Freitag (2005) employs information theoretic co-clustering (Dhillon, Mallela, and Modha, 2003) to automatically induce a set of term clusters such that the difference in mutual information between clusters is maximized. Then it induces a set of affixes on a method that is similar to the trie used in Schone and Jurafsky (2000). Next, it goes further then the two previous studies by building a finite state automaton from these clusters and affixes. This automaton is built from the Wall Street Journal corpus and is evaluated over CELEX. Note that it is the automaton that is evaluated and not any clusters that have been induced. In these experiments, the f-score ranges from 81 to 92 depending on the size of the automaton that is evaluated.

One common feature of these studies is that, in spite of the fact that all of the above models have parameters that need to be or can be tuned, no held-out development sets are used to set them. Instead, either separate results are presented according to parameter value or they are assumed (as in the size of the rank for LSA approximation in Schone and Jurafsky (2000)). My core two-constraint approach faces the same limitations, and evaluation results will be presented for various parameter values and subsets of results in section 5. The previous incarnation of two-constraint clustering was an attempt to eliminate as many manually determined parameters as possible while using the Levenshtein edit distance as a string similarity measure. I present this pilot approach in detail in section 3.2.

Another common aspect of these studies is their limitation in terms of the morphological phenomena they deal with: prefixation, suffixation and–only in the case of Schone and Jurafsky (2001)–circumfixation. This is a limitation that cannot be overcome with trie based methods. Instead, I present a novel string similarity measure that can deal with a greater variety of morphological processes in section 3.3.

3.2 Pilot clustering

The intent of this approach was to examine whether lemmatization with AT is possible without the transitivity function function (2.1). First, I limit the set of candidate lemmata to the word types in the target language which have the greatest possibility of being associated with some lemma in the source language. With this candidate lemma, I generate one set of lemmata to inflected form mappings by limiting the linkages to those source lemmata and target word type associations which exceed a manually determined probability threshold. I generate a second set of mappings from a candidate lemma to a set of target word types which has been limited to those which have been observed in alignment with a source lemma and then further reduced through an automatically induced edit distance threshold.

3.2.1 Lemmatization candidate trimming

Through some alignment model, whether it is a heuristic alignment such as the Dice coefficient or EM-based alignment (Och and Ney, 2003), the conditional probabilities between POS tagged and lemmatized words in the source and raw types in the target are calculated:

$$P(\ell_s T_s | w_t) \tag{3.1}$$

$$P(w_t | \ell_s T_s) \tag{3.2}$$

where subscripts s and t are source and target texts, respectively, ℓ and T are lemma and POS tag, respectively, and w is a word type in the target language. ℓ_s is an element of the set Λ_s which is the set of all lemmata observed in the source language and w_t is an element of the set W_t which is the set of all types observed in the target language. In contrast to Yarowsky, Ngai, and Wicentowski (2001) where only morphology for verbs were induced, I attempt here and later in section 3.3 to induce the inflectional morphology for nouns, verbs, and adjectives–in short, all the content word categories in English except for the adverbs.

This is one reason that all lemmata in the English source had to be considered with their respective POS tags, considering that many lemmata in English can be ambiguous with regard to word category when judged on their surface form alone. In this section, to simplify notation, all source lemma arguments in functions shall be assumed to also be tagged with relevant POS information. As such, the above equations are equivalent to

$$P(\ell_s|w_t) \tag{3.3}$$

$$P(w_t|\ell_s) \tag{3.4}$$

Also, note that words in the aligned target text are merely assumed to be a word type in the most general sense, since no assumptions can be made at this point whether a particular word form observed in the target language is the inflected form of some lemma or is itself the general "dictionary entry form".

In the estimation of the probablities in (3.3) and (3.4), I made an unjustified but practical decision to limit the set of target word types under examination to those which have string lengths of four or longer. This was mainly due to the fact that the Levenshtein edit distance algorithm is incapable of calculating meaningful scores when the strings being compared are both very short.

To limit the search space, I build two mapping tables, one from the target word types to the source lemmata and another from the source lemmata to the target word types.

The mapping from the target to the source, $TS : W_t \rightarrow \Lambda_s$, is built by

$$TS(w_t) = \ell_s \text{ iff } P(\ell_s|w_t) > \theta_{al} \tag{3.5}$$

The mapping from the source to the target, $ST : \Lambda_s \rightarrow W_t$ is built by

$$ST(\ell_s) = \arg\max_{w_t} P(w_t|\ell_s) \tag{3.6}$$

Using the two mappings TS and ST, I automatically determine a minimal Levenshtein edit distance threshold by comparing the edit distance between all possible W_t to W_t mappings,

$$ST(TS(w_t)) = w'_t \tag{3.7}$$

where $w_t, w'_t \in W_t$. The mapping obtained here will be necessary for limiting the search space for the first set of candidate lemma to candidate inflectional form mappings.

```
Declare: $a[0 \ldots n]$
for $j$ from 0 to $n$ do
  $a[j] := 0$
end for
for all $w_t \in W_t$ do
  if $TS(w_t) \neq$ NONE then
    $w'_t := ST(TS(w_t))$
    $d :=$ edit_distance($w_t$,$w'_t$)
    if $d < n + 1$ then
      $a[d] := a[d] + 1$
    end if
  end if
end for
return $min(a[0 \ldots n])$
```

Figure 3.2: Algorithm for computing edit distance threshold

The specific algorithm for computing the edit distance threshold is laid out in Figure 3.2. The edit distance for every w_t, w'_t pair in (3.7) is calculated and I tabulate how many times each edit distance score was observed (which is stored in an array a of length n in the algorithm; in this case, an array of length 9 is used). Finally, the edit distance threshold is determined to be the minima among the frequency counts by edit distance score. Furthermore, even if the number of edit distance scores I keep track of is increased to include all edit distance scores, it is evident that a score and its frequency count will continue to increase until reaching some asymptotic upper limit for all real-word data. Therefore, though the highest edit distance score the model maintains a frequency count of is 9, there is no possibility that the frequency count will decrease at some point above that score. The intuition behind the approach is that two target words which have an edit distance beyond a certain threshold are more likely to be noise and those which do not exceed it will be related within some inflectional paradigm; and that this threshold exists at the minima of the frequency counts.

3.2.2 Candidate set induction

The first set of lemma group candidates were induced as follows. First, generate a mapping M from a source lemma ℓ_s to a set of target word types $\Omega_t \subset W_t$ where

$$\Omega_t = \{w_t | P(w_t | \ell_s) > 0\}$$

With this mapping

$$M(\ell_s) = \Omega_t \tag{3.8}$$

further trim Ω_t by pegging the lemma candidate as $ST(\ell_s)$ (see equation (3.6)) and removing all the elements in Ω_t which have a Levenshtein edit distance score from $ST(\ell_s)$ greater than the distance threshold obtained through the algorithm in Figure 3.2, resulting in Ω'_t, a subset of Ω_t.

Thus, a set of lemma candidates Λ_t in the target language is obtained

$$\Lambda_t = \{\ell_t | \forall \ell_s \in \Lambda_s, ST(\ell_s) = \ell_t\}$$

and a set of inflections associated with each ℓ_t in Λ_t

$$C_1(\ell_t) = \Omega'_t$$

Furthermore, the candidate lemma ℓ_t inherits the POS tag from the source language, so that ℓ_t is also specified for whether it is an adjective, noun, or verb.

A second candidate set, or a mapping from candidate lemma to candidate inflected forms, is induced by trimming the mapping TS to a subset of mappings where if the length of the common substring between the input and the output is less than 4, it is removed. However, the common substring in this case is not the longest common substring assumed in general, but merely the common substring from the beginning of each string being compared.

The justification for this is as follows. A very simple assumption can be made that a language will be either prefixal or suffixal in its inflectional system. By implementing two tries over the entire set of word types in the target language W_t, one trie starting from the beginning of the strings and another starting from the end of the strings, it seems possible to compare how many terminal nodes there are for the forward trie and the reverse trie, the intuition being that the more terminal nodes a particular trie has, the less likely it is that morphological affixation occurs at the terminal nodes of that trie. In the case of this approach, it was found that the forward trie had 898 terminal nodes whereas the reverse trie had 4387 terminal nodes. Hence, I came to the simplified conclusion that the target language was suffixal rather than prefixal in generating inflected forms.

The second candidate lemma to candidate inflection mapping, unlike the first candidate mapping, is not from a word type to a set, but from a word type to a word type. I define the second candidate mapping C_2 as follows:

```
for all $w_t \in W_t$ do
    if $ST(TS(w_t)) \neq$ NONE then
        $w'_t := ST(TS(w_t))$
        if $CS(w_t, w'_t) < 3$ then
            $C_2(w_t) = w'_t$
        end if
    end if
end for
```

where CS is a function on two strings which returns an integer value of the longest common substring starting from the beginning of the two arguments and $ST(TS(w_t))$ is the mapping stated in (3.7).

Finally, the model combines the two candidate mappings into a final candidate mapping C which is a relation from a word type to a set of word types. If there are coinciding ℓ_t in C_1 and C_2, then the output of C_2 is merged into the set generated by C_1. Otherwise, candidates are simply added to the mapping C.

3.3 AT-based two-constraint clustering

In this revised approach, the aim is the same: it is to induce clusters of inflectional variants of a stem for the target language, based on a speculative lemmatization inferred from the parallel corpus. This is implemented as a two constraint clustering task where one set of measures based on alignment probability between words in a bitext pair and another set of measures based on string similarity between words in the target are used to define the clusters.

Prior to alignment, as in the previous approach, the source text is POS tagged and lemmatized. The intent is to partition the source in such a way that the word space is clearly reduced and segmented along the semantic and syntactic dimension. By lemmatizing the source, diverse word forms are reduced to a corresponding lemma with a resulting reduction of the source semantic space that the target must align with. That is, if one assumes that inflected variants of a stem all share semantics, sources of noise are reduced for the target by reducing this space. However, if you also assume that the eliminated morphemes are at the same time somehow indicative of the syntactic properties of a source word, POS tagging the source can maintain the reduction in semantic space and simultaneously create a partition with reduced noise along the syntactic domain so that alignments from and to the target have clear indicators regarding syntax. This approach allows the model to go beyond morphological clustering and perform lemmatization.

3.3.1 Word alignment

Again, an alignment dictionary is built between the source and target using a statistical alignment model. However, the use of a heuristic alignment algorithm such as the Dice coefficient of the Jaccard measure is not ruled out.

3.3.2 String similarity: longest common subsequence measure.

The longest common subsequence (LCS) algorithm searches not for the longest *contiguous* substring between two strings but the longest sequence of characters in common between two strings regardless of intervening material. Illustrating this with the string pair *william* and *willlaim* (Bilenko et al., 2003), the LCS is either *willim* or *willam* depending on which string one assumes as the pivot of comparison. I devise a measure $LCSM$ based on the LCS that outputs a score between 0 and 1 that proportionally indicates a higher degree of string relatedness:

$$LCSM(t_1, t_2) = \frac{1}{2} \cdot \left(\frac{|\mathrm{LCS}(t_1, t_2)|}{|t_1|} + \frac{|\mathrm{LCS}(t_1, t_2)|}{|t_2|} \right)$$

Basically, this averages over the LCS value normalized by the length of each string. Using this measure, the distance between *william* and *willlaim* is 0.79.

Though there are other measures of string similarity that are commonly used in information retrieval and morphology clustering such as Jaro-Winkler (Cohen, Ravikumar, and Fienberg, 2003) or the Levenshtein edit distance measure, I found that they were unsuitable to both our assumptions and this task. While I make no assumptions whatsoever about the morphological patterning of our target language, the Jaro-Winkler measure is weighted to favor matching prefixes, and therefore only suitable for suffixal languages. The Levenshtein edit distance (LED) measure is affected by long sequences of non-matching material, making it inadequate for detecting morphological variation. LED is also unnormalized[3], and therefore the LED between a string pair such as *ab* and *abc* is equal to the difference between *abcdef* and *abcdefg*. It would be preferable to penalize differences in short strings over differences in longer strings. $LCSM$ is free of these problems.

3.3.3 Clustering.

Standard bottom-up clustering is performed, starting with each target word form in its own clusters, and merging clusters iteratively (Jain, Murty,

[3] Though normalizing it to have a value between 0 and 1 is trivial. I note that Baroni, Matiasek, and Trost (2002) do not normalize it.

and Flynn, 1999). Merging proceeds as long as merging thresholds are exceeded.

3.3.4 Thresholds for clusters.

Two different thresholds for merging two clusters are used:

- an alignment threshold: θ_{al},
- a similarity threshold for $LCSM$: θ_{sim}

For a source language word form s, the set of target words aligned with s according to θ_{al} are defined as

$$AL_{\theta_{al}}(s) = \{t \in T \mid P_{st}(t|s) > \theta_{al} \vee P_{ts}(s|t) > \theta_{al}\}$$

where T is the vocabulary of the target language, P_{st} is alignment probability for the alignment from English to the target language, and P_{ts} is the alignment probability for the inverse alignment.

The distance between any two clusters (both singleton and non-singleton) is defined as:

$$d(C_1, C_2) = \frac{1}{|C_1| \cdot |C_2|} \sum_{t_1 \in C_1, t_2 \in C_2} LCSM(t_1, t_2) > \theta_{sim}$$

That is, the distance is the mean of the $LCSM$ between each and all elements in the two clusters.

Then, clusters are merged if they fit the following criterion:

$$\begin{aligned}(\exists s \in S : \exists t_1 \in AL_{\theta_{al}}(s) \wedge \exists t_2 \in AL_{\theta_{al}}(s)) \quad \wedge \\ d(C_1, C_2) > \theta_{sim}\end{aligned}$$

That is, two clusters are merged if at least one of the members in each cluster have a high alignment probability with a common source word, and if their string similarity according to the LCS measure is high.

Use of the alignment scores in conjunction with string distance measures ensures that many false positives, which would have been matches under a methodology that is based purely on string similarity, are culled from the search.

3.3.5 POS tagging the target and induced clusters

The lemma and POS information that are assigned to the English text is transferred to the target text, then used to train an HMM tagger to tag the target.

Words in the target sentences are initially given a POS tag by direct transfer from the source sentences that they are aligned with. Then ten forward-backward iterations are then run with HMM tagger from the AustinNLP suite over the tagged target text to obtain the HMM model parameters. This model is used to tag the target text that it trained, overwriting the previous tags which had existed. This HMM tagged target is used to calculate the unigram tag probabilities $P(\text{POS}|\text{target word type})$. Next, the clusters derived through the core two-constraint approach above are given a POS tag. With the probabilities derived through the HMM, I take a simple naive Bayes assumption and give the cluster C_k the POS tag τ as follows:

$$\arg\max_{\tau} \prod_{t \in C_k} P(\tau|t)$$

Chapter 4

Data and Experiments

4.1 Data

The German and English sections of the Europarl parallel corpus (Koehn, 2005) were used in this study. The Europarl parallel corpus is a collection of texts in 11 languages [1] extracted from the proceedings of the European parliament with each text comprising some 25 to 30 million words. Different portions of this corpus were used for the pilot clustering and the two-constraint clustering task.

Normalization steps are taken for both texts. All ISO 8859-1 upper ASCII alphabet characters are simplified to corresponding lower ascii characters (e.g. *è* → *e*, *ñ* → *n*) or, in the case of German specific characters, converted to conform to notations in the CELEX database[2]. Then, all uppercase characters are lowercased and all non-alphabetic characters are excluded from the clustering process.

[1] i.e. 11 of the 23 official languages in the European Union: Danish, Dutch, English, Finnish, French, German, Greek, Italian, Portuguese, Spanish and Swedish

[2] i.e. the standard conversions of *ä* → *ae*, *ö* → *oe*, *ü* → *ue*, and *ß* → *ss*

4.2 Word alignment

Word alignment was done with GIZA++ (Och and Ney, 2003). I use both an alignment with English as source and German as the target languages, and the inverse alignment with German as the source.

4.3 Clustering experiments

4.3.1 Pilot clustering experiments

Data In the pilot clustering experiment I used the entire English and German portions of Europarl, though sentences which were longer than 45 words in length were excluded to reduce the burden of the alignment task.

POS tagging of source text Before alignment, the English source text was tagged for part-of-speech. POS tagging for the English text was done with the maximum entropy based C&C tagger (Curran and Clark, 2003), which was trained on the Wall Street Journal of the Penn Treebank.

Alignment GIZA++ alignment was executed using the default parameters of five iterations each of Model-1 > HMM > Model-3 > Model-4. Given the amount of data available, accuracy is pursued over coverage.

Induction of edit distance threshold Using the algorithm outlined in Figure (3.2), I induce the minima of the edit distance for the entire data set.

The frequencies can be observed in Figure 4.1, the graph of which is a convex function.

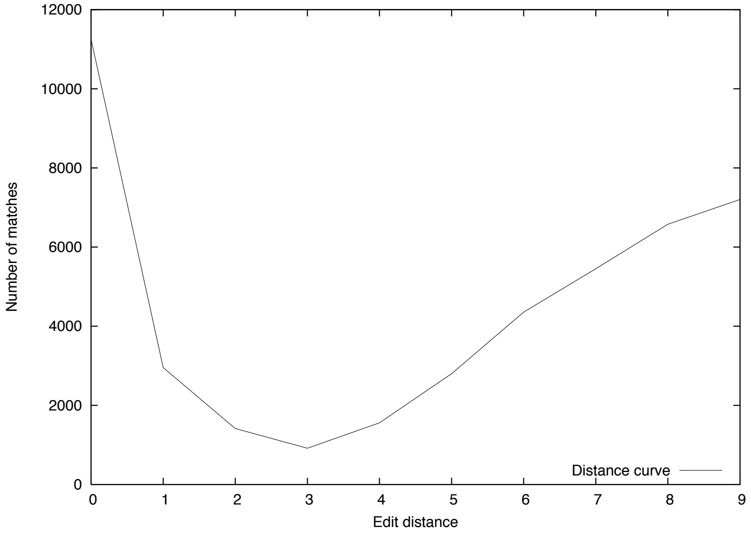

Figure 4.1: Extraction of Levenshtein edit distance threshold

In this figure, the minima of the function is found at 3. This threshold is used to trim the candidate clusters obtained in section 3.2 through (3.8).

4.3.2 Two-constraint clustering experiments

Data For the task at hand, sentences of ≤ 30 words length were selected at random from the corpus English/German portions of Europarl. To be able to check corpus size effects, I built Europarl subsets of 1K, 2K, 4K, ..., 512K words, in geometric increments, each subset incorporating the previous one. Rather than use the full amount of text in Europarl, I handicap the size of the data sets, as mentioned in section 2, to test the feasibility of this approach on underdocumented languages.

Lemmatization and POS tagging of source text Before alignment, the English source text was tagged for part-of-speech and lemmatized. POS tagging for the English text was done with the maximum entropy based C&C tagger (Curran and Clark, 2003), which was trained on the Wall Street Journal of the Penn Treebank. The POS tagged source text was then supplied to the lemmatizer, Morpha (Minnen, Carroll, and Pearce, 2001). This preprocessing step reduces the number of source types that can be aligned to some type in the target. This process of POS tagging and lemmatization restricts the alignments to a specific subset of clusters so that a partition is established between POS boundaries within a reduced lexical space.

Alignment For this task, GIZA++ alignment was executed using only a combination of Model-1 and HMM instead of a standard four model combination. For the purposes of the current experiment, Model-3 and Model-4 take into account fertilities and distortion probabilities and thus perform too much smoothing by eliminating alignments generated in the previous models, alignments which might prove useful in our situation where we are not depending on these probabilities to estimate new probabilities but merely depending on them to build a loose, initial cluster. Since the corpora are small, I err in favor of coverage rather than accuracy.

Chapter 5

Results and discussion

In this section, I discuss the results from an experiment with the transitivity function (2.1) as used in Yarowsky, Ngai, and Wicentowski (2001). Next, I present the results from my pilot clustering experiment and finally the two-constraint clustering.

5.1 Evaluation of transitivity function

In my implementation of the transitivity function in (2.1), I modified the model so that it would not make any assumptions about which words in the target are lemmata and which are not. A small subsample of the results can be observed in Figure 5.1. In addition to the examples observed in the subsample, the amount of noise in the results in general were excessive and ultimately unfit for inducing lemmatization schemes. A qualitative inspection of the results for this method, however, do indicate that a form of clustering according to a vague notion of semantics occurs.

	Lemma	Inflections
Verbs	*ergänzen*	unternommenen betriebsrat ergänzung abrunden abkehr zusammenführen vervollständigen weswegen flüchtlingskonvention entwicklungschancen staatsangehörigkeit ergänzend ergänzen einander durchschlagen . . .
	sterben	sterben verhungern helfern designierten jährlich zutritt meistens amerikanern irakern fünfte tod planeten industriegebieten fonds dramatisch us-regierung
Nouns	*knie*	asiatischen zusammengestellt zufügt kniefall knie knien apartheid-regime rechtsanspruch
	euro	ausübt euroraums euroumstellung euro-raums euros euro-länder euro-ländern euro-zusammenarbeit euro euro-raum euroländer euro-system . . .

Figure 5.1: A list of German candidate verb and noun lemmas and their inflected forms extracted automatically through alignment and transitive linkage. List of candidate inflections is unordered either in terms of frequency or in terms of dictionary precedence.

5.2 Pilot clustering

In this section, I discuss the results obtained from the pilot clustering experiments.

5.2.1 TIGER Treebank Corpus

The TIGER Treebank (Brants and Hansen, 2002) corpus was used as the evaluation corpus on which to test the initial lemmatization schemes. The corpus, which is currently at version 2.1, is a collection of German newspaper text gathered from the Frankfurter Rundschau and consists of app. 900,000 tokens. It is annotated with POS tags and lemmata for terminal nodes and has been manually annotated for syntactic information. Since this corpus is a full-text corpus, it provided a window into how well the scheme induced from one domain would translate to another.

5.2.2 Results

There were 193582 word types in the German portion of the Europarl corpus. From this set W_t, 15945 lemma candidates were induced after applying the culling outlined in section 3. These lemma candidates were mapped to a total of 29056 candidate inflected forms, an average of 1.8 inflectional candidates to a lemma candidate.

Evaluation was conducted using two separate measures. One was over the tokens observed in the TIGER corpus (Figure 5.1) and another was over types (Figure 5.2).

	ADJ	N	V	OVERALL
Precision	0.711	0.903	0.718	0.836
Recall	0.277	0.330	0.080	0.267
F-Score	0.399	0.483	0.144	0.405

Table 5.1: Scores by tokens and POS tag

	ADJ	N	V	OVERALL
Precision	0.711	0.795	0.840	0.772
Recall	0.822	0.899	0.463	0.874
F-Score	0.762	0.844	0.596	0.791

Table 5.2: Scores by types and POS tag

To evaluate type accuracy in this task, I used a measure similar to the

Jaccard distance between true and induced inflectional forms for a lemma. The precision of an individual clustering was defined as the size of the intersection between an induced set of inflectional forms and the standard set of inflectional forms divided by the size of the standard set. By summing the individual clustering precision figures over the entire set Λ of sets of inflectional forms I_i, and normalizing this by $N = |\Lambda|$, the precision was calculated as

$$\frac{1}{N}\sum_{I_i \in \Lambda} \frac{|I_i \cap I_g|}{|I_i|}$$

where I_g is a cluster from the gold TIGER Treebank and one that is defined to have at least one element in common with I_i.

Recall was defined similar to the above but divided by $|I_g|$ instead:

$$\frac{1}{N}\sum_{I_i \in \Lambda} \frac{|I_i \cap I_g|}{|I_g|}$$

These results are given in Fig. 5.2.

5.3 Two-constraint clustering

Here, I first lay out the evaluation metric used in Schone and Jurafsky (2000). Then I evaluate two baselines and present the results for the two-constraint clustering.

As with Schone and Jurafsky (2000), evaluation is conducted on CELEX. The CELEX lexical database (Baayen, Piepenbrock, and van H., 1993) has been built for Dutch, English and German and provides detailed entries that list and analyze the phonological, morphological and syntactic properties of word forms along with frequency information and orthographic variations. In the case of German, one subset of the data holds 51,728 stems with 365,530 corresponding wordforms or inflectional variants for an average of 7 inflected forms per stem for nouns, adjectives and verbs. Though it does not exhaustively list all possible inflections for the stems in its list, it is still the largest database of its kind that I know of.

5.3.1 Evaluation metric

Schone and Jurafsky (2000) define what they call a "conflation set". It is a set of word types which are related through either inflectional or derivational morphology. They calculate the sums of ratios for each model derived conflation set in relation to the conflation set in CELEX and calculate separate values that are *correct* ($\mathcal{C}$), *inserted* ($\mathcal{I}$), and *deleted* ($\mathcal{D}$):

$$\begin{aligned}
\mathcal{C} &= \sum_{\forall w}(|X_w \cap Y_w|/|Y_w|) \\
\mathcal{I} &= \sum_{\forall w}(|X_w - (X_w \cap Y_w)|/|Y_w|) \\
\mathcal{D} &= \sum_{\forall w}(|Y_w - (X_w \cap Y_w)|/|Y_w|)
\end{aligned}$$

where X_w is the model conflation set and Y_w is the standard conflation set.

However, $|Y_w|$ is only counted for those word types which have been observed in the training text. Next, precision, recall, and the f-score are defined as:

$$\begin{aligned} \mathit{precision} &= \mathcal{C}/(\mathcal{C}+\mathcal{I}) \\ \mathit{recall} &= \mathcal{C}/(\mathcal{C}+\mathcal{D}) \\ \mathit{f\text{-}score} &= \frac{2 \cdot \mathit{precision} \cdot \mathit{recall}}{\mathit{precision} + \mathit{recall}} \end{aligned}$$

The subset of the CELEX database that is used in this experiment lists inflectional variants of a stem as well as its part-of-speech. Every entry in this subset of CELEX is given a unique number that it shares with other entries that are inflectional variants. I use this data set to measure the current model's performance in terms of lemmatization as opposed to morphological clustering. I also use it to measure the performance of two baseline models.

5.3.2 Baseline

Two baselines are evaluated: (1) each word type forms its own cluster (2) a word type which is of length > 2 can constitute the centroid of a cluster and any other word types which include it as a prefix are considered members of the cluster. To provide an example of the second, if *ein* is the centroid, the words *ein, einbeziehung, eindeutig, eine, einem, einmal, einsatz,* ... will be members of this cluster. Table 5.3 tabulates the first baseline and Table 5.4 tabulates the latter.

Data-set	Recall	Precision	F-score
1K	41.4	100	58.5
2K	37.7	100	54.7
4K	34.4	100	51.1
8K	31.9	100	48.3
16K	30.9	100	47.2
32K	29.1	100	45.1
64K	27.8	100	43.5
128K	27.1	100	42.6
256K	26.3	100	41.6
512K	25.5	100	40.6

Table 5.3: Singleton cluster baseline.

In the singleton cluster baseline, I state the obvious that it is possible to get perfect precision by limiting the size of the clusters. It can be seen that recall drops monotonically as the size of the dataset is increased, but the decrease is most pronounced from the 1k to 16k dataset. While in no way conclusive, I surmise that the recall figures are indicative of the general stem to inflected form ratio for the respective datasets. For example, in the 1k dataset, every word type that occurs in the data has 2.41 inflected forms in the same set whereas there are 3.92 inflected forms in the 512k dataset. This conforms to the intuitive notion that more inflected variants of a word type would be observed as the size of the data is increased.

The prefix based baseline was implemented to measure the effect of favoring recall over precision. It is obvious that perfect recall could have been achieved if every word was placed in the same cluster. Also, the baseline was implemented assuming that the target language is suffixal in its inflectional

Data-set	Recall	Precision	F-score
1K	56.6	32.6	41.4
2K	56.1	25.8	35.4
4K	59.1	21.6	31.6
8K	60.2	15.0	24.0
16K	63.1	9.9	17.1
32K	67.2	6.9	12.5
64K	68.5	4.7	08.8
128K	71.0	3.3	06.3
256K	75.6	2.3	04.5
512K	80.8	1.3	02.6

Table 5.4: Prefix based baseline.

behavior, which German is to a certain degree. I surmise that recall could be higher if a similar baseline had been implemented for English.

5.3.3 Evaluation on CELEX

Though Schone and Jurafsky (2000) is used to evaluate the results, it must be mentioned that the results are not directly comparable. Their experiments were based on an English training set, a language with considerable differences compared to German in terms of morphosyntax. Also, their evaluation was based on a comparison with the CELEX suffix table for English and evaluation was conducted only for words which had a frequency of ≥ 10. Applying different thresholds on the vector space in their model, they obtain a consistent f-score in the range of 83 and 85. The evaluation here upon strictly defined inflection clusters could therefore be considered more rigorous.

In Figures 5.2˜ 5.5, I observe the effect of various theshold values for

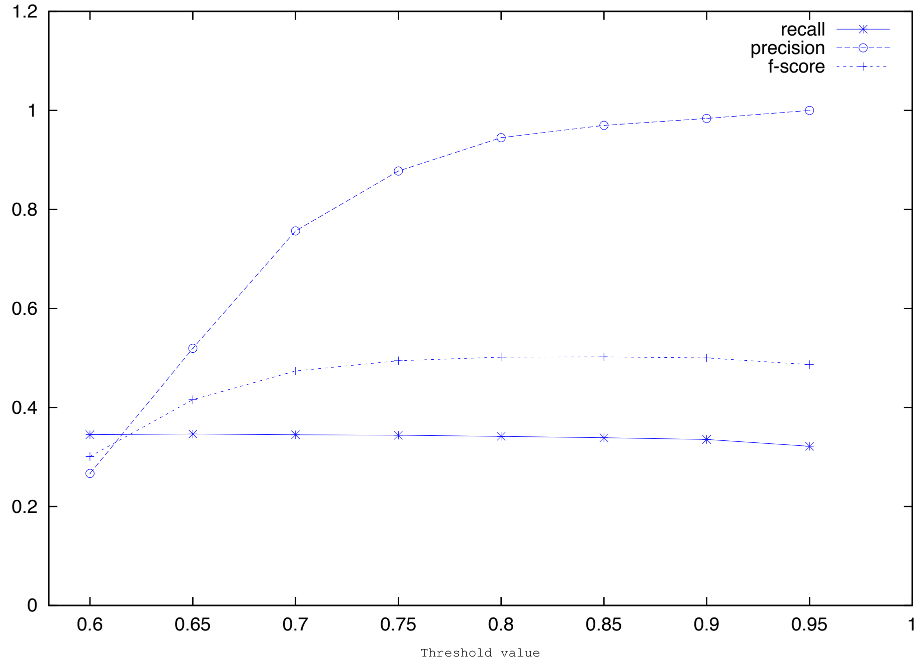

Figure 5.2: Learning curve of recall, precision and f-score as the value of θ_{sim} is increased in increments of 0.05. $\theta_{al} = 0.0001$ with 8K corpus.

θ_{al} and θ_{sim} upon the recall, precision and f-score of this model. No matter the changes in the value of θ_{al} and θ_{sim}, recall decreases by approximately two percentage points for the 8k dataset and by approximately four percentage points for the 64k dataset. The curves show that θ_{sim} has the greatest influence on precision as precision increases by almost 100 percentage points from worst to best in the 64k dataset and by about 70 percentage points in the 8k dataset. This is not to say that the alignment threshold has no effect upon performance. It does indicate that once alignment links are formed between the vocabularies, the threshold for inclusion in the clusters does not play as great a role. In Table 5.5, I tabulate performance for all datasets when the parameters are set

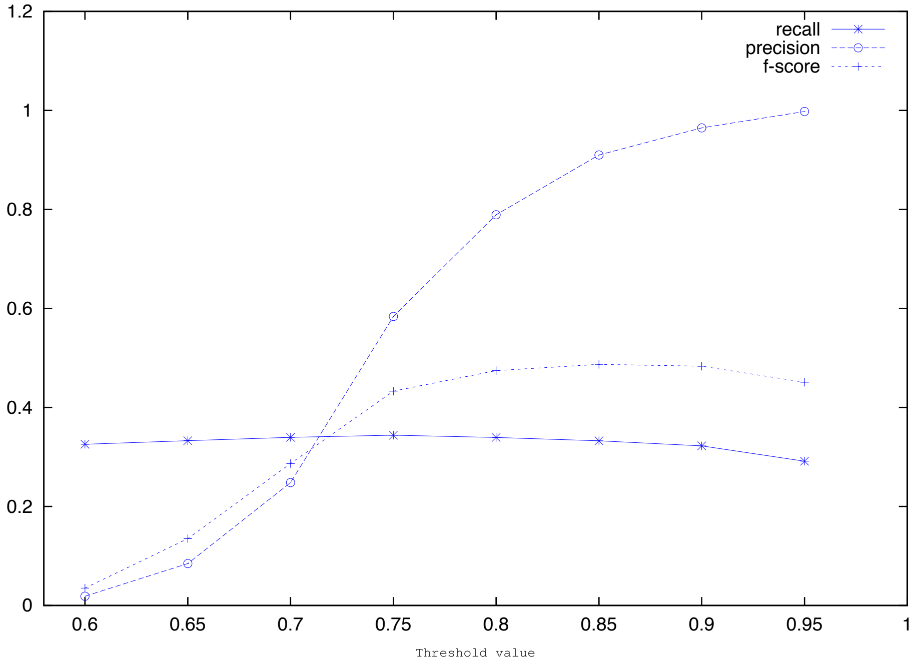

Figure 5.3: Learning curve of recall, precision and f-score as the value of θ_{sim} is increased in increments of 0.05. $\theta_{al} = 0.0001$ with 64K corpus.

to $\theta_{sim} = 0.88, \theta_{al} = 0.0001$.

While it seems that the consistency of the affect of alignment thresholds upon performance might be similar for most languages, I do not think it is equally valid to assume that changes in string similarity thresholds would show a similar learning curve across different languages. Exactly what effect it does have will require further examination over diverse languages.

The weakness of the current approach is in its low level of recall and in general its low cluster to word type ratio (Table 5.6). While recall does seem to increase steadily as more data is added, it is nowhere near the rates

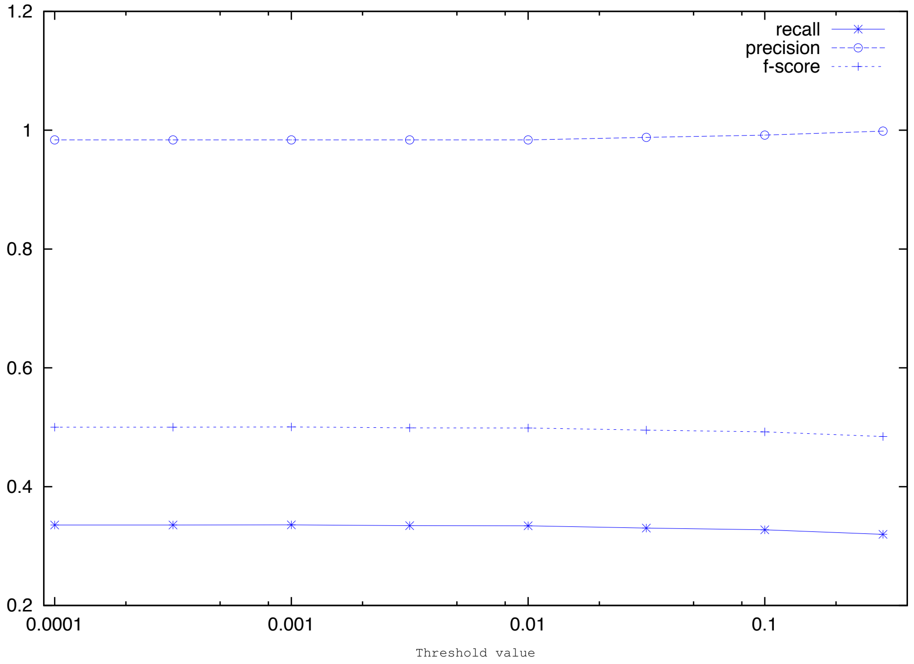

Figure 5.4: Learning curve of recall, precision and f-score as the value of θ_{al} is geometrically increased in multiples of $\sqrt{10}$. $\theta_{sim} = 0.88$ with 8k corpus.

suggested by our prefix based baseline (Table 5.4).

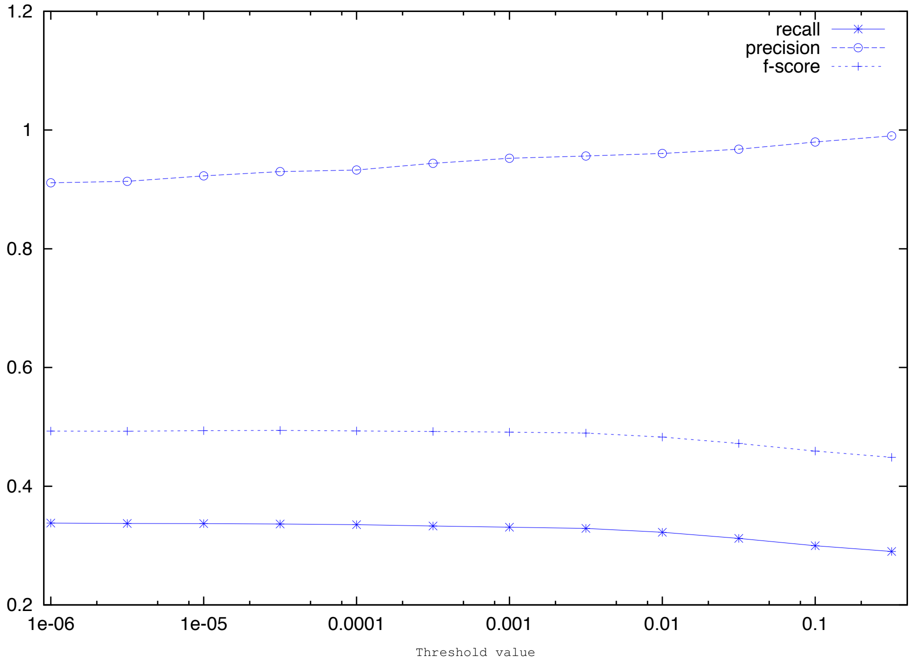

Figure 5.5: Learning curve of recall, precision and f-score as the value of θ_{al} is geometrically increased in multiples of $\sqrt{10}$. $\theta_{sim} = 0.88$ with 64k corpus.

	Recall	Precision	F-score
1K	41.7	99.6	58.8
2K	38.5	99.8	55.6
4K	35.6	98.2	52.3
8K	33.5	98.3	50.0
16K	33.2	97.8	49.5
32K	32.8	96.5	48.9
64K	32.5	95.3	48.5
128K	33.5	93.3	49.3
256K	34.6	89.7	49.9
512K	36.0	87.3	51.0

Table 5.5: Performance at $\theta_{sim} = 0.88, \theta_{al} = 0.0001$

	8k	32k	128k	512k
words	2373	5964	13965	30139
clusters	2274	5532	12272	24786
ratio	1.043	1.078	1.138	1.216

Table 5.6: Number of word types to cluster ratio by POS category.

Chapter 6

Segmentation

In this section, I use the results of the previous lemmatization model to generate a set of inflectional prefixes and suffixes for the target language for separate POS categories using a directed acyclic graph. First, I discuss related work, then present the approach.

6.1 Related work

Unsupervised monolingual morphology segmentation is a topic that has been tackled many times in the literature (Goldsmith, 2001; Sassano, 2001; Goldwater, 2006; Hammarström, 2006; Creutz and Lagus, 2007). Though such approaches generally manage to provide relatively reliable segmentation schemes with precisions between the ranges of 0.8 and 0.9, it is difficult to generalize beyond the segmentation of individual word types to how they relate to the POS categories in a given language or its syntax.

There is also a considerable amount of literature on finite state machines and morphology (Koskenniemi, 1983; Karttunen, Kaplan, and Zaenen, 1992; Dhonnchadha, Pháidín, and Genabith, 2003; Pretorius and Bosch, 2003). Also, Freitag (2005) was an attempt to build a finite state machine that describes the morphology of a text using unsupervised means. While the former approach is precise at the cost of implementation time, the latter has minimal labor costs but the results do not advance beyond any loose notion of morphology.

6.2 Directed acyclic graph segmentation

I build a directed acyclic graph of the words for each cluster that was induced. The generation process of the state machine itself creates segmentations. First, I build a standard acyclic finite state machine based on the character strings inside the cluster so that the states are single characters. Then I perform a horizontal merge operation where we collapse sequential states if and only if the preceding node has one outgoing transition and the following node has only one incoming transition. The resulting state is a concatenation of the characters representing the states. Next, I perform a vertical merge operation where any and all outgoing transitions from a given state are merged if and only if the character sequences representing the states are identical.

The implementation is an acyclic FSM, but there are some additional constraints on how it inserts or creates new nodes. Some of the constraints are:

1. Favor graphs with fewer nodes and arcs. This constraint ensures that contiguous substrings will generally have no or fewer intervening nodes

between each other. This makes analysis of the graph easier and less time consuming for any recursive procedures and, in general, tends to "bunch up" the stems towards the middle. Figures 6.3 and 6.4 illustrate this constraint with a nonsensical example of the cluster *ein einem eieineinemem*:

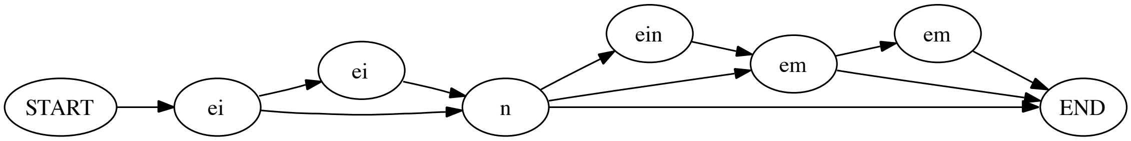

Figure 6.1: 8 nodes and 11 arcs. The paths for *ein* and *einem* have intervening nodes and are distributed across more nodes than is minimally possible

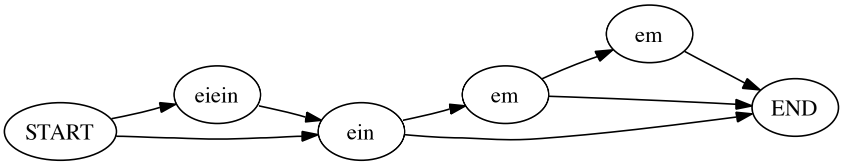

Figure 6.2: 6 nodes and 8 arcs. *ein* and *einem* have no intervening nodes and the minimum number of nodes necessary to represent the cluster is used

2. Disfavor arcs that share nodes with single, isolated characters. This constraint also has the effect of reducing the number of nodes and arcs, but erases some substring commonalities between strings. It is used to favor node sharing between stems but not between affixes (on the assumption that stems will be longer than affixes in general, and that there might be affixes which share a single character but no more). An

example with the cluster *verehrte geehrte* is shown in figures 6.3 and 6.4.

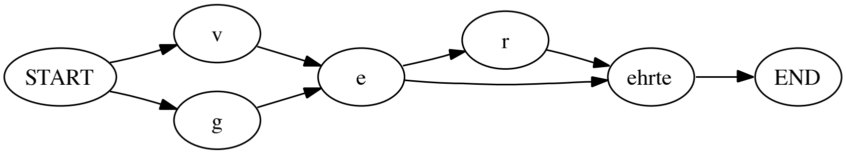

Figure 6.3: The *e* in *ver* and *ge* are shared

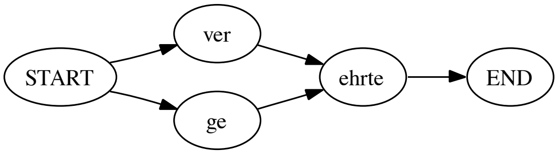

Figure 6.4: *ver* and *ge* are split across different nodes

Then the segments are categorized according to this simple heuristic:

- If a segment has an outgoing transition to the terminating (empty) state but does not have an incoming transition from the beginning (empty) state, then it is a suffix
- If a segment has an incoming transition from the beginning state but does not have an outgoing transition to the terminating state, then it is a prefix.

6.3 Data

Output from the task defined in section 4.3.2 is used in this stage.

6.4 Segmentation results

Tables 6.0(a) and 6.0(b) list the five most common suffixes and two most common prefixes for the entire set of corpora when $\theta_{al} = 0.01$, $\theta_{sim} = 0.8$. It can be seen that smaller datasets for the previous tasks can result in less reliable results. A comparison of the results between the prefixes and suffixes hints that German is probably suffixal rather than prefixal in its inflectional morphology.

(a) Five most common suffixes with counts by POS category and corpus size. Thresholds set at $\theta_{al} = 0.01$, $\theta_{sim} = 0.8$

	1K		2K		4K		8K		16K		32K		64K		128K		256K		512K	
V			tzes	1	t	2	en	6	en	11	n	24	t	33	t	59	t	83	t	129
			t	1	n	2	t	5	n	10	en	19	n	29	en	56	en	81	en	122
			n	1	e	2	n	4	t	8	t	18	en	29	n	42	n	54	n	65
			e	1	santrag	1	e	4	e	6	ung	9	e	14	e	22	e	27	e	34
					en	1	ung	1	er	2	e	9	ung	9	ung	19	ung	24	ung	29
N	n	1	en	3	n	4	en	11	en	29	en	65	en	101	en	144	en	220	en	316
	en	1	e	2	e	4	n	8	n	21	n	53	n	82	n	129	n	194	s	305
			s	1	r	2	s	7	e	16	s	36	s	64	s	121	s	176	n	274
			n	1	en	2	e	7	s	13	e	30	e	40	e	69	e	106	e	154
					äge	1	in	3	es	7	es	11	es	18	es	27	es	38	es	60
J	n	5	n	7	n	15	n	27	n	62	n	102	n	164	n	245	en	364	en	535
	r	4	en	5	en	10	en	23	en	33	en	64	en	123	en	210	n	321	e	419
	e	4	e	4	e	8	e	17	e	25	e	55	e	97	e	158	e	274	n	406
	m	2	t	2	r	7	s	7	s	17	s	38	r	47	r	85	s	141	s	207
	en	2	r	2	s	5	r	6	r	13	r	31	s	43	s	77	r	124	er	190

(b) Two most common prefixes with counts by POS category and corpus size. Thresholds set at $\theta_{al} = 0.01$, $\theta_{sim} = 0.8$

	1K		2K		4K		8K		16K		32K		64K		128K		256K		512K	
V			grunds	1	s	1	versuch	1	be	2	ge	5	ge	8	ge	13	ge	17	ge	23
			erinner	1	erinner	1	soll	1	versuch	1	e	2	zu	3	ab	4	an	8	ein	9
N			staate	1	vorschl	1	änderungsantr	1	be	2	be	3	be	5	v	3	be	5	an	6
			rats	1	vor	1	zusammen	1	zusammen	1	an	3	welt	2	mit	3	grund	4	be	5
J	w	1	wu	1	zu	1	w	2	ver	3	ver	4	ge	11	ge	12	ge	19	ge	25
	vorl	1	wo	1	we	1	ge	2	un	2	ge	4	ver	4	be	9	be	14	be	20

Table 6.1: Affixes induced by segmentation of clusters with DAG and tabulating those which occur most frequently.

Chapter 7

Conclusion

I have presented an approach that induces the inflectional morphology of a target language using bitexts. This approach does not require the substantial amount of text used by monolingual unsupervised approaches, potentially allowing it to be applied to data collected from endangered language documentation projects where most of the text is aligned and glossed. Furthermore, even with reduced amounts of data, it is able to induce informative affixes and prefixes for separate part-of-speech categories, paving the way for the generation of unobserved forms.

This has been made possible by bringing together three disparate, yet simple elements in a novel way such that the semantic, syntactic and surface components of a target vocabulary are considered within the model.

The first element is the alignment and transfer method which uses the rich linguistic resources developed for one language and projects this information to an aligned target text with some degree of noise. This allows the induction of more restricted, informative clusters that conform to linguistic notions of inflection and POS categories.

The second element is a string similarity measure that makes no assumptions about the morphological properties of the language in question by favoring longer common subsequences, regardless of contiguity and absolute position of characters.

The final element is a morphologically informed directed acyclic graph that is used for discovering boundary affixation patterns without the space requirements of the trie that is often used in morphology induction. Further experiments and improvements will be aimed at discovering infixation patterns and phonological distance between the orthographic characters in a language, patterns that would be very challenging to discover with a trie.

References

Airio, Eija. 2006. Word normalization and decompounding in mono- and bilingual ir. *Inf. Retr.*, 9(3):249–271.

Al-Onaizan, Y., J. Curin, M. Jahr, K. Knight, J. Lafferty, D. Melamed, F. Och, D. Purdy, N. Smith, and D. Yarowsky. 1999. Statistical machine translation. In *Final Report, JHU Workshop*, Baltimore, MD.

Baayen, R. H., R. Piepenbrock, and Rijn van H. 1993. *The CELEX lexical data base on CD-ROM.* Linguistic Data Consortium, Philadelphia, PA.

Bannard, Colin and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 597–604, Morristown, NJ, USA. Association for Computational Linguistics.

Baroni, Marco, Johannes Matiasek, and Harald Trost. 2002. Unsupervised discovery of morphologically related words based on orthographic and semantic similarity. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning*, pages 48–57, Morristown, NJ, USA. Association for Computational Linguistics.

Bilenko, Mikhail, Raymond Mooney, William Cohen, Pradeep Ravikumar, and Stephen Fienberg. 2003. Adaptive name matching in information integration. *IEEE Intelligent Systems*, 18(5):16–23.

Brants, Sabine and Silvia Hansen. 2002. Developments in the TIGER annotation scheme and their realization in the corpus. In *Proceedings of the Third Conference on Language Resources and Evaluation (LREC 2002)*, pages 1643–1649, Las Palmas.

Brent, Michael R. 1999. An efficient, probabilistically sound algorithm for segmentation andword discovery. *Mach. Learn.*, 34(1-3):71–105.

Cohen, W., P. Ravikumar, and S. Fienberg. 2003. A comparison of string distance metrics for name-matching tasks. In *Proceedings of the IJCAI-2003*, pages 73–78.

Creutz, Mathias and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4(1):3.

Curran, James R and Stephen Clark. 2003. Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-03)*.

Deerwester, Scott C., Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

Dhillon, Inderjit S., Subramanyam Mallela, and Dharmendra S. Modha. 2003. Information-theoretic co-clustering. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98, New York, NY, USA. ACM.

Dhonnchadha, Uí, Caoilfhionn Nic Pháidín, and Josef Van Genabith. 2003. Design, implementation and evaluation of an inflectional morphology finite state transducer for irish. *Machine Translation*, 18(3):173–193.

Diab, Mona. 2000. An unsupervised method for multilingual word sense tagging using parallel corpora: a preliminary investigation. In *Proceedings of the ACL-2000 workshop on Word senses and multi-linguality*, pages 1–9, Morristown, NJ, USA. Association for Computational Linguistics.

Drábek, Elliott Franco and David Yarowsky. 2005. Induction of fine-grained part-of-speech taggers via classifier combination and crosslingual projection. In *Proceedings of the ACL Workshop on Building and Using*

Parallel Texts, pages 49–56, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Fredkin, Edward. 1960. Trie memory. *Commun. ACM*, 3(9):490–499.

Freitag, Dayne. 2005. Morphology induction from term clusters. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 128–135, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Goldsmith, John. 2001. Unsupervised learning of the morphology of a natural language. *Computational Linguististics*, 27(2):153–198.

Goldwater, Sharon. 2006. *Nonparametric Bayesian Models of Lexical Acquisition*. Ph.D. thesis, Brown University.

Goldwater, Sharon and David McClosky. 2005. Improving statistical mt through morphological analysis. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 676–683, Morristown, NJ, USA. Association for Computational Linguistics.

Hacioglu, Kadri, Bryan Pellom, Tolga Ciloglu, Ozlem Ozturk, Mikko Kurimo, and Mathias Creutz. 2003. On lexicon creation for turkish lvcsr. In *Proceedings of the 8th European Conference on Speech Communication and Technology (Eurospeech)*, pages 1165–1168, Geneva, Switzerland.

Hajič, Jan, Jan Hric, and Vladislav Kuboň. 2000. Machine translation of very close languages. In *Proceedings of the sixth conference on Applied natural language processing*, pages 7–12, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Hammarström, Harald. 2006. A naive theory of affixation and an algorithm for extraction. In *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology at HLT-NAACL 2006*, pages 79–88, New York City, USA, June. Association for Computational Linguistics.

Jacquemin, Christian. 1997. Guessing morphology from terms and corpora. In *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 156–165, New York, NY, USA. ACM.

Jain, A. K., M. N. Murty, and P. J. Flynn. 1999. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323.

Jones, Michael N. and Douglas J. Mewhort. 2007. Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, 114(1):1–37, January.

Karttunen, Lauri, Ronald M. Kaplan, and Annie Zaenen. 1992. Two-level morphology with composition. In *Proceedings of the 14th conference on Computational linguistics*, pages 141–148, Morristown, NJ, USA. Association for Computational Linguistics.

Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit 2005*.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54, Morristown, NJ, USA. Association for Computational Linguistics.

Kornai, András. 1996. Extended finite state models of language. *Natural Language Engineering*, 2:287–290.

Koskenniemi, Kimmo. 1983. *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production.* Ph.D. thesis, University of Helsinki, University of Helsinki, Dept of General Linguistics, Hallituskatu 11-33, SF-00100 Helsinki 10, Finland.

Kraaij, Wessel and Renée Pohlmann. 1996. Viewing stemming as recall enhancement. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 40–48, New York, NY, USA. ACM.

Krovetz, Robert. 2000. Viewing morphology as an inference process. *Artif. Intell.*, 118(1-2):277–294.

Larkey, Leah S., Lisa Ballesteros, and Margaret E. Connell. 2002. Improving stemming for arabic information retrieval: light stemming and cooccurrence analysis. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–282, New York, NY, USA. ACM.

Lee, Young-Suk, Kishore Papineni, Salim Roukos, Ossama Emam, and Hany Hassan. 2003. Language model based arabic word segmentation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 399–406, Morristown, NJ, USA. Association for Computational Linguistics.

Minnen, Guido, John Carroll, and Darren Pearce. 2001. Applied morphological processing of English. *Nat. Lang. Eng.*, 7(3):207–223.

Navarro, Gonzalo. 2001. A guided tour to approximate string matching. *ACM Comput. Surv.*, 33(1):31–88.

Nießen, Sonja and Hermann Ney. 2001. Toward hierarchical models for statistical machine translation of inflected languages. In *Proceedings of the workshop on Data-driven methods in machine translation*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Och, Franz Josef and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ozdowska, Sylwia. 2006. Projecting POS tags and syntactic dependencies from English and French to Polish in aligned corpora. In *EACL 2006 Workshop on Cross-Language Knowledge Induction*.

Pang, Bo, Kevin Knight, and Daniel Marcu. 2003. Syntax-based alignment of multiple translations: extracting paraphrases and generating new sentences. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 102–109, Morristown, NJ, USA. Association for Computational Linguistics.

Pretorius, Laurette and Sonja E. Bosch. 2003. Finite-state computational morphology: An analyzer prototype for zulu. *Machine Translation*, 18(3):195–216.

Sassano, Manabu. 2001. An empirical study of active learning with support vector machines for Japanese word segmentation. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 505–512, Morristown, NJ, USA. Association for Computational Linguistics.

Schone, Patrick and Daniel Jurafsky. 2000. Knowledge-free induction of morphology using latent sematic analysis. In *Proceedings of CoNLL-2000 and LLL-2000*, pages 67–72.

Schone, Patrick and Daniel Jurafsky. 2001. Knowledge-free induction of inflectional morphologies. In *NAACL '01: Second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies 2001*, pages 1–9, Morristown, NJ, USA. Association for Computational Linguistics.

Shin, Jung H., Young S. Han, and Key-Sun Choi. 1996. Bilingual knowledge acquisition from Korean-English parallel corpus using alignment method: Korean-English alignment at word and phrase level. In *Proceedings of the 16th conference on Computational linguistics*, pages 230–235, Morristown, NJ, USA. Association for Computational Linguistics.

Trost, Harald. 1990. The applicaton of two-level morphology to non-concatenative german morphology. In *Proceedings of the 13th conference on Computational linguistics*, pages 371–376, Morristown, NJ, USA. Association for Computational Linguistics.

Wu, Dekai. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–403.

Xu, Jinxi and W. Bruce Croft. 1998. Corpus-based stemming using cooccurrence of word variants. *ACM Trans. Inf. Syst.*, 16(1):61–81.

Yarowsky, David and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *NAACL '01: Second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies 2001*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Yarowsky, David, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *HLT '01: Proceedings of the first international conference on Human language technology research*, pages 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

Yarowsky, David and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In *ACL '00: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 207–216, Morristown, NJ, USA. Association for Computational Linguistics.