UNSUPERVISED MORPHOLOGICAL SEGMENTATION FOR STATISTICAL MACHINE TRANSLATION

by

Ann Clifton
B.A., Reed College, 2001

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
in the School
of
Computing Science

SIMON FRASER UNIVERSITY
Summer 2010

APPROVAL

Name: Ann Clifton

Degree: Master of Science

Title of thesis: Unsupervised Morphological Segmentation for Statistical Machine Translation

Examining Committee: Dr. Torsten Moeller
Chair

Dr. Anoop Sarkar, Associate Professor, School of Computing Science
Simon Fraser University
Senior Supervisor

Dr. Fred Popowich, Professor, School of Computing Science
Simon Fraser University
Supervisor

Dr. Greg Mori, Assistant Professor, School of Computing Science
Simon Fraser University
Examiner

Date Approved:

Abstract

Statistical Machine Translation (SMT) techniques often assume the word is the basic unit of analysis. These techniques work well when producing output in languages like English, which has simple morphology and hence few word forms, but tend to perform poorly on languages like Finnish with very complex morphological systems with a large vocabulary. This thesis examines various methods of augmenting SMT models to use morphological information to improve the quality of translation into morphologically rich languages, comparing them on an English-Finnish translation task.

We investigate the use of the three main methods to integrate morphological awareness into SMT systems: factored models, segmented translation, and morphology generation models. We incorporate previously proposed unsupervised morphological segmentation methods into the translation model and combine this segmentation-based system with a Conditional Random Field morphology prediction model. We find the morphology aware models yield significantly more fluent translation output compared to a baseline word-based model.

Keywords: natural language processing, statistical machine translation, morphology generation, segmentation-based translation

To my family

Acknowledgments

It is my pleasure to express my profound gratitude to my supervisor, Dr. Anoop Sarkar. Not only does he hold expert command of the subject of Natural Language Processing, but he communicates this knowledge will great skill, enthusiasm, and patience, without which this work would not have been possible.

I would also like to thank Dr. Fred Popowich for his outstanding teaching and keen advice during the course of my Master's program, as well as his astute feedback on this thesis.

I also very much appreciate the thoughtful consideration of my examiner, Dr. Greg Mori, who provided insightful comments on this thesis, and whose excellence as a teacher has been invaluable.

This course of study has been made richer and more enjoyable by the help and camaraderie of my cohort and labmates (past and present), to whom I am very grateful, particularly Winona, Shawn, Baskaran, Milan, Majid, Reza, Ajeet, and Yudong.

Finally, I would like to offer my endless gratitude to John for his unshakeable love and support, and to my family for their unfailing encouragement and kindness.

Contents

Approval ii

Abstract iii

Dedication iv

Acknowledgments v

Contents vi

List of Tables ix

List of Figures xi

1 Introduction 1

1.1 Challenges of morphological complexity for SMT 2

1.2 Motivation to study this problem 4

1.3 General Approaches 6

1.3.1 Factored Models 6

1.3.2 Segmented Translation 7

1.3.3 Post-Processing Morphology Generation 7

1.4 Our approach and contributions 8

1.5 Preliminaries - Data, Moses, MERT, and BLEU 9

1.5.1 Data 9

1.5.2 Moses 9

1.5.3 Evaluation - BLEU, WER and TER 10

1.5.4 Optimization - MERT 12

1.6 Thesis Outline 12

2 Factored models 14

2.1 Phrase-Based Translation 14

2.2 Moses decoder 19

2.3 Factored models 19

2.4 Using Factored Models for Finnish MT 22

2.4.1 Experiments 23

2.4.2 Results and Analysis 26

2.5 Chapter Summary 28

3 Segmented Translation 29

3.1 Segmented translation 29

3.2 The Task 31

3.3 Overall Experimental Setup 32

3.4 Supervised Segmentation 34

3.5 Unsupervised segmentation 35

3.5.1 Unsupervised segmentation - Morfessor 35

3.5.2 Unsupervised Segmentation - Experimental Setup 40

3.6 Experiments and Analysis 42

3.6.1 Experimental Corpora Statistics 42

3.6.2 Results and Analysis 42

3.7 Chapter Summary 47

4 Morphology generation 49

4.1 Morphology generation 49

4.1.1 Generative Models - HMM 50

4.1.2 Discriminative Models - MEMM and CRF 52

4.1.3 Maximum entropy models for morphology generation 53

4.2 Factored Translation, Segmented Translation, and CRF Post-Processing Morphology Prediction - Model Comparison 57

4.2.1 Factored Model vs. CRF Post-Processing Model 57

4.2.2 Factored Model and Post-Processing Model vs. Segmented Model . . 61

4.3 Using CRFs for Morphology Generation 62

4.4 CRF Morphology Prediction Using Supervised Segmentation 64

4.4.1 Experimental Setup 64

4.4.2 Experiments 65

4.4.3 Results and Analysis 65

4.5 CRF Morphology Prediction Using Unsupervised Segmented Translation . . . 66

4.5.1 Experimental Setup 67

4.5.2 Experiments 68

4.5.3 Results 69

4.5.4 Morphological Fluency Analysis 71

4.6 Chapter Summary 77

5 Conclusion **79**

5.1 Thesis Summary 79

5.2 Contribution 80

5.3 Future Research 80

6 List of Morphological Abbreviations **83**

Bibliography **85**

List of Tables

2.1 Corpus Explicit Suffix Factor Percentage 26

2.2 Factored Model Scores 26

3.1 Corpus Ratios Against English Parallel Text. Sup refers to the model trained on the Omorfi supervised segmentation, while Un- denotes the Morfessor unsupervised segmentation. 5k means the model was trained on just the top 5,000 most frequently occurring words in the training data set, while full denotes the full corpus. PPL indicates the perplexity threshold used. L-match denotes the segmented corpus with additional splitting based on the longest-suffix-match heuristic, while F-match denotes that split with the most-frequently-seen-suffix-match. 43

3.2 Segmented Model Scores. Words means that the output was evaluated in whole-word form, while Segs indicates that the segmented output was evaluated before morph re-stitching against a segmented version of the baseline. Uni indicates the segmented BLEU score without consideration of unigrams. 43

3.3 Morphs in Phrase Table. RegSeg denotes the unsupervised segmentation trained on the 5,000 most frequent words in the corpus, with a perplexity threshold of 100. SplitSeg denotes this same model with additional longest-match suffix splitting. Morph refers to those phrases in the phrase table that included a segmented form, while Hanging Morph refers to those phrases that were bounded by a productive word-internal morph. 45

3.4 Phrases Used in Translation 45

3.5 Nominative Case Outputs 48

4.1 Supervised Prediction Model Feature Set 65

4.2 Supervised Prediction Model Scores 66

4.3 Unsupervised Stemmed MT Model BLEU Scores Against a Stemmed Reference. Seg5k refers to the segmentation trained on the 5,000 most frequent words in the training data; fullSeg means that it was trained on the full data set. PPL denotes the perplexity threshold used. 68

4.4 Unsupervised Prediction Model Feature Set 69

4.5 Model Accuracy 70

4.6 Model Predictions 70

4.7 Unsupervised Prediction Model Scores 71

4.8 Translation Tokens with Morphological Analyses 71

4.9 Model Accuracy: Morphological Constructions. Frequency refers to the construction's average number of occurrences per sentence (correctly or otherwise), also averaged over the various translations. The constructions are listed in descending order of their frequency in the texts. The highlighted value in each column is the most accurate with respect to the reference value. 72

5.1 Summary Model Scores: two versions of BLEU, with WER and TER. 80

List of Figures

1.1 BLEU Scores vs. Training Vocabulary Size Across Languages (Koehn, 2005) 5

2.1 Unidirectional Alignment (Koehn et al., 2007) 15

2.2 Bidirectional Alignment (Koehn et al., 2007) 16

2.3 Alignment Matrix (Koehn et al., 2007) 16

2.4 Factored Model Translation-Generation Pipeline 24

3.1 Segmented Translation Model Pipeline 33

3.2 Hierarchical Segmentation Example (Creutz and Lagus, 2005) 38

3.3 Morph Length vs. Frequency 41

4.1 HMM 51

4.2 MEMM 52

4.3 A Label Bias Example 55

4.4 CRF 56

4.5 CRF for Morphology Prediction. In this example, the observations are the stems with POS-tags, and the states are the morphology tags. 58

4.6 Post-Processing Model Translation-Generation Pipeline 63

Chapter 1

Introduction

Many of the Statistical Machine Translation (SMT) techniques we have today were originally developed on English and closely related languages. The central claim of SMT is that we can create models that learn to translate between languages in an unsupervised manner, using large parallel texts in the source and target languages. However, the focus on translation into English has some problems that become apparent when we apply such techniques to languages that are structurally quite dissimilar to English. Traditional SMT techniques tend to assume the word as the basic unit of analysis, an assumption that can be problematic outside of English. Such is the case for SMT for morphologically complex languages.

In linguistics, languages are classified according to a scale of morphological complexity. Morphemes refer to the minimal units of meaning in language; morphological complexity refers to the degree to which languages use bound morphemes, typically to express grammatical or derivational relations. English, with its low ratio of morphemes per word, is typologically classified as an isolating language, instead largely relying on word order to express syntactic relations. For example, while we do find words inflected with bound morphemes in English (e.g., 'inflect-ed'), we are more likely to find words that consist of a single free morpheme (e.g., 'single'). Morphologically complex languages, on the other hand, can express a wealth of information using multiple morphemes that constitute what we conventionally think of as a single word. This can be seen in the Turkish word 'çöplüklerimizdekiledenmiydi':

(1)	çöp	+lük	+ler	+imiz	+de	+ki	+ler	+den	+mi	+y	+di
	garbage	+AFF	+PL	+1p/PL	+LOC	+REL	+PL	+ABL	+INT	+AUX	+PAST

'Was it from those that were in our garbage cans?'

(Wintner, 2002)

The above example is taken from Turkish, a canonical example of a morphologically complex agglutinating language. Languages with rich morphological systems are classified according to how morphemes interact; 'agglutinative' languages tend to concatenate multiple discrete morphemes, while 'fusional' languages tend to conflate multiple atoms of meaning into a single affix. Spanish is an example of a fusional language, as seen in the conjugation of the verb 'estudiar' ('to study'):

(2) estudiasteis
study+PAST/2p/PL
'you all studied'

In this example, the morphological suffix cannot be further broken down into meaningful constituents denoting tense, person, and number separately.

Since typical word-based MT systems treat words as atomic units, 'sing' and 'sang,' 'singer,' and 'singers' are all treated as completely different words. So while this may not present an overwhelming obstacle in a morphologically poor language like English, the problem becomes quite serious when a stem can take a large number of morphologically inflected forms. In order to address the poor performance of standard SMT techniques on languages with morphological systems we look to go beyond the word-based analysis assumption and use sub-word morphological information; for many languages, a large part of the task of automatic translation is to get this morphology right. While this may be relatively trivial in an isolating language, it is far more complex and important in a language with rich morphology; not only does morphological correctness dictate the fluency of the output, but the morphology also does much of the work of expressing syntactic relations in the sentence and many aspects of word or phrase meaning.

1.1 Challenges of morphological complexity for SMT

Languages with rich morphological systems present significant hurdles for SMT, most notably:

- Languages with a lot of morphology tend to have freer word order compared to isolating languages, as word order is not as syntactically salient. This means that traditional n-gram statistics may be less informative.

- Words in morphologically complex languages are frequently composed of multiple morphemes inflecting a single or compound word stem. While there may be a limited stem and morphological lexicon, since these are productively combinable, this can lead to a combinatorial explosion of surface forms. Thus, any given surface form may occur quite infrequently in a text, leading to a significant data sparsity problem. This increases the number of out-of-vocabulary words in a morphologically complex text as well as makes text word occurrence statistics less robust.

- Translation between a morphologically rich language and one with little morphology suffers from the problem of source-target asymmetry. From a non-inflecting source language that expresses most syntactic relations lexically or through word order, the MT model must perform significant manipulation of the input data to derive correct inflecting-language output. Lacking straightforward word correspondences between the two languages, the translation model requires more complex lexical fertility and distortion relations[1].

- A simple but still important difficulty is that many morphologically complex languages tend to have been relatively unstudied. This means that we lack the wealth of resources available for languages like English with which to build better MT systems. In general, the more data on which a MT system is trained, the more accurately it performs. In addition to monolingual or parallel textual resources, other vastly helpful resources include corpora that has been annotated for syntactic, semantic, or other information. It also includes analytic resources that can generate supplemental levels of information on a text or language, such as sentence parsers or word analyzers. Of the previous work done on introducing morphological information to SMT, much has been built upon supervised morphological analysis used to train the model. However, these tools require human annotation and are limited in their coverage; moreover, for many languages, they simply do not exist.

[1] Fertility and distortion are common components of alignment between languages. Fertility denotes the number of words that can map to a single word; distortion refers to the displacement of words in another language.

- Another challenge lies in the automatic evaluation of MT systems. The current standard evaluation measures consider translation aspects such as the number of correct n-grams of words in the translated text against a human-translated reference text (Papineni et al., 2002). Since words in these measures are treated as unanalyzed strings, this of course means that for a complex word comprised of multiple morphemes, getting any single piece of inflection wrong discounts the entire word, preventing it from contributing any gain to the translation score. Other less widely used measures evaluate the translation quality by the edit distance in words between the translation and the reference. While some are supplemented with the ability to back off to word stems to find a match, this dictionary-based utility is only available for a few languages with the necessary resources for linguistic analysis. In addition, since morphologically complex languages tend to have freer word order, different orderings of the words in a sentence may be equally correct, but by edit distance measures the reordering would be penalized.

1.2 Motivation to study this problem

- Languages such as English with quite limited morphology represent only a small fraction of the world's languages. By incorporating morphological information into the SMT model and moving away from the English-like/lexical-based assumptions underlying many current MT approaches, we can dramatically increase the number of the world's languages for which machine translation applications are effective. This promises to improve the state of the art, from which to create more generalizable, robust techniques and translation tools.

- Another aspect of the task worth noting is the role it can play in the preservation of marginalized or endangered languages. For example, many languages of the Americas have rich morphological systems and comparatively few speakers. Opening up the possibility of creating effective automated tools for such languages promises to be a means to help decrease their marginalization.

For this thesis, we use Finnish as our target language. Finnish was chosen because with its rich morphology system it exemplifies many of the problems such languages present for SMT. This was demonstrated in the MT Summit shared task (Koehn, 2005), which compared the performance of state of the art systems across eleven European languages. The shared task took as training data the Europarl corpus, consisting of an approximately one million sentence parallel corpus in each of the eleven languages. This corpus was used to train MT systems pairing each of the languages; in a comparison of translation scores for each of the languages into and from English, the Finnish system performed the most poorly out of all the languages included in the study. Figure 1.1 plots MT performance against training data vocabulary size; we see that as the vocabulary size increases, the translation scores go down. Finnish, with the largest vocabulary out of all the languages in the evaluation, had the worst MT scores by a very significant margin, despite the fact that the test data sets were the same in content and in size.

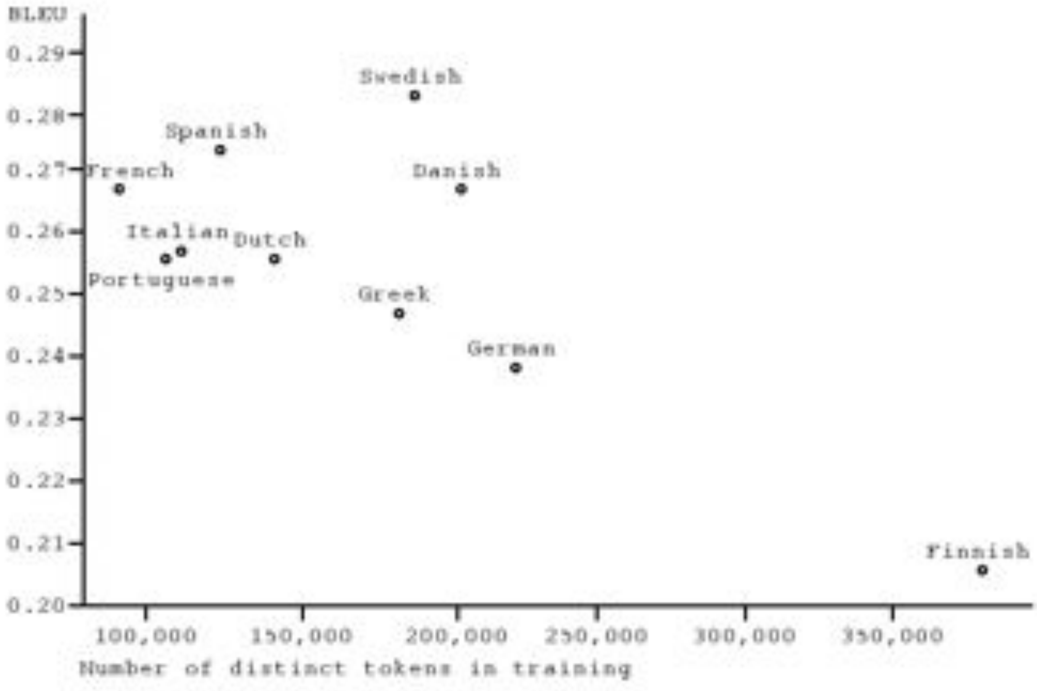


Figure 1.1: BLEU Scores vs. Training Vocabulary Size Across Languages (Koehn, 2005)

Moreover, translation into Finnish from English proved far more difficult than the other direction. It was generally the case that translating into English was easier that translating from English into a more morphologically complex language. We hypothesize that this is because while many morphological categories are expressed lexically in English (for example, morphological case frequently corresponds to English prepositions), there are many inflectional categories which are not required to be observed in English at all, or only rarely (such as mood). Thus, the translation task into English loses some information from Finnish, whereas the opposite direction requires information generation[2].

Another reason to take Finnish as our target is that there has been less research done on SMT for its typological family. As noted earlier, morphological systems are classified between agglutinating and fusional; of the current literature, it is far more common to see work on SMT for fusional languages such as German or Spanish than on agglutinative languages like Turkish or Finnish, which present a different and frequently more complex pattern of morphological behavior.

1.3 General Approaches

The idea that morphological richness may create challenges for SMT is not new. In the seminal paper (Brown et al., 1993) on alignment techniques for translation, there is discussion of using morphological analysis to generalize over inflected word forms, though only for French and English, languages with relatively simple morphological systems. The work that has been done thus far using morphological information in machine translation can be grouped roughly into three main categories, which this chapter will discuss. We refer to these as factored models, segmented translation, and morphology generation.

1.3.1 Factored Models

The first approach we discuss is the use of factored models (Koehn and Hoang, 2007), (Yang and Kirchhoff, 2006), (Avramidis and Koehn, 2008). Factored models are referred to as such because they factor the phrase translation probabilities over additional information annotated to each word. In the case of applying these models to the challenge of morphologically complex data, they allow for text to be represented on multiple levels of analysis, so that when a word is encountered that was unseen in the training data, the model may

[2] According to the Google MT Group (personal communication) increasing the amount of English-Finnish parallel data leads to better translation when the target is English, but improvements are not as good when Finnish is the target.

back off to a more general and frequently seen form for which it has gathered statistics in the training data. A possible drawback to this approach is its tendency toward reliance on linguistically motivated segmentation and morphological information, which again is a significant weakness for a group of languages which tend to lack this type of resource. A more significant problem with this approach, however, is that it still performs phrase-based translation on the word level. This means that any morphological information is tied to the word within the phrase and is generated on the basis the word alone, and thus does not allow for productive morpheme combinations to be incorporated into the model.

1.3.2 Segmented Translation

The next approach we examine, segmented translation, performs morphological analysis on the morphologically complex text for use in the translation model (Brown et al., 1993), (Goldwater and McClosky, 2005), (de Gispert and Mariño, 2008). Each morpheme output from the segmentation is treated as a separate word; this segmented version of the text is used to train the MT system. The intuition behind the various methods used within this approach is that it may not only unpack complex forms into simpler, more frequently occurring components, but that it will increase the symmetry between the amount and type of lexically realized content in the source and target languages. There are a few drawbacks of this approach, first in deriving the segmentation. While supervised segmentation methods have high segmentation accuracy, they are expensive, rare, and tend to lack full coverage. Unsupervised methods on the other hand, while practical and extensible, may be inaccurate or under-segment a language with a high degree of morphological complexity. Moreover, the correctness of the segmentation aside, it is an open question what degree of segmentation is most useful for the particular task of SMT. In the literature, this approach tends to suffer from either naively segmenting all words in the text, whether it is useful to do so or not, or else adopting a language specific heuristic for selectively using segmentation that is not necessarily generalizable beyond that particular language pair.

1.3.3 Post-Processing Morphology Generation

The last major approach we examine is morphology generation (Minkov, Toutanova, and Suzuki, 2007), (Toutanova, Suzuki, and Ruopp, 2008). This approach exploits the relative simplicity of translating stemmed text, and leaves the morphology to be handled by a separate generation model in postprocessing. The generation model takes the stemmed output of the MT system and predicts the inflection sequence for the stems in a sentence based on bilingual lexical and syntactic features of the parallel texts. The main drawbacks to this approach stem from the fact that it removes the morphological information from the translation model for some word types. This means that it presupposes a high quality stemmed translation model, and can be a problem for those languages in which morphology expresses lexical content in the parallel text.

1.4 Our approach and contributions

In this thesis, we propose to address the problem of morphological complexity in MT using phrase-based translation models, the current state of the art for SMT. Within this phrase-based framework, we explore techniques adapted from three major approaches. We first examine the use of factored models, which allow multiple levels of representation of the data from the most specific (surface word) level to more general levels of analysis such as lemma, part-of-speech, or morphological category. The second approach investigated in this thesis is segmented translation, wherein the words of the complex target language are segmented prior to training the MT system, and the segments are treated as words by the translation model. The last approach removes morphology from the translation model, instead translating on a stemmed version of the original text, and generates morphology for the MT output in post-processing.

Within each of these approaches, we focus on unsupervised segmentation methods to derive the morphological information supplied to the MT model in order to create maximally extensible models for languages with scant hand-annotated resources. Putting linguistically motivated segmentation evaluation aside, we examine what forms of non-language-specific segmentation are most useful particularly for the translation task, and use these segmented forms in conjunction with factored translation and morphology generation models.

In all three approaches, we attempt to use the morphological information in a way that is as language-general as possible, avoiding tactics that specifically apply to Finnish. Rather than focusing on a few linguistically motivated aspects of Finnish morphological behavior that may be problematic for MT, we seek to develop techniques for handling morphological complexity that can be generally applicable to languages whose morphology presents challenges to MT. With this in mind, we propose a method of combining segmented translation models with Conditional Random Field (CRF) based morphological generation in post-processing.

- We show that integrating morphological information into the translation model in a way that treats morphologically analyzed segments as lexical items improves translation scores into the morphologically complex target.
- We find that unsupervised segmentation methods can be used to inform the MT model, thus making our accurate morphologically aware model independent of expensive and rare supervised morphological analysis tools.
- We show that using a CRF morphology generation model in conjunction with a segmented translation model improves translation fluency into the morphologically complex target.

1.5 Preliminaries - Data, Moses, MERT, and BLEU

The following section describes the basic translation model, tuning method, and evaluation measure that we used in all the experiments described in this thesis.

1.5.1 Data

For all of the models built in this thesis, we used the Europarl corpus (Koehn, 2005) English-Finnish training, development, and test data sets. The training data consists of an approximately 1.2 million word parallel corpus, while the development and test sets were each 2,000 sentences long. Some parallel sentences in this corpus had vastly different lengths from one language to the other, which causes problems for MT system training. So to make this data set compatible with our MT system, we filtered the training data for only sentences of 40 words or less. This reduced the overall size of the data set to 986,166 sentences.

1.5.2 Moses

In all the experiments conducted in this thesis, we used the Moses[3] phrase-based translation system (Koehn et al., 2007), 2008 version. We trained all of the Moses systems herein using

[3] http://www.statmt.org/moses/

the standard features: language model, reordering model, translation model, and word penalty; in addition to these, the factored experiments called for additional translation and generation features for the added factors as noted above. We used in all experiments the standard settings: a hypothesis stack size 100, distortion limit 6, phrase translations limit 20, and maximum phrase length 20. For the language models, we used SRILM 5-gram language models (Stolcke, 2002) for all factors.

1.5.3 Evaluation - BLEU, WER and TER

As mentioned in Chapter 1, the evaluation measures used for measuring the quality of a translation warrant discussion. Since it is impractical to have access to human judgement of translation on a quick and regular basis, evaluation techniques were developed to approximate this judgement automatically. The most common standard measure is the BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002) score, which has been shown to be closely correlated with human judgments of translation quality. BLEU performs a comparison over the whole corpus based on the number of ngrams from the reference translation that appear correctly in the MT output translation. By factoring in the scores for each order of ngram, the BLEU method captures both content accuracy and fluency; where unigrams reflect the extent to which the reference captures all the information, longer ngrams reflect fluent, well-constructed sentences. Scoring a candidate translation on the basis of ngrams allows for multiple possible orderings of a translation to be deemed equally valid. This approach was designed with an eye to the fact that there are typically many ways to translate a sentence, and is more robust if multiple references are provided. Thus, in order to make the evaluation compatible with multiple references without requiring a candidate translation to include all the different phrasings of the same input, this measure captures precision but sacrifices recall.

The candidate translation's ngram precision score is modified to make sure that it does not get credit for repeating an ngram more frequently than it was seen in the reference. This is known as *clipping* the ngram counts so as not to exceed their maximum in the reference. Using these clipped ngram counts, the modified ngram precision score is calculated as the clipped counts of the candidate ngrams that match the reference ngrams added over each sentence, divided by the number of ngrams in the candidate test corpus:

$$p_n = \frac{\sum_{C \in Candidates} \sum_{ngram \in C} Count_{clip}(ngram)}{\sum_{C' \in Candidates} \sum_{ngram' \in C} Count(ngram')}. \qquad (1.1)$$

where C is the the set of sentences in the candidate translation.

BLEU does not explicitly account for recall, however, it can capture recall to some degree when supplied with a large number of reference translations. In order to compensate for this, BLEU includes a brevity penalty to keep the scores of translations that are composed of reference words but are shorter than the reference from scoring artificially highly. To get an intuition for why this is necessary, we include the following example:

(3) Candidate: of the

(4) Reference: It is the practical guide for the army always to heed the directions **of the** party. (Papineni et al., 2002)

In this example, the candidate translation is clearly very poor, but since it is composed of elements found the reference, and these appear the same number of times in the candidate and reference, its unigram precision is $\frac{2}{2}$ ('of' and 'the'), and its bigram precision is $\frac{1}{1}$ ('of the'). To keep this type of short translation from getting a perfect score, then, the brevity penalty is included, which decreases the candidate's score proportionally to how much shorter it is than the reference. The brevity penalty (BP) is computed as:

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r, \\ e^{\frac{1-r}{c}} & \text{if } c \leq r. \end{cases}$$

Then the BLEU score is:

$$\log \mathrm{BLEU} = \min(1 - \frac{r}{c}, 0) + \sum_{n=1}^{N} w_n \log p_n, \tag{1.2}$$

where w_n are the positive weights for each order ngram, summing to one.

Since BLEU keeps track of the number of reference ngrams that appear in the system output translations, in theory the best score is 100%. In practice, however, scores are much lower. The best systems tend to achieve scores in the high 30% to 40% range, for example, (Chiang and Knight, 2009) scores as high as 40.6% BLEU on a Chinese to English translation task.

There are several different implementations available; in this thesis we use the Portage[4] 2007 version; however, we also report scores from the MT Workshop Shared Task[5] 2007 version as a sanity check.

[4] http://www.nrc-cnrc.gc.ca/eng/projects/iit/machine-learning.html

[5] http://www.statmt.org/wmt07/baseline.html

It is important to be able to judge how much small changes in BLEU scores actually matter. To this end, the scores are reported along with a confidence interval accounting for the degree to which random error may affect the final scores. As an additional means of judging scores, we use a bootstrapping-resampling method (Koehn, 2004) that computes the significance of the difference between two translations by repeatedly drawing sample sentences from each translation, evaluating them, and noting how frequently one translation is better than the other. If this happens over a certain percentage of the time, the results are deemed significant. In this case the significance threshold is 95% or higher, or a ‘p-value’ of 0.05 or lower.

Since BLEU is the current standard measure in most common use and arguably has the highest correlation with human judgement (Cer, Manning, and Jurafsky, 2010), we use this as our primary evaluation tool. However, we also include translations’ Word Error Rate (WER) (Neißen, Och, and Ney, 2000) and Translation Edit Rate (TER) (Snover et al., 2000). These measures are based on the edit-distance between a translation candidate and a reference. WER is an edit-distance measure that looks for the shortest sequence of insertion, deletion, and substitution operations to get from the candidate to a reference. TER also includes the operation of swapping of adjacent phrases, allowing for phrase reordering. These edit-distance based measures are intended to have a high correlation with human judgement with fewer references than BLEU, and penalize phrasal shifts less than BLEU. Unlike BLEU, for WER and TER, lower scores are better. Throughout this thesis we will report BLEU scores, but we will include WER and TER scores in model summaries.

1.5.4 Optimization - MERT

Many state of the art SMT systems such as those used in this thesis are optimized with respect to BLEU scores. The optimization is done using minimum error rate training (MERT) (Och, 2003). MERT trains the model weights using an objective function that maximizes the BLEU score of the MT output translation with respect to the reference translation.

1.6 Thesis Outline

The rest of the thesis will be organized as follows. Chapter 2 describes factored models and our investigation of using them with morphological analysis for translation into Finnish. In Chapter 3, we look at segmented translation, giving a detailed description unsupervised segmentation and our approach to this method in the translation model along with our experiments. Chapter 4 details the CRF model family, drawing a comparison between this and the other approaches examined in thesis, and our work using CRFs for morphology generation for finnish MT. Finally, in Chapter 5, we summarize the thesis and our conclusions and future work.

Chapter 2

Factored models

This thesis focuses on phrase-based translation models and the different ways they can be used to improve SMT with morphological information. Phrase-based models (Koehn, Och, and Marcu, 2003) as a general framework are the most commonly used in SMT and reflect the current state of the art. In this chapter, we look at using morphology with a particular subtype of phrase-based translation models: factored translation models. So, to better understand how these models work, we will begin with some background about phrase-based translation. We will then discuss the specializations that distinguish factored models, before turning to a description of how we used them for morphologically-aware MT into Finnish. Throughout this section, we refer to (Brown et al., 1993) and (Knight, 1999) for alignment and (Koehn, Och, and Marcu, 2003) (Koehn et al., 2007) for phrase-based translation.

2.1 Phrase-Based Translation

Phrase-based models translate on the basis of word phrases, rather than individual words (Koehn, Och, and Marcu, 2003). To learn these phrases and their translation probabilites between the source and target languages, the model starts with word-by-word alignments between each parallel sentence in the corpora.

In a general translation model, we are looking for the target sentence t that is the best translation of source sentence s, i.e., that maximizes $\Pr(t|s)$:

$$\hat{t} = \operatorname*{argmax}_t \Pr(t|s), \tag{2.1}$$

which, by Bayes' rule, we can reformulate as:

$$\hat{t} = \underset{t}{\operatorname{argmax}} \Pr(t) \Pr(s|t). \tag{2.2}$$

The first term, $\Pr(t)$, is just the likelihood of sentence t occurring in the target language on its own, so then to find the translation probability $\Pr(s|t)$, we use word alignments between the source and target sentences. For example, Figure 2.1 shows the aligned words of an English-French parallel sentence.

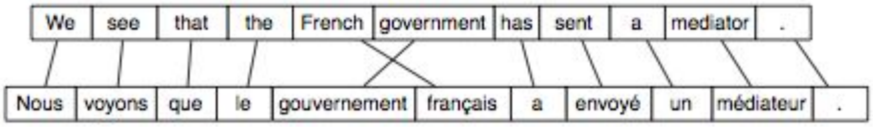


Figure 2.1: Unidirectional Alignment (Koehn et al., 2007)

Between any given source sentence $s_1^J = s_1, .., s_J$ and target sentence $t_1^I = t_1, .., t_I$, there may be many possible alignments, some more probable than others; we denote the probability of any particular alignment a as $\Pr(a|s_1^J, t_1^I)$. Since the probability of translating sentence s_1^J as sentence t_1^I is equal to the sum over all possible alignments between them, we can use this in conjunction with the Bayes rule to define $\Pr(s_1^J|t_1^I)$ in terms of alignments:

$$\Pr(s_1^I|t_1^J) = \sum_a \Pr(a, s_1^J|t_1^I), \tag{2.3}$$

which gives us a way to estimate the translation probabilities. It is important to note that while GIZA++ (Och and Ney, 2000), the alignment model used by Moses, allows multiple source words to be aligned to any given target word, it allows only one target word to be aligned to each source word. In order to overcome this limitation, the alignments are calculated in both directions, from source to target, and target to source.

These word-by-word alignments are learned using the expectation-maximization (EM) algorithm, which starts with an initial estimate of uniform probabilities for the word-to-word translation probabilities, $\Pr(t|s)$. These word translation probabilities are then used to compute the whole-sentence alignment probabilities for each sentence in the corpora, $\Pr(a|s_1^J, t_1^I)$. The sentence alignments are then in turn used to recalculate the word alignments; this process repeats iteratively until convergence.

Once the model has the whole-sentence word alignments, it can then extract phrase pairs for building the phrase-based translation model by creating a set of alignment points. To do this, the model uses the bidirectional alignments, starting with the intersection of the two. This set of alignment points on which both alignment directions agree is of high precision and represents a lower bound on the size of the set. From here, the set of alignment points is enlarged according to expansion heuristics, with an upper bound of the union of the two alignment directions. Figure 2.2 shows the bidirectional alignment between a parallel English-French sentence, while 2.3 gives its corresponding alignment matrix.

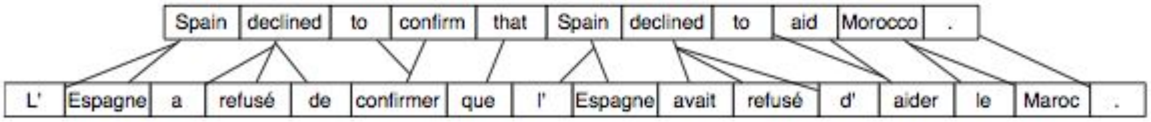
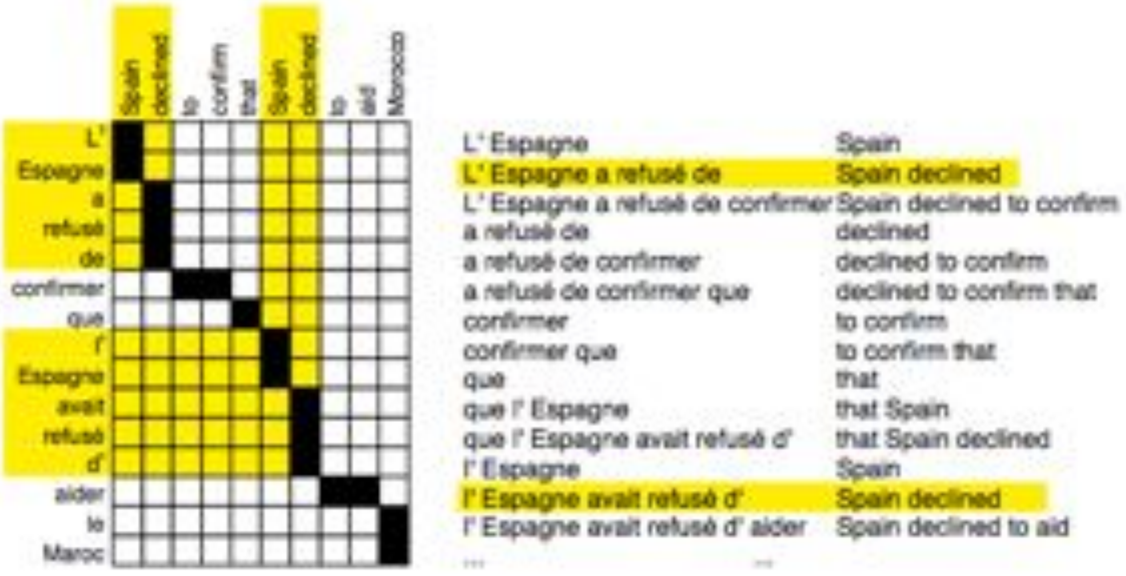


Figure 2.2: Bidirectional Alignment (Koehn et al., 2007)



L' Espagne	Spain
L' Espagne a refusé de	Spain declined
L' Espagne a refusé de confirmer	Spain declined to confirm
a refusé de	declined
a refusé de confirmer	declined to confirm
a refusé de confirmer que	declined to confirm that
confirmer	to confirm
confirmer que	to confirm that
que	that
que l' Espagne	that Spain
que l' Espagne avait refusé d'	that Spain declined
l' Espagne	Spain
l' Espagne avait refusé d'	Spain declined
l' Espagne avait refusé d' aider	Spain declined to aid
...	...

Figure 2.3: Alignment Matrix (Koehn et al., 2007)

The basic expansion heuristic used to build the set of alignment points (from which the phrases will be extracted), and the one we use in the systems built in this thesis, starts with an alignment point set from the intersection of the two alignments and then adds neighboring points from the directed alignments not already in the set. In this case, a point's 'neighbors' are defined as those points in the matrix which are vertically, horizontally, or diagonally adjacent; neighboring points from the directed alignments are added to the set until all candidate points have been added.

From this completed set, each legal phrase pair $(\bar{s}, \bar{t})$ that is consistent with the alignment is extracted. The phrases may be a single word, or a group of words, as long as the aligned words in the phrase are only aligned to each other, not any word beyond the phrase boundaries. These phrase pairs can then be used to compute the phrase translation probabilities for each direction, based on their overall counts in the training corpora:

$$\Pr(\bar{s}|\bar{t}) = \frac{\text{count}(\bar{s}, \bar{t})}{\text{count}(\bar{t})}. \tag{2.4}$$

From the phrase pair translation probabilities, we get the sentence translation probabilities, by splitting source sentence s_1^J into phrases $\bar{s}_1^P = \bar{s}_1, .., \bar{s}_P$. The component phrases are translated into P target phrases $\bar{t}_1^P = \bar{t}_1, .., \bar{t}_P$, giving the sentence translation probability as:

$$\Pr(t_1^I|s_1^J) = \prod_{p=1}^{P} \Pr(\bar{t}_p|\bar{s}_p). \tag{2.5}$$

When selecting the best translation output, we would like the model to take into account various other features of the data, in addition to these phrase-based sentence translation probabilities. So in order to make it easier to add other arbitrary feature functions into our model, we take the log of the sentence translation probabilities as well as any other features we wish to add, to combine them in a log-linear model:

$$f_{\mathcal{T}} = \log \Pr(t_1^I|s_1^J) = \sum_{p=1}^{P} \log \Pr(\bar{t}_p|\bar{s}_p), \tag{2.6}$$

where $f_{\mathcal{T}}$ is the translation feature. We can now add other feature components to this in a linear fashion. By taking the exponent, we can formulate the sum of all of our model features thus:

$$\Pr(t_1^I|s_1^J) = \frac{\exp(\sum_{i=1}^{n} \lambda_i f_i(t_1^I, s_1^J))}{\sum_{t'^I_1} \exp(\sum_{i=1}^{n} \lambda_i f_i(t'^I_1, s_1^J))} \tag{2.7}$$

where λ_i is the weight parameter for feature f_i. It is infeasible to enumerate all possible combinations to calculate the normalization constant in the denominator; we therefore use nbest-lists to approximate this in training the model weights. However, since the denominator is a constant, we can ignore this when finding the best translation for sentence s_1^J, given by:

$$\hat{t}_1^I = \operatorname*{argmax}_{t_1^I} \exp(\sum_{i=1}^{n} \lambda_i f_i(t_1^I, s_1^J)). \tag{2.8}$$

The components we wish to add in addition to the translation probabilities are the distortion model, the word penalty, and the phrase penalty, and the language model. The distortion model captures how much reordering of the target-side phrases should be done; the word penalty regulates the length of the target-side translation output, and the phrase penalty captures how many phrase pairs should be used for each sentence.

The language model is a feature based on the target-side language alone; it seeks to captures how likely it is that a given sequence of words will occur in that language. This model gives us word sequence probabilities computed from the counts of the ngrams (consecutive word sequences of length n) in the monolingual target text. Then for ngram of length n, its probability is given by:

$$\Pr(t_i|t_{i-1}, .., t_{i-n+1}) = \frac{count(t_i, t_{i-1}, .., t_{i-n+1})}{\sum_{t'_i} count(t'_i, t_{i-1}, .., t_{i-n+1})}. \tag{2.9}$$

This probability tell us how likely a word is given its immediate word history in the text.

Using this probability, we may add a language model feature to our log-linear model analogously to the translation features above:

$$f_{lm} = \log \Pr(t_1^I). \tag{2.10}$$

This feature can capture the fluency of the generated target translation sentence t_1^I.

2.2 Moses decoder

The decoder itself uses beam search and a pre-computed set of translation options to find the best translation for an input sentence. The translation options are pre-computed for the source phrases so as to avoid having to consult the entire phrase-table for each input string during translation. The translation options are stored with their probability, their target translation, and the first and last source words they cover.

In very simple terms, the algorithm works by generating translation hypotheses covering each phrase in the input sentence. Once a hypothesis is generated, it can be expanded to generate new hypotheses covering more words in the source sentence, extending the target output from left to right. At each step the probability of partial translation is updated; once all of the words of the input sentence are covered, the most probable translation hypothesis is kept.

In generating hypotheses, the search space can grow very large very quickly. The estimated upper bound is calculated[1] as $N \sim 2^{n_s}|V_t|^2$, where n_s is the number of source-side words and V_t is the size of the target-side vocabulary; we see that there is an exponential explosion corresponding to all the possible configurations of source-side words covered by a hypothesis.

The decoder does have some tactics to limit this, however. Hypothesis recombination decreases the search space by a small degree by combining those hypotheses that agree in terms of: the source words covered so far, the last two target words generated, and the end of the last source phrase covered. In addition to this, pruning methods are necessary; the decoder uses histogram and threshold pruning to limit the hypotheses.

2.3 Factored models

Like standard phrase-based models, the factored translation model combines the various components in a log-linear model composed of weighted feature functions over the individual components. Then under this model the translation between source sentence s and target sentence t has the same formulation as the standard phrase-based model:

[1] *http://www.statmt.org/moses/*

$$\Pr(t_1^I|s_1^J) \propto \exp(\sum_{i=1}^{n} \lambda_i f_i(t_1^I, s_1^J)), \tag{2.11}$$

where λ_i is the weight parameter for feature f_i. The components include the standard phrase-based model features; in the case of factored models, they also include a generation model well as multiple translation steps.

For both the factored and basic phrase-based models, the translation feature functions for a sentence pair are decomposed into phrase translations covering the sentence pair, which are given a scoring function, given by:

$$h_\tau(\bar{t}_p, \bar{s}_p) = \sum_{p=1}^{P} \Pr(\bar{t}_p|\bar{s}_p). \tag{2.12}$$

The factored model has the addition of separate feature functions corresponding to the multiple translation and generation steps. These feature functions can include factor-for-factor probabilities (instead of lexical translation probabilities, these can be POS-to-POS etc), as well as additional language models over the different factors. The generation feature functions are defined as scoring functions over the words (rather than phrases) summed over the sentence on target side only:

$$h_\gamma(\bar{t}_p, \bar{s}_p) = \sum_{p=1}^{P} \Pr(\bar{t}_p). \tag{2.13}$$

These scoring functions are computed from the generation table conditional probabilities between factors, and capture, in the task at hand, the likelihood of a morph occurring with a stem, based upon the stem-morph co-occurrences seen in the training data, as well as the language model over the stems or over the morphs.

It is the addition of these translation and generation steps that allow factored models access to both lexical and morphological information, drawn from training text annotated with various factors, such as part-of-speech, morphological features or word classes. With the annotated forms, standard phrase-based translation models are built on the specified factors, with available language models for each. This allows the model to consider not just translation probabilities between words but also between, for example (sequences of)

POS-tags. In addition, the overall model includes monolingual generation models of the annotated information which use a combination of factors for a given word to generate another factor, e.g., a stem with a morpheme may together be used to generate a surface form.

In this way, the translation process is decomposed into distinct steps of translation and generation. The translation model then has the option of translating the different levels of representation separately. To get a feel for how this works, consider an example factored translation task from Latin into English. Suppose the decoder encounters the factored Latin adjective 'filias|NN| ACC/PL' ('daughters'), and that this word has not previously been seen in the training data. Then in an ordinary model, the mapping from 'filias' to 'daughters' would have a zero count in the phrase table. However, in the case of the factored model, it has the option of using the additional factors of lemma and morphology. If some other inflected form of the lemma 'filia' has been seen as a translation for (some form of) the word 'daughter' in the training data, then the lemma mapping ('filia' to 'daughter') will have a nonzero probability, and so the model can hypothesize a translation for the lemma of the unseen word. Next, it looks to the morphology factor. Suppose there are other nouns that have been seen with the same 'ACC/PL' morphology factor, such as , 'puellās|puella| ACC/PL' and that this factored word maps to the factored form 'girls|girl|PL.' Since the translation probabilities have been computed for the morphological factors as well, the model has a nonzero probability for the Latin to English mapping between 'ACC/PL to PL,' so the morphology mapping hypothesis for our unknown input word becomes 'PL.' Now that we have a hypothesis for the English translated lemma and morphology, the generation step supplies the surface form. Using English text only probabilities generation model tells us that the lemma 'daughter' plus the the morphology 'PL' corresponds to the surface form 'daughters.' Thus, never having seen the input surface word before, the model is able to come up with the correctly inflected translation output. This formulation makes it possible for the model to prefer more specific forms (the surface word) when available but to back off to more general forms (lemma plus morphological category) otherwise.

Factored translation models have been used effectively on several language pairs, for translation both into and out of the more morphologically complex of the two. (Koehn and Hoang, 2007) were able to improve translation scores for translation from English into German, Spanish, and Czech, by representing the source text as a vector of the factors: word, lemma, POS, and morphological inflection. This work does not specify the source of their morphological analysis, however it does represent morphology in terms of linguistically motivated morphological categories. (Yang and Kirchhoff, 2006) achieved better translation scores for Finnish-English (22.0% BLEU on the test set) and German-English (24.8% BLEU on the test set) over a word-based baseline using unsupervised segmentation methods to supply sub-word information, but this was for translation in the direction of greater to lesser morphological complexity. Moreover, the unsupervised segmentation performed therein was for compound-splitting only, rather than morphology. While compound-word analysis is a similar task that also has the effect of reducing OOV words, it is a far less pervasive problem than a rich morphological system. For translation from English into Greek and Czech, (Avramidis and Koehn, 2008) use factored translation models to successfully approach target morphological variance. In this work, the authors enrich the morphologically poorer language before translation, annotating the source words with syntactic information expressed morphologically in the target. The syntactic source-side factors are then used in mapping to target-side morphology.

Under the factored approach, while the translation models are phrase-based, it is important to note that the generation models are word-by-word and based on the target language alone. Unlike the segmented model (see Chapter 3), which considers phrases of morphemes, all morphology is considered on a single-word basis. Thus, while including morphology as a translation factor creates a morphemes-only phrase-table, this can be used to capture sequences of inflections that are likely to co-occur in short phrases, but not productive morphology that can be recombined across phrase boundaries. Unlike the morphology generation model, morphological information is used to generate surface forms at the monolingual word level only, so while it may take into account inflectional phrases when selecting a morphological translation, it may not take advantage of the bilingual word or stem context when generating the fully inflected surface form.

2.4 Using Factored Models for Finnish MT

The following section describes our experiments using stemmed and lemmatized data and suffixes with factored phrase-based translation models an approach to morphologically aware SMT. First, a note about terminology: in this thesis, we use the term ‘stem’ rather than ‘lemma’ to differentiate between a generalized abstract lemma consistent for all of its surface reflexes and a stem that has been cut off from a word surface string after a certain point to strip away inflectional suffixes, but that has not been restored to a posited underlying form and may still display morphophonemic alternation such as vowel harmony.

Thus far, the majority of the work using morphology in factored models has depended upon some form of supervised morphological analysis to derive the factored representation and has been applied to more synthetic languages with far less morphological complexity than agglutinative Finnish. In general, factored models seem somewhat geared towards more general and limited morphological information than that derived from unsupervised morphological analysis; that is, morphologically decomposed factors might best be exploited by this model when represented as lemmas and a short vector of morphological category values, rather than actual stem and suffix strings denoting complex agglutinations of morphs. However, the level of morphological generality used in related work is achieved by using supervised morphological analysis, a limitation we wish to avoid.

We attempted to contrast the supervised and unsupervised segmentation methods for use in factored model training, but we found that the combinatorial explosion of the generation model using the set of morpheme category tag sequences made the model infeasible to train. As noted in Section 2.2, the complexity of the decoding problem is already exponential in the size of the input; when we expand the translation options available at each step in order of the generation factor options, the translation model can easily grow to an unmanageable size. We therefore restricted our examination of factored models to those trained on the unsupervised segmentations, which allow manipulation of the morpheme set in the generation model.

2.4.1 Experiments

We used the University of Tokyo POS tagger (Tsuruoka and Tsujii, 2005) for English part-of-speech tagging, and we used the Morfessor segmentation software (see Chapter 3.5) to supply stem and morphology factors to the translation model. On the English side, each word in the corpus appeared as a set of two factors, for the word and POS tag. On the Finnish side, words appeared as three factors, representing the word, stem, and suffix, for example:

varmaan|varma|an

In the case where the segmentation model did not split the word, the stem factor was left identical to the word, while the suffix factor was marked as null. We then decomposed the translation into separate steps for the word, stem, and morphological suffix, with two translation paths. The first translation path was simply to translate between surface forms on both sides. In the case where there was no available translation for a word, the model instead for a stem translation, using the alternate translation path, wherein the translation and generation steps were specified to interleave as follows:

1. translate source word to target stem
2. generate set of possible suffixes for the target stem
3. translate source POS tag to target suffix
4. generate surface form corresponding to stem and suffix

Figure 2.4 gives the full model overview.

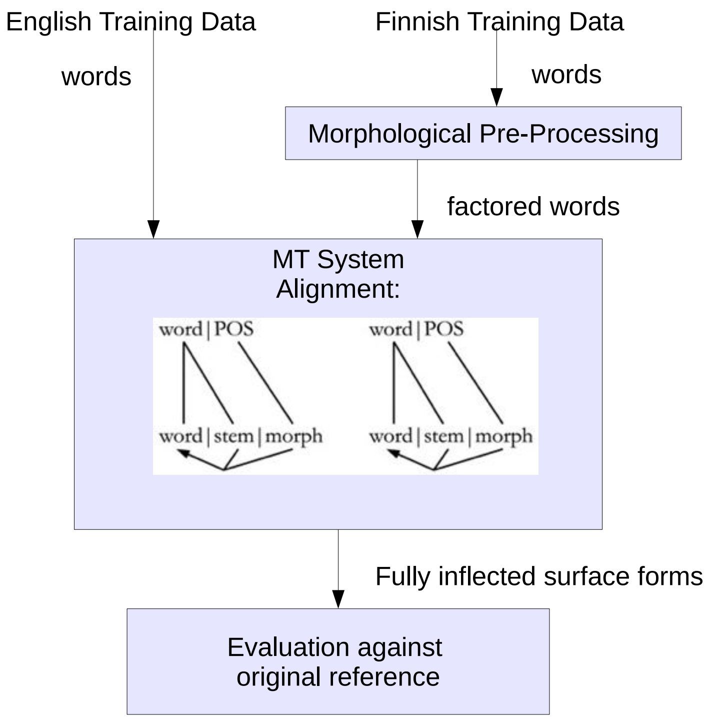


Figure 2.4: Factored Model Translation-Generation Pipeline

We have noted that one would have reason to question the translation step between

POS and suffixes. However, we reason that while English POS tags are not likely to display correspondences with morphological category values (i.e., that POS tags could correspond to, say, the singular versus the plural), they may correspond to the morphological category by distributional criteria. For example, case markers will be seen on nominals and adjective, but not verbs.

The segmentation model we used was trained on the full training corpus with the perplexity threshold set to 30. As will be described more fully in Chapter 3, this model performed nearly as well as the best performing segmentation model but had much higher segmentation coverage of the data. Using the entire word-final suffix set extracted from this Morfessor segmentation caused the same computational problem experienced with the supervised model and made it impossible to complete model training. Therefore, from the set of all available word-final suffixes, we chose a subset to use as morpheme factors for MT training. Looking for the most productively combinable forms, we selected the suffixes for the factor subset based on the number of different stems with which each had been seen in the training data, according to the segmentation model. We established that setting the number of different stems threshold of 150 allowed the model to capture 70.37 percent of all word-final suffixes in the training data, but still allowed for efficient MT model training. Despite the fact that imposing a co-occurring stem threshold of 150 reduced the set of suffixes to 81 out of the total 3,532 word-final suffix types in the training data, its suffix token coverage was proportionally much higher. Out of the total 3,708,802 word-final suffix tokens in the training data, this threshold captured 2,609,866.

Using this suffix subset, we created factored corpora for training two MT models. The first of these simply used the suffix subset to make decomposed factored representations for those words segmented by the segmentation model with word-final suffixes from the set. In the rare cases where a word was decomposed into more than two segments, all but the last were concatenated together to form the stem.

We created the second factored corpus to enrich the factored target corpus with as much morphological information as possible. As noted above, the primary translation path translates between surface inflected word forms when available, so adding morphologically decomposed representation above the original unsupervised segmentation should increase its coverage of the data set without losing information about and preference for whole words. We therefore split the already segmented corpus in an additional preprocessing step to make a decomposed factored form available for all words which had a match in the same reduced (most productive) suffix subset extracted from the Morfessor segmentation, as described above. When words had multiple candidate matches, the longest match was chosen in order to increase the model's exposure to more highly marked forms, resulting in fewer words with stem factors identical to the surface word and null morph factors. This pre-processing step significantly increased the number of tokens in the data with suffix predictions, though as we see in Table 2.1, since most words are not segmented in either case, there is still a majority of tokens with a null suffix factor.

Factored Segmentation	Non-Null Suffix Factor	Percent
FullSeg	7,960	18.96
FullSeg - Longest-match	16,241	38.68

Table 2.1: Corpus Explicit Suffix Factor Percentage

Table 2.1 gives the comparison between differently segmented corpora of null suffix factors for the development sets, out of the total 41,989 tokens. It shows that only a small percentage of the tokens in the data had available suffixes, though this amount is increased with the additional suffix-splitting pre-processing.

2.4.2 Results and Analysis

Table 2.2 show the resulting BLEU scores for the factored models, with their respective p-values, along with their WER, and TER scores. These measure the statistical significance of the result against the baseline by estimating the probability of the same system randomly generating the two translations; lower p-values indicate greater significance. We see that neither factored model is able to match the scores achieved by the simple word-based baseline, in either BLEU, WER, or TER.

Model	BLEU Score	p-value	WER	TER
Baseline	14.39±0.74		74.96	72.42
Factored	13.98±0.72	0.032	76.68	74.15
Factored with Longest-match	13.81±0.74	0.002	77.08	74.57

Table 2.2: Factored Model Scores

That the regular factored model does slightly better than the longest-match pre-processed model implies that additional morphological information when unnecessary may in fact do harm under a factored model. As noted earlier, factored models decompose the translation probabilities over the factors. If, by splitting a suffix from a word not decomposed by the segmentation model we create a less likely stem form, this may harm the overall likelihood of the word.

Overall, we hypothesize that the factored models do more poorly than the baseline for several reasons. For one, this may be an inherently difficult representational form for a language with the degree of morphological complexity found in Finnish. In this model, we have a binary division between stem and morphology, and all morphology that will be included in the model must be concatenated into the morphology factor. When multiple affixes are commonly attached to stems as in Finnish, we must either choose between excluding morphology from the morphology factor (and leaving as part of the stem or else undetermined), or else concatenating compound morphological elements into a single form. In these experiments, we took the former path, as the inclusion of all morphology in the morphology factor caused a combinatorial explosion that prevented model training. This forces a limit on the amount of morphology that the model can capture.

Another issue that may affect the poor performance of the factored models has to do with morphological pre-processing of the source-side text. Previous work that has found that factored models improve translation scores (Avramidis and Koehn, 2008) has used techniques that boost the source-to-target morphological factor correspondence. This is done by performing textual analysis on the morphologically poor language so as to add morphological annotation where not necessarily explicitly realized in the surface strings. We chose not to use this approach for two reasons: for one, it is unclear how to add English morphological information consistently based on Finnish morphemes. Furthermore, this approach would be inconsistent with our philosophy of avoiding language-specific techniques in favor of knowledge-poor methods that can be applied to arbitrary language pairs.

Moreover, by utilizing the generation model, factored translation models allow the decoder to pick morphs for stems based on all the morphs with which a stem has been seen in the training data. However, because the morphs are generated on a per-word basis within a given phrase, it does not allow for morphs to combine with new stems. This excludes productive morphological combination across phrase boundaries, as well as makes it impossible for the model to take into account any long-distance dependencies between morphemes. We discuss these issues further in a comparison to other models in section 4.2. We conclude from this result that it may be more useful for an agglutinative language to use morphology beyond the confines of the phrasal unit, and condition its generation on more than just the local target stem.

2.5 Chapter Summary

In this chapter, we investigate the use of unsupervised morphological segmentation for use in training factored translation models (Koehn and Hoang, 2007). We find that factoring the translations over the morphological information of this kind hinders the model's performance compared to the use of surface words alone for translation.

Chapter 3

Segmented Translation

In this chapter, we describe segmented translation models and how we use them for translation into Finnish. In our experiments, we use supervised and unsupervised segmentation methods to train a phrase-based MT system. We then test our system on the Europarl data set and show that the unsupervised segmentation MT model outperforms a word-based baseline model.

3.1 Segmented translation

In this approach, the morphologically complex data is segmented before training the translation model. This can be done by unsupervised or by supervised, dictionary-based means, and to greater or lesser degree depending on the nature of the particular approach. Supervised segmentation methods are typically based on a finite-state transducer compiled from a set of morpheme combination and transformation rules and rely on a predefined lexicon of stems and inflection classes. These methods have high accuracy but limited coverage and are used to derive linguistically motivated morphological analysis.

The unsupervised approach to word segmentation uses machine learning algorithms to perform segmentation, rather than training on hand-annotated data. This type of approach has the advantage of wider coverage, but is limited in the amount of morphological analysis it can return and is not constrained to any a priori notion of linguistic correctness in choosing a segmentation.

To illustrate the difference between these types of analysis, we give two analyses for the word ‘toimivaltaan’ (‘authority’); first from Omorfi(Pirinen and Listenmaa, 2007), a supervised morphological analyzer, then from Morfessor (Creutz and Lagus, 2005), an unsupervised model:

(5) toimia|VERB/ACT/INF/SG/ABL/POSS
toimi+ +valtaa+ +n

The former analysis returns the lemma (which is not an actual substring of the surface inflected form) of the verb meaning 'act' or 'work', along with a conflated string of derivational and grammatical morphological category values. It has parsed the substring '-lta-' as the ablative suffix, while the unsupervised analyzer has designated this substring as part of the middle segment; these segments, if translated separately, roughly mean 'action+ +power+ +GEN.'

An early example of the segmented translation approach takes advantage of morphological analysis to reduce corpus vocabulary size by generalizing over inflected forms for word stems. In (Brown et al., 1993), morphological analysis of both source and target is performed before training for verbs, nouns, adjectives, and adverbs. These words in the texts are then invertibly transformed into a lemma annotated with morphological information, with which they expect to reduce the French vocabulary by $\sim 50\%$ and English by $\sim 20\%$. Using the morphologically decomposed representations, they derive alignments with more robust statistics than those built on inflected surface forms alone.

Aside from the use of segmented translation to overcome data sparsity problem, much work in this vein has been geared towards increasing the degree of symmetry between source and target. (Popoviç and Ney, 2004) perform segmentation to reduce morphological complexity in a morphologically complex source text to translate into a more isolating target. Targeting verbs in Spanish-to-English, the authors transform the source text into either stem-POS representation or split the input into a stem and suffix. In the Serbian-to-English experiments, they use automatic methods to split off and discard suffixes from verb stems. These approaches yield a reduction in translation error rate for the English target. For Czech-to-English, (Goldwater and McClosky, 2005) lemmatized the source text and inserted into it a set of 'pseudowords' that they expected to have lexical reflexes in English, mildly improving the output translation scores.

One approach to using unsupervised segmentation makes use of multiple segmentation hypotheses in training the translation model (Dyer, 2009). In this approach, (most commonly applied to text segmentation for languages written with ambiguous or unmarked word boundaries, such as Korean and Chinese), multiple unsupervised segmentation or tokenization hypotheses are retained in a lattice. In this way, the model propagates segmentation uncertainty forward to the translation model (Chung and Gildea, 2009), rather than leaving it as an issue tackled earlier in the pipeline. The intuition motivating this approach is that the translation model may decide on the fly the best granularity of segmentation, and may alternate between coarser and finer grained hypotheses as yields the best translation/alignment.

As exemplified in the work described earlier, much of this work in the literature is performed using some form of linguistically motivated morphological analysis selectively applied based on some language-specific heuristic. A typical approach is to select a class of words which tend to exhibit the most morphological alternation and segment them for a particular inflectional category, for example, to identify verbs and slice off personal pronouns (de Gispert and Mariño, 2008) or to identify nouns and slice off case markers (Ramanathan et al., 2009). While these linguistically targeted techniques can aid in improving translation, we seek to develop techniques that are useful for more than just a specific language pair.

3.2 The Task

When investigating the use of morphological information in an MT model, there are several angles to consider, most notably granularity, linguistic accuracy, linguistic specificity, coverage, and extensibility.

Handwritten morphological analyzers such as the supervised method used herein, Omorfi (Pirinen and Listenmaa, 2007), can achieve high rates of linguistic accuracy, however such resources are not widely available for all languages of interest. In addition, since they are lexicon-based, they have intrinsically limited extensibility and coverage for varied training corpora. In a study (Lindén and Tuovila, 2009) using Omorfi to produce morphological analyses for inflected forms of new words, it achieved an F-score of 83 to 86 percent. However, Omorfi has for example, particular difficulty with named entities, which in morphologically complex languages are as productively inflected as any other nouns and contribute to the data sparsity problem when treated as unanalyzed strings.

Moreover, the segmentation model's linguistic accuracy may not result in the optimal segmentation for MT. (Chang, Galley, and Manning, 2008) show that for the task of Chinese word segmentation, a task similar to morphological segmentation, the linguistic accuracy of the segmentation does not necessarily improve MT scores. The authors found that the best segmentation for training an MT model was instead produced by optimizing the segmentation length for MT scores to one that was shorter than the gold standard linguistic segmentation.

While it cannot be taken for granted that greater accuracy in segmentation yields improved translation, neither can it be assumed that a more thorough segmentation aids in translation more than one that is coarser-grained. In contrast to the task of perfecting the morphological inflection in the target, the goal in segmentation for translation is instead to maximize the amount of lexical content carrying morphology, while generalizing over the information not helpful to improving the translation model.

The following example illustrates this intuition. In the case of translating the sentence, 'Should I sit down for a while in the big house?' should be translated as 'istahtaisin+ko (sit+COND-ACT-SG1-INTER) suure+sa (big+IN) talo+ssa (house-IN).' In this example, we see that the conditional, interrogative, and case markers all have lexical reflexes in the English parallel sentence; if we can match these content bearing morphemes to their lexical counterparts, then we may hope to improve the translation model. However, certain categories are either not distinguished or else expressed with word order, such as the active mood marker in the above example. The challenge present in learning which methods of segmentation are most useful to the MT task for a particular language pair is to maintain our goal of avoiding language-specificity. To accomplish this, we avoid reliance on the heuristic strategy of hand-picked morphological categories for inclusion in the translation model, though others have shown this to improve MT performance in specific language pairs. An example of this approach can be found in (de Gispert and Mariño, 2008), wherein the authors successfully employ linguistically targeted morphological analysis techniques such as the separation of enclitic pronouns from verbs. However, in addition to being highly specific to given language pairs, this type of strategy is simply not possible for many morphologically complex languages, which lack the morphologically annotated resources that are required to build and use such a model.

3.3 Overall Experimental Setup

We performed segmentation on the training data using both supervised and unsupervised methods. After the segmentation, word-internal morpheme boundary markers were inserted into the segmented text to be used to reconstruct the surface forms in the MT output. For example, 'vuodenvaihteeseen' becomes in the segmented training data 'vuoden+ +vaihte+ +eseen'. We then trained the Moses phrase-based system on the segmented and marked text, treating each segment as a word. After decoding, it was a simple matter to join together all adjacent morphemes with word-internal boundary markers to reconstruct the surface forms. We also tested joining together marked segments that were non-adjacent, affixing forms with a morph-initial marker to the preceding form, even when without a morph-final marker. This manner of joining morphs was aimed at capturing any uninflected base forms occurring in the text that can accept productive morphological phrases. Figure 3.1 gives the full model overview.

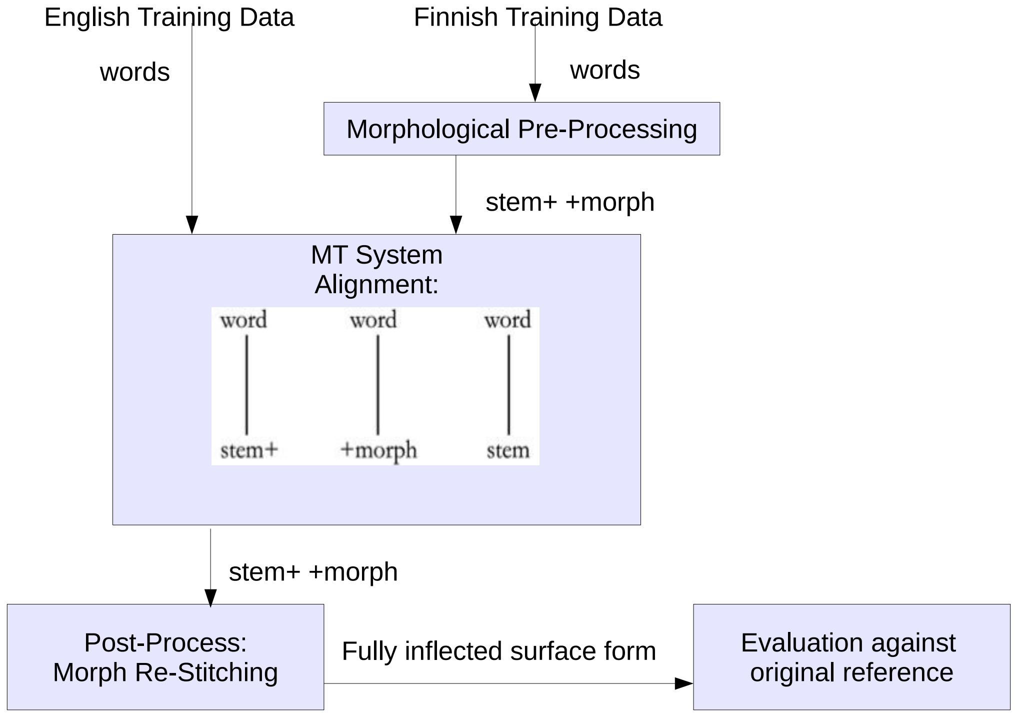


Figure 3.1: Segmented Translation Model Pipeline

We evaluated the test data in two forms: we measured the reconstructed test translation output against the original reference translation, as well as the translation output in segmented form against a segmented version of the reference. We did this to see how well the model was doing at translating the individual morphs, since this distinction is lost when evaluating at the word-level alone. For this second evaluation, we also measured the BLEU scores without consideration of unigrams. We did this to determine whether any score increase in the segmented versus word version was simply due to a larger number of correct unigrams, which is not by itself relevant to the goal of correctly combining productive morphology and increasing correct surface translation forms.

For our initial baseline, we trained a word-based model using the same Moses system with identical settings. For evaluation against segmented translation systems in segmented forms before word reconstruction, we also segmented the baseline system's word-based output.

3.4 Supervised Segmentation

Some hand-written morphological analyzers return a morphological parse, for example, the (National Research Council, Institute for Information Technology, 2005) Inuktitut morphological analyzer gives a morpheme-by-morpheme segmentation along with their respective meanings, as in this analysis for the word ‘tamaaniinnama’:

(6)

tama	+ani	+in	+nama
right-here-beside-me	+IN	+EXIST	+BECAUSE

‘because I am right here’

However, an open-source morpheme-by-morpheme morphological parser is not available for Finnish. Therefore, to obtain supervised segmentations, we used the Omorfi analyzer. This is a hand-written morphological analyzer which provides morphological analysis in the form of the word lemma followed by a sequence of concatenated morphological categories and their values, not associated with particular characters of the original inflected string. For example, when given the input word ‘tulevaisuudessa’, the analyzer returned ‘tulevaisuus/NOUN/SG/INE’, supplying values for part-of-speech, inflection class, number and case, but not a correspondence between these morphology values and the surface string characters.

Omorfi is based on a finite-state transducer and uses a predefined lexicon of lemmas, morphological categories, and declension classes to analyze fully inflected word forms. We used the segmentation output in order to find the point at which to segment the word strings in a straightforward fashion. The lemma is represented as the underlying (more general) uninflected form of the surface word, and may not be a substring of the surface word derived from it. A more exaggerated example of this is the Finnish word 'on' (meaning 'is'), which is the form of the verb 'to be' inflected for the categories of PRES/3P/SG. However, its lemma absent of morphology, is represented as 'olla.' In able to use analyzer-generated lemmas like these in our translation model, we would need to account for arbitrary and context-specific phenomena (such as vowel harmony or consonant gradation) that cause the lemma to change to some context-specific stem form so that it could be combined with morphology in a way that would result in viable surface forms. As with the example 'olla', the 'tulevaisuudessa' segmentation does not resolve tidily into a root form with a suffix concatenated onto the end, since 'tulevaisuus' is not an actual substring of 'tulevaisuudessa'. So instead we simply drew the segmentation boundary on the original word based on the length of the hypothesized lemma; words shorter than their lemmas were left unsegmented; otherwise, everything that occurred after the length of the lemma was treated as a suffix, resulting in the segmented output 'tulevaisuud+ +essa' for the example at hand. We did this to allow morph to be recombinable in the output and avoid undoing the transformation of the 'stem' morph into a linguistically motivated lemma form with an arbitrary relationship to the surface 'stem' morph. While this approach sacrifices some generality on the part of the lemmas, it increases the likelihood that the reconstructed MT output will result in well-formed surface strings.

3.5 Unsupervised segmentation

3.5.1 Unsupervised segmentation - Morfessor

In this thesis, to derive the unsupervised segmentations of the data, we used Morfessor (Creutz and Lagus, 2005). Morfessor uses minimum description length (MDL) and *maximum posteriori* (MAP) techniques to create a model optimized for accuracy with minimal model complexity to perform segmentation without the benefit of hand-segmented training data.

Morfessor works in a two-step process, beginning with the MDL-based algorithm, which, given an input corpus, defines a lexicon of morphs that can be concatenated to produce any word in the corpus. In linguistics, a morph refers to the surface realization of a morpheme in a given context, in contrast to a morpheme, which is an abstract unified representation of all context-dependent surface alternations the morpheme may take. Since this algorithm segments words into substrings, rather than hypothesized abstract classes, we refer to the segmented output as morphs. The MDL algorithm iterates over each input word, considering every possible split as a morph to be added to the lexicon, including the word as a whole, and selecting that split with the highest probability. The MDL algorithm considers possible morphs without regard to their context; it cares only for the frequency of the string in the text, not where it occurs within a word. This algorithm models words as HMMs having only one category, which emits the context-independent morph. This process continues recursively until the overall probability of the split corpus converges, yielding a segmentation based upon a flat lexicon of morphs without internal substructure. A consequence of this MDL approach is that frequent words are more likely to remain whole, while infrequent words are likely to be over-segmented. This follows from the representational conciseness of representing infrequent words by their frequent constituents in the morph lexicon and frequent words as a whole. This MDL segmentation is used to initialize the second phase of the model.

The Morfessor second phase (referred to as Categories-MAP model) reanalyzes the MDL segmentation into a recursive hierarchical morph lexicon using a greedy search MAP algorithm. This model phase represents words as HMMs, wherein the hidden states are latent morph categories. They define their categories as prefix, stem, and suffix, as well as a temporary non-morph 'noise' category and a word boundary category. The HMM is constrained to make the states context-sensitive: it prohibits the prefix category morphs to appear in the word-final position and suffixes to be word-initial; and it prohibits the transition directly from prefix to suffix, requiring an intervening stem.

The Categories-MAP framework used herein uses the prior probabilities of the morph lexicon along with the corpus data likelihood given the morph lexicon to find the best lexicon and segmentation for the training corpus, defined thus:

$$\underset{lexicon}{\operatorname{argmax}} P(lexicon|corpus) = \underset{lexicon}{\operatorname{argmax}} P(corpus|lexicon)P(lexicon). \tag{3.1}$$

The probability of the corpus is defined as the product of the probabilities of the words in the corpus over each of their morph μ_j and corresponding category transition probabilities, as follows:

$$P(\mathit{corpus}|\mathit{lexicon}) = \prod_{j=1}^{W} P(C_j1|C_j0) \prod_{k=1}^{n_j} P(\mu_j k|C_j k) P(C_{j(k+1)}|C_j k), \tag{3.2}$$

where W denotes the word types in the corpus, n_j are the n morphs in word j, $C_j k$ is the morph category, and $P(\mu_j k|C_j k)$ is its emission probability. $C_0|C_1$ denotes the word start probability– the transition probability from word boundary to first morph; $P(C_{j(k+1)}|C_j k)$ are the remaining transition probabilities, which include the transition $P(C_{j(n_j+1)}|C_j n_j)$ from the last morph to the word boundary.

The probability of the morph lexicon M is given as the product of the morph probabilities for each morph (calculated from their 'usage' and 'form') times the number of their possible orderings:

$$P(\mathit{lexicon}) = |M|! \prod_{i=1}^{|M|} [P(\mathit{usage}(\mu_i)) P(\mathit{form}(\mu_i))]. \tag{3.3}$$

The 'usage' and 'form' refer to features of the morph distribution and composition, respectively.

The probability of the form of a morph is determined by whether the morph is a flat string or has substructure of two concatenated submorphs:

$$P(\mathit{form}(\mu)) = \begin{cases} (1 - P(\mathit{sub})) \prod_{j=1}^{\mathit{length}(\mu)} P(c_j), \\ P(\mathit{sub}) P(C_1|\mathit{sub}) P(\mu_1|C_1) P(C_2|C_1) P(\mu_2|C_2), \end{cases}$$

where $P(sub)$ is the probability of a morph having substructure, estimated from the proportion of morphs with substructure in the lexicon. $P(c_{ij})$ is the probability of the letters in the flat string, determined by the distribution of the letters of the alphabet in the corpus. The second line of this equation is composed of the probabilities of the two submorphs $P(\mu_i)$ conditioned on their respective categories, as well as the transition probabilities between the two submorph categories.

The usage of a morph is given by:

$$P(\mathit{usage}(\mu)) = \mathit{frequency}(\mu)(\mathit{length}(\mu))(\mathit{right\_ppl}(\mu))(\mathit{left\_ppl}(\mu)). \tag{3.4}$$

The perplexities give an indication of the morph's context, measuring how much variation there is on what can occur on either side. The meaning features of a morph determine the morph's emission probabilities, or what category it gets assigned. A morph is assigned the category 'prefix' if its right perplexity is above a certain threshold, indicating that it can occur before a wide variety of other morphs. Analogously, the variety of different morphs seen on the left side of a morph (its left perplexity) determines whether it will be categorized as a suffix. A morph's right perplexity is given by:

$$\mathit{right\_ppl}(\mu) = \prod_{v_j \in right\_of(\mu)} P(v_j|\mu)^{-\frac{1}{f_\mu}}, \tag{3.5}$$

where v_j are the morph tokens occurring to the right of μ; the left perplexity is defined analogously.

These perplexities are used to assign morphs to categories using a sigmoid function with the threshold parameter b:

$$\mathit{prefix\_like}(\mu) = (1 + \exp[-a(\mathit{right\_ppl}(\mu) - b)])^{-1}, \tag{3.6}$$

with the parameter a determining the steepness of the sigmoid. The setting of the perplexity threshold parameter controls the granularity of segmentation. In general, the higher the perplexity threshold, the more coarsely the algorithm will segment the data, as more different contexts are allowed to be observed with each morph in the lexicon.

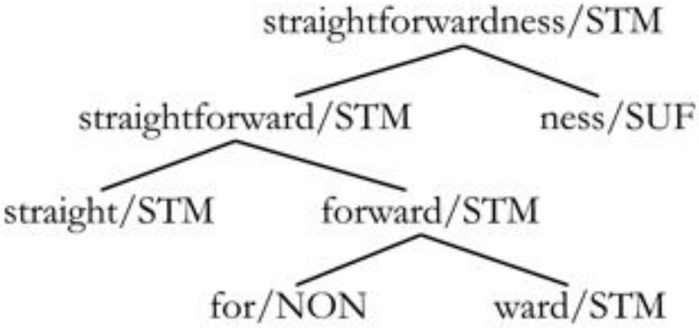


Figure 3.2: Hierarchical Segmentation Example (Creutz and Lagus, 2005)

Figure 3.2 demonstrates the hierarchical segmentation of a frequently seen word. In this example, 'straightforwardness' is frequently occurring enough in the training corpus to retain a separate entry as an independent unit, and in this context is analyzed thus. However, the model retains the substructure analysis, the constituent morphs of which remain in the lexicon for future reference.

The categories-MAP algorithm learns both the structure and the lexicon of the segmentation model. It does this iteratively, by first using the lexicon (initially provided by the run of the MDL-algorithm) to define the HMM states and compute its parameters given the lexicon. It next uses these model states and parameters to get a new lexicon, repeating this alternation until convergence.
In more specific terms, the algorithm works as follows:

1. initialize the model using the MDL segmentation
2. tag the initial morph hypotheses with category tags
3. split the morphs
4. join the morphs from the bottom-up
5. split the morphs
6. run the HMM Viterbi algorithm to resegment the corpus and use the Forward-Backward algorithm to reestimate the probabilities until they converge
7. repeat steps 4-6
8. do expansion of the morph substructures until no further expansion is possible without non-morphs.

Step three explores every possible binary split of the each morph in the current morph set into submorphs. The most probable split is chosen taking into account the changes to the context that the splits induce, with their corresponding transition probabilities. Step four joins morphs based on bigram frequency. At this step, the morphs are either concatenated together into a single morph and a new higher level morph is created consisting of the concatenated morphs as its substructure, or else the morphs are left separate.

When tested against a human-annotated gold standard of linguistic morpheme segmentations for Finnish, this algorithm outperforms competing unsupervised methods, achieving an F-score of $\sim 70\%$ on a 16 million word corpus consisting of 1.4 million different word forms.

3.5.2 Unsupervised Segmentation - Experimental Setup

To create the unsupervised segmentation models, we considered factors of granularity, coverage, and source-target symmetry. With respect to source-target symmetry, some (Matusov, Zens, and Ney, 2004) have hypothesized that equalizing the ratio of words per sentence in the source and target data might yield the best MT output. Using this hypothesis as a guide, we explored the parameters of the Morfessor segmentation for achieving the best MT performance, running several different settings to build multiple segmentation models. We altered the perplexity threshold and the training set which can generally be said to affect the segmentation granularity and data coverage, respectively.

With respect to the training data used, we examined whether it might be most beneficial to the quality of the segmentation for MT to use all of the training data, or to use instead a subset of those items occurring in the training data above some frequency threshold. The intuition behind this was that more frequent words may be more indicative of productive syllabic structure and morphophonemic rules governing inflection behavior.

In exploring parameters of the Morfessor segmentation model, we found that altering the perplexity threshold parameter to increase segmentation tended to make the per-word segmentation finer grained at a rate that outpaced the increased coverage of the data. For example, consider the following sentence's segmentation by two models trained on the same data set, the first with a perplexity threshold of 100, the second with a threshold of 10:

(7) polttoaineen hintakysymyskin vaikuttaa minusta erityisen tärkeä+ +ltä viime ai+ +kojen tapahtumien valossa.

(8) pol+ +t+ +toaine+ +en hintakysymyskin vaikuttaa mi+ +n+ +u+ +sta erity+ +isen tärkeäl+ +t+ +ä viime aikojen tapah+ +tumi+ +en valo+ +ssa.

Therefore, in addition to using the basic segmentation output to train the MT system, we did some additional data pre-processing to increase segmentation coverage. We approached this by beginning with an undersegmenting model and then preparing a set of the word-final suffix morphs gathered from the training data's initial segmentation. For this undersegmenting model, we used that trained on just the 5,000 most frequent words in the corpus rather than the model trained on the full corpus, since the latter model did not undersegment the corpus even with a high perplexity threshold. We then performed a simple substring match for any unsegmented words ending in the suffix characters. We took the longest match available for each word as the basis of splitting it. The intuition behind this approach was that the model was biased towards the least marked forms, which it tended to overgenerate in the test data. We assume that this is due to the translation model's word penalty, which keeps the decoder from overgenerating output by penalizing longer translations. The noun 'rakenne' ('structure') is an example of this– this barest form of the word, identical to the lemma, expresses the nominative case and is unlikely to be segmented; any other case requires the addition of an explicit suffix morpheme more easily identified in multiple contexts for segmentation.

Therefore, the goal of the longest-suffix-match was to provide the translation model with more examples of the bare stem as well as more examples of the more sparsely distributed suffixes. Figure 3.3 shows morph frequency plotted against its length in characters; we see that all the most frequently occurring morphs in the text are the shortest. As a corollary to the above approach, we also looked at taking the most frequent suffix match, as well as

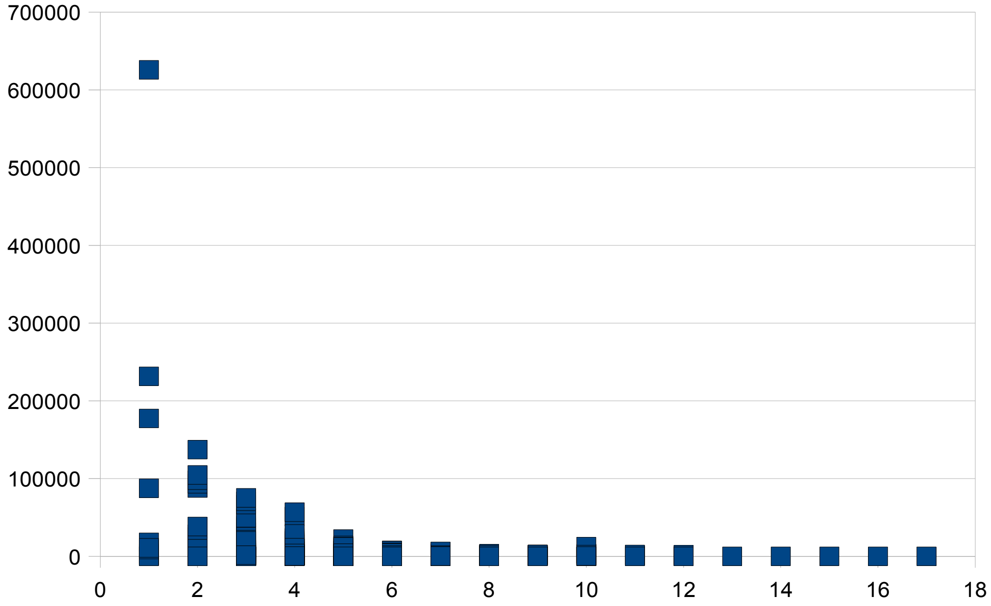


Figure 3.3: Morph Length vs. Frequency

the suffix match with the largest number of cooccurring stems in the training data. The idea of this approach was to provide the translation model with more examples of the most productively combinable suffixes.

These additional preprocessing techniques can result in an oversegmented model, with the number of target words/morphs in the parallel corpus significantly exceeding the number in the source, so we employed two heuristics to limit the suffix-match splitting to degree that would result in a more even ratio. For the longest-suffix-match, we imposed a suffix length limit, requiring it to be greater than two characters to be a match candidate, the idea being to reinforce the translation model's exposure to morphs other than the minimally marked forms. For the most-frequent-suffix match, the limiting heuristic we used was to impose a minimum threshold on the number of stems with which the suffix had been seen in the training data, again reinforcing a bias in the MT model towards the most productively combinable forms. For both these heuristics, we also required that the word be at least two characters longer than the candidate suffix for splitting, as Finnish syllabic structure generally required that stems be at least 2 characters (Sulkala and Karjalainen, 2008), so as to have the segmentation result in a more general but identifiable stem-like form.

3.6 Experiments and Analysis

3.6.1 Experimental Corpora Statistics

Table 3.1 shows the ratios for the various training corpora against the English course text, filtered for sentence length of 40 or less from the original 1,203,026 sentence data set

Using the source-target word ratio as a guideline for the substring splitting models, the length and frequency thresholds of 3 characters and 2,000 co-occurring stems respectively resulted in the closest parity between the English-Finnish data set token counts.

3.6.2 Results and Analysis

Using the same training, development, and test sets as the previous experiments, we trained MT systems on data segmented by the various unsupervised segmentation models, using Moses with identical settings as before. As shown in Table 3.2, the unsupervised model outperforms the supervised model. While BLEU differs from WER and TER in the best-scoring model, the best unsupervised segmented translation models consistently outperform

Segmentation	Train	Dev	Eval
Whole Words	0.77	0.71	0.71
Sup	0.99	0.99	1.00
Un- 5k PPL 10	1.27	1.27	1.29
Un- 5k PPL 30	0.86	0.96	0.96
Un- 5k PPL 100	0.86	0.85	0.85
Un- full PPL 10	0.87	0.94	0.95
Un- full PPL 30	0.96	0.95	0.97
Un- full PPL 100	1.01	1.02	1.02
Un- 5k PPL 100 L-match	1.11	1.12	1.13
Un- 5k PPL 100 F-match	1.00	0.95	0.96

Table 3.1: Corpus Ratios Against English Parallel Text. Sup refers to the model trained on the Omorfi supervised segmentation, while Un- denotes the Morfessor unsupervised segmentation. 5k means the model was trained on just the top 5,000 most frequently occurring words in the training data set, while full denotes the full corpus. PPL indicates the perplexity threshold used. L-match denotes the segmented corpus with additional splitting based on the longest-suffix-match heuristic, while F-match denotes that split with the most-frequently-seen-suffix-match.

Segmentation	Words	p-val	Segs	No Uni	WER	TER
Baseline	14.39±0.74		14.84±0.69	9.89	74.96	72.42
Sup	14.58±0.77	0.119	18.41±0.69	13.49	74.56	71.84
Un- 5k PPL 10	13.27±0.77	0.010	14.06±0.47	10.23	72.57	70.04
Un- 5k PPL 30	13.35±0.67	0.010	12.93±0.57	8.88	78.84	76.25
Un- 5k PPL 100	14.94±0.74	0.004	16.07±0.86	10.46	74.53	71.85
Un- full PPL 10	13.08±0.69	0.010	13.28±0.57	9.31	82.93	80.27
Un- full PPL 30	14.79±0.74	0.021	20.39±0.32	15.39	**71.99**	**69.35**
Un- full PPL 100	14.34±0.73	0.308	20.86±0.70	15.85	73.13	70.50
Un- 5k PPL 100 L-match	**15.15±0.78**	0.001	20.74±0.68	15.89	74.46	71.78
Unsup 5k PPL 100 F-match	13.87±0.78	0.002	19.40±0.72	14.19	73.62	71.13

Table 3.2: Segmented Model Scores. Words means that the output was evaluated in whole-word form, while Segs indicates that the segmented output was evaluated before morph re-stitching against a segmented version of the baseline. Uni indicates the segmented BLEU score without consideration of unigrams.

the baseline in all measures. This suggests that for the MT task, more important than the accuracy of the segmentation, or even the coverage, are the source-target symmetry and the consistency of the segmentation, as the unsupervised model returns consistent frequently seen substrings as morphs, rather than morphs based on posited underlying lemmas as in the supervised model.

Emphasizing the importance of token per sentence symmetry between the parallel texts are the results that show that among these more successful unsupervised methods of segmentation, oversegmenting, (by both coverage and granularity) leads to severe overfitting in the MT model. We found that greater segmentation coverage can be achieved through maximizing the size of the data used to train the segmentation model, and that with greater segmentation coverage, we were able equalize the token counts of the parallel corpora. However, we found that this does not necessarily benefit the MT system's performance; a segmentation trained on only the most frequent 5000 words in the training data performed roughly as well as (in fact slightly better than) the model trained on the full data set.

Among the unsupervised segmentation trained models, we were interested to see to what degree the MT system was actually using the morphological information. When looking at the phrase table for the best performing segmentation system without additional pre-processing, nearly half of the phrases available in the phrase table included segmentations. Of these, we are primarily interested in those that were bounded by hanging morph boundaries, as these are the phrases that allow morphemes to be productively combined, rather than simply memorizing the morphs that constitute a word and in effect treating the word as an atomic unit one-gram phrase that ignores morphological sub-word information. We found that one third of the phrases that included segmentations were of this productive morph-bounded type. For the best performing model that had been additionally split according to the longest-suffix match, while the overall phrase table was smaller, far more of these contained or were bounded by hanging morphs, as we would expect for a model trained on a more segmented corpus. These results are shown in Table 3.3.

However, examining the actual use of phrases in the the development and test data, we see that this proportion is far less, as Figure 3.4 shows.

For the regular segmentation model, roughly a quarter of the phrases that generated the translated output included segmentations, but of these, only a small fraction (6%) were morph-bounded phrases. While the additionally split segmentation model used more segmented phrases (as we might expect), a proportionally smaller number of these (3%)

	RegSeg	SplitSeg
Total	64,106,047	38,989,187
Morph	30,837,615	35,869,502
Hanging Morph	10,906,406	12,746,418

Table 3.3: Morphs in Phrase Table. RegSeg denotes the unsupervised segmentation trained on the 5,000 most frequent words in the corpus, with a perplexity threshold of 100. SplitSeg denotes this same model with additional longest-match suffix splitting. Morph refers to those phrases in the phrase table that included a segmented form, while Hanging Morph refers to those phrases that were bounded by a productive word-internal morph.

	RegSeg Test	SplitSeg Test
Total	21,938	22,816
Morph	5,191	17,611
Hanging	296	533

Table 3.4: Phrases Used in Translation

were bounded by hanging morphs.

It is interesting to note that even among the unsupervised models, while segmentations that increase the corpus total token symmetry between source and target do generally perform better than models with a wide divergence, the BLEU score results suggest that this is not the only aspect of morphological segmentation that benefits the translation model. As shown in Table 3.2, while the full-data-trained unsupervised segmentation with perplexity 30 had the strictly closest word ratios to the parallel text, the best performing model was the one that in used an additional splitting heuristic, resulting in a slightly more lop-sided source-to-target ratio.

We conclude from this result that additional preprocessing to the segmentation output improved the MT performance. We hypothesize that this may be due to the fact that the Morfessor model retains the substructure of words that contain substrings in its morphological lexicon, but if the word as a whole is frequent enough it warrants its own entry in the lexicon and may be chosen as an output, rather than its constituent morphs. While the segmentation model is aware of this substructure, the translation model is not, treating all whole words as atomic strings. Therefore, the substructure is lost; the additional suffix-match preprocessing seems in effect to carry this substructural awareness forward into the translation model. This positive effect is seen in the longest-suffix-substring match approach, but not the most frequent-suffix-substring match. We assume that this is due to how well the most frequent suffixes are already represented in the corpora; additional preprocessing segmentation of these forms seems to make them overly dominant over less frequently seen competing forms.

To get a better understanding of this result, we used the hand-written morphological analyzer to perform linguistic analysis of the segmented MT model output. We found that the model generally preferred to output the least-marked form, to a greater degree than that seen in the training data or references. In the case of nominals, this was the nominative case, which is the unmarked form, in contrast to the other cases which are marked by the addition of an explicit suffix. The following example translations demonstrate this. The first is the reference translation; the second translation was produced by the regular segmentation model, and the third by the additionally split segmentation. We have included back-translations generated by Google Translate[1] to help point out the translation differences, though these are not of as high a quality as a translation by a human.

(9) a. Input: ‘the csu ’ s europe group welcomes the tabling of the final draft of the charter of fundamental rights because it summarises and makes visible **the fundamental rights which the public are entitled to** in respect of the institutions and bodies of the eu’

b. Reference: (de) csu : n euroopan/GEN,ACC parlamentin/GEN,ACC ryhmä/-NOM suhtautuu myönteisesti siihen , että käsiteltävänä/VERB/ESS olevassa/-INE ehdotuksessa/INE perusoikeuskirjaksi/TRA kootaan yhteen ja tehdään näkyviksi/TRA kansalaisten/GEN **perusoikeudet/NOM,ACC** eu : n toimielimiin/ILL ja laitoksiin/ILL nähden

(10) a. Regular Segmentation: ‘csu : n ryhmä/NOM suhtautuu myönteisesti siihen , että euroopan/GEN,ACC unioni/NOM esittää lopullisen/GEN,ACC luonnoksen/GEN,ACC perusoikeuskirjasta/ELA , koska se/NOM ilmentää/VERB/LAT perusoikeuksia/PAR ja näkyvyyttä/PAR , johon/ILL kansalaisilla/ADE on **oikeus/NOM** toimielinten/GEN ja eu : n toimielimissä/INE’

[1] *http://translate.google.com*

b. Back-translation: ‘CSU group welcomes the fact that the European Union will present its final draft of the Charter, because it reflects **the fundamental rights and visibility, which the citizens have the right** institutions and the eu institutions’

(11) a. Longest-Match: ‘csu : n puhemiehistöltä/NOM ryhmä/NOM suhtautuu myönteisesti euroopan/GEN,ACC lopullisessa/INE luonnoksessa/INE perusoikeuskirjasta/ELA , koska siinä yhdistyvät ja tekee näkyvä perusoikeudet/NOM,ACC, jotka kansalaiset/NOM,ACC ovat **oikeutettuja/PAR** saamaan toimielinten/-GEN ja eu : n toimielimissä/INE’

b. Back-translation: ‘CSU’s Bureau of the Group welcomes the european final draft of the Charter, because it combines and makes visual **basic rights that citizens are entitled to** the institutions and the eu institutions.’

By translating ‘oikeus/NOM-right’ in the nominative, the regular model severs the scope of ‘right’ belonging to the ‘kansalaiset-citizens/public’. The additionally splitting model, on the other hand ties the ‘oikeutettuja/PAR-right’ to the ‘kansalaiset-citizens/public’ by inflecting it with the partitive case.

Nominals make up over a third of all tokens in the Finnish training corpus (8,217,727 out of the total 23,641,401). Of these, roughly 25% are inflected with the unmarked nominative case in the training corpus. In the reference translation set, the proportion of unmarked nouns is somewhat higher at 30.98%. Table 3.5 shows a comparison between the number of nouns inflected with the nominative case between the different models. We see that the regular segmentation model trained on the 5,000 most frequent words has a higher preference for the nominative, where the longest-match version of this model comes closer to the correct proportion in the reference text.

3.7 Chapter Summary

In this chapter, we have explored how various forms of morphological segmentation may affect machine translation quality. We find that segmentation of the morphologically complex

Translation	Total Nouns	Total in Nominative Case	Percentage
Reference	11,438	3,544	30.98
Seg5k	11,286	3,751	33.78
L-match	11,607	3,812	32.32

Table 3.5: Nominative Case Outputs

target language improves model performance over an unsegmented baseline.

Chapter 4

Morphology generation

In this chapter, we describe techniques for morphology generation, focusing on a Conditional Random Field (CRF) model for morphology generation. We begin with an introduction to this approach and related work, then give a detailed description of the model family. We then contrast this approach with those we have seen thus far in Chapter 3 and 2 to provide our rationale for using the CRF-based post-processing morphology generation approach. Next, we provide an account of our experiments with this approach along with results and analysis.

4.1 Morphology generation

In this approach, all (Minkov, Toutanova, and Suzuki, 2007) or certain classes of words(de Gispert and Mariño, 2008) in the morphologically complex language are stemmed before translation. As in the segmented translation approach, the stemming procedure strips off morphological inflections from the word stem, using the types of segmentations exemplified above. This can be done by unsupervised or by dictionary-based means, depending on the nature of the particular approach and the availability of such tools.

This type of approach allows major vocabulary reduction in the translation model, which can be quite helpful, and allows the use of morphologically targeted features for the specific purpose of modeling inflection. The possible disadvantage that this approach presents is that unlike the segmented translation or factored models, in the post-processing morphology generation model there is no opportunity to consider the morphology in translation since it is remove prior to training the translation model. For example, (de Gispert and Mariño, 2008) strips morphology verbs before training the translation model (e.g., ‘díganos’, composed of the stem ‘diga’ and the pronominal enclitic ‘nos’ is reduced to a stem plus POS tag ‘diga:VERB’).

After stemmed translation, morphology generation is done in post-processing on the MT output in order to reconstruct surface forms for the text. These models can use a variety of bilingual and contextual information to capture dependencies between morphemes, often more long-distance than what is possible using traditional ngram language models.

To make an inflection prediction for a stem in its sentence context, morphology generation models can combine features of the current stem including the previous and following bilingual context (target language stems/words and source aligned words), lexical (e.g., part-of-speech) and syntactic (e.g., head word) features of bilingual context, as well as conditioning on the n previous inflection predictions.

These kinds of features can then be combined to train a classifier (de Gispert and Mariño, 2008) to predict the most appropriate morph ending. In the following sections, we discuss some of the most prevalent models used to tackle this type of NLP task, and establish our motivation for using a CRF to model inflection prediction.

4.1.1 Generative Models - HMM

Hidden markov models are commonly used in NLP for tasks such as speech recognition and POS tagging. They are well-suited to model tasks which can be seen as a set of underlying conditions probabilistically generating observable events, and have the advantage of being efficiently trained. HMMs are composed of a set of (hidden) states, with a set of transitions defined between them; each state can emit tokens or labels. For a probabilistic HMM, each state transition is associated with a transition probability. Given an observation sequence, an HMM is used for to solve either pattern recognition problems or learning problems (Rabiner, 1989). The two kinds of pattern recognition problems are: given an HMM, find the probability of an observation sequence (evaluation), or, given an observation sequence, find the sequence of hidden states most likely to have generated it (decoding). The learning problem HMMs are used to solve amounts to: given an observation sequence, find the HMM that generated it. Morfessor, the unsupervised morphological segmentation model used in this thesis is an example of an HMM used for decoding, wherein given a surface word, the model finds the set of constituent morphemes that give rise to it.

HMMs are defined by two probability distributions. For the set of states S and observations O, we have the state transition probability $P(s|s')$ and the observation probability $P(o|s)$ which are used to model the joint probability of states and observations $P(o, s)$.

HMMs entail certain assumptions: First, HMMs condition the likelihood of the current state on the previous state only (first-order HMMs); second, transition probabilities are assumed not to change over time; third, the probability distributions of each of the observations are mutually independent (each must sum to one). Figure 4.1 illustrates the HMM and how it models each state S_i as relying on the previous state and its corresponding observation O_i only.

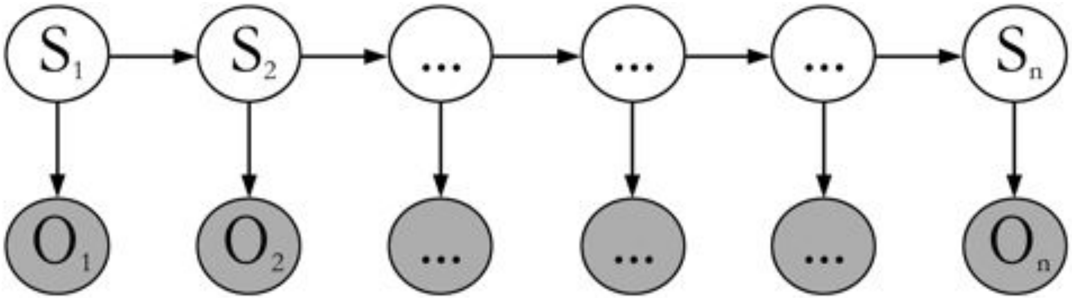


Figure 4.1: HMM

HMMs' independence assumptions limit their expressiveness; for the purposes of language modeling, we look instead to a model that is able to take into account global or overlapping features. Rather than conditioning the current state on the current observation only, it is helpful to consider the entire sequence of observations. For example, when predicting the proper translation of a word, we may wish to take into account not just the source word or the previous translated word, but also the length of the sentence, or the syntactic category of the headword to which the source word is bound. In addition, generative models spend part of their computational power modeling the observation sequence as well as the labels, but for the decoding or morphology prediction task, we only need to model the labels.

4.1.2 Discriminative Models - MEMM and CRF

Rather than modeling the joint probability of a state and observation, discriminative models define a conditional distribution over the states given the observations. Discriminative models do not waste modeling effort on the observations and can make higher order models computationally feasible (they can condition the next state; also, they can use features from any part of the input). Maximum entropy markov models and conditional random fields are defined by the following conditional distribution over states, given the corresponding observation and the previous state: $P(s|s', o)$ Moreover, by conditioning on the entire observation sequence, they allow the introduction of arbitrary overlapping features into the model.

The feature functions take the following form, where k is a (binary) feature of the observation and s is a destination state (McCallum, Freitag, and Pereira, 2000):

$$f_k = \begin{cases} 1 & k(s) \text{ is true and } s' = s, \\ 0 & \text{otherwise.} \end{cases}$$

The following sections describe how these features functions are used to make predictions such as morphology generation in discriminative models.

Maximum entropy models are conditional models that predict the next state given the current state (per-state normalization), represented in Figure 4.2.

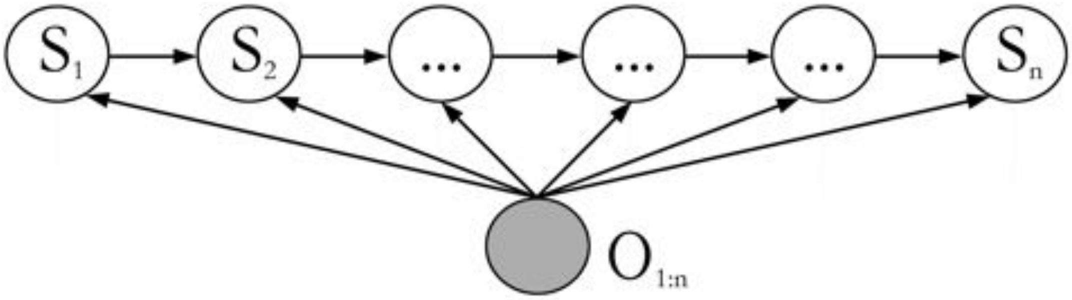


Figure 4.2: MEMM

Rather than creating a model that simply emits the proper labels, a maximum entropy model can be trained to generate output that displays the same distribution of these features as that seen in the training data. This is useful when we have features of the data that may tell us more than the training data output tokens alone. By maximizing the entropy of the model, it is trained to be maximally general while subject to the feature constraints. These constraints are expected values of the features in the distributions learned from their counts in the training data. Over a state and observation sequence $1..t$, for a feature k and each state s, the feature's expected value is equal to its average value in the training data, and takes the form:

$$E_k = \frac{1}{t_{s'}} \sum_{n=1}^{t_{s'}} \sum_{s} P(s|s', o_n) f_k(o_n, s) \tag{4.1}$$

$$= \frac{1}{t_{s'}} \sum_{n=1}^{t_{s'}} f_k(o_n, s_n) \tag{4.2}$$

where $P(s|s', o)$ is the transition function from state s' to state s.

Maximum Entropy Markov Models (MEMM) decompose the probability of a state sequence into the local probabilities for each state. Then for each state, the distribution takes the usual log-linear form:

$$P_{s'}(s|o) = \frac{1}{Z(o, s')} \exp(\sum_k \lambda_k f_k(o, s)), \tag{4.3}$$

where λ_k is the weight parameter for feature f_k, and $Z(o, s')$ is the normalizing factor for each state. To model an entire sequence of states and observations, MEMMs combine these feature functions over the states and observations as the product of the normalized probabilities thus:

$$P(S = s_1, ..., s_n|O, \lambda) = \prod_n P_{s'}(s|o). \tag{4.4}$$

Then the probability of the best state sequence is given by:

$$s_1*, ..., s_n* = \underset{S=s_1,...,s_n}{\operatorname{argmax}} \prod_n P_{s'}(s|o). \tag{4.5}$$

4.1.3 Maximum entropy models for morphology generation

In (Minkov, Toutanova, and Suzuki, 2007), MEMMS with overlapping features are used to successfully predict inflection on Russian and Arabic stems. Each word in the training data is segmented into stem plus morphologically analyzed inflection. Then for each stem in the training data, an inflection set is gathered consisting of every inflection seen in the data for that stem. From this set, the model makes a prediction for each stem in the test data according to the context in which it appears. Crucially, this model allows the use of overlapping features of the observations and the predictions, so as to be able to capture overlapping dependencies between morphemes like agreement phenomena.

$$f_k = \begin{cases} 1 & \text{ACC and CASE}(s_{n-1}) = \text{ACC}, \\ 0 & \text{otherwise.} \end{cases}$$

The above example feature could be used by a MEMM to capture the case agreement between an adjective and its noun. These kinds of dependencies are difficult for a standard n-gram language model, which can only take into account fixed statistics of sequences of context words.

An interesting effect of this modeling approach is known as label bias (John Lafferty and Pereira, 2001). Since the MEMM is a per-state exponential model, it is normalized locally for each state. Roughly, this means that transitions leaving a given state compete only with each other for probability mass, rather than against the other transitions in the model, in a way that may not reflect the actual dependencies in the data (dependencies typical of linguistic data). Its per-state normalization requires that all the probability mass that comes in to a state must be passed along to its successor states: the observations can effect which successor states get the probability mass, but not the total amount of mass that gets passed on. The effect is that states with fewer outgoing transitions are preferred, and that the observations can be to a certain extent ignored.

To give an example of how the label bias problem works, consider a finite state model to disambiguate between the sentences ‘I'm going to the park’ and ‘I'm going to fall asleep’ (Figure 4.3). Imagine that both sentences occur with roughly equal frequency, but that in this case the actual observation sequence is ‘I'm going to fall asleep’. Since the observation ‘to’ matches both transitions from state 1, the probability mass gets distributed evenly between states 2 and 5. We next come to the observation ‘fall,’ which has never been seen in the training data from state 5. However, this model conditions the successor state on the observation, but does not generate it, so since there is only one outgoing transition from state 5, it moves all of its probability mass forward to state 6. Then, if P(DET|PP, to) is greater than P(VB|INF, to) then the model recognizes the sentence ‘I'm going to the park’,

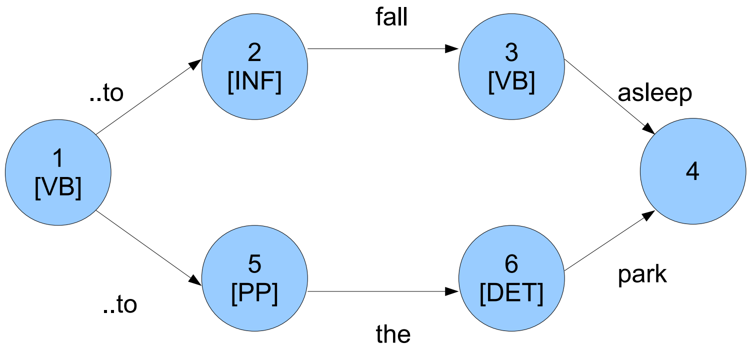


Figure 4.3: A Label Bias Example

and effectively ignores the observation.

In (Toutanova et al., 2003), the authors point out a phenomenon similar to label bias, which could analogously be termed ‘observation bias.’ In doing so, they point out that directed models can suffer from the effect of severing the influence of how either a label or an observation affects another because they artificially model nodes in a sequence as being in causal competition.

Conditional random fields, on the other hand, are partially directed models which avoid the label bias problem by using global normalization, utilizing the same arbitrary overlapping features as the MEMM, but allowing these models to take into account the entire observation sequence O with each prediction without restricting each state S_i to depend on the previous state only. Figure 4.4 illustrates this.

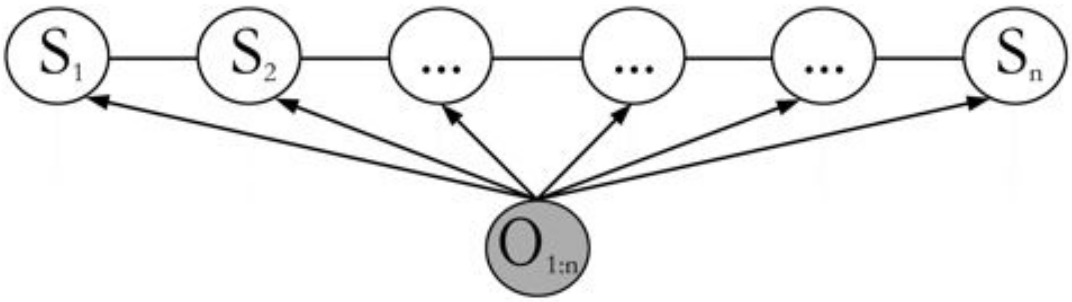


Figure 4.4: CRF

Following the example above, we can look at the CRF as a finite state model with unnormalized transition probabilities, which instead normalizes globally over all state sequences for a given input word sequence. Alternatively, we can contrast the CRF to the MEMM per-state exponential model, where the CRF is a single exponential model of the joint probability of the whole state/label sequence given the observation sequence. In the CRF model, the probability of a given inflected sequence S given an observation sequence O is given by:

$$p(S|O) = \frac{1}{Z(O)} \exp(\sum_n \sum_k \lambda_k f_k(s, s', O)), \tag{4.6}$$

where λ_k is the weight parameter for feature f_k, and where the normalizer $Z(O)$ sums over all label sequences for O:

$$Z(O) = \sum_{s} \exp(\sum_{n} \sum_{k} \lambda_k f_k(s, s', O). \tag{4.7}$$

Formulated in log-linear fashion, the best label sequence is then given by:

$$s_1*, ..., s_n* = \underset{S=s_1,...,s_n}{\operatorname{argmax}} \log P(S|O, \lambda), \tag{4.8}$$

where

$$\log P(S = s_1, ..., s_n|O, \lambda) = \sum_{n} \sum_{k} \lambda_k * f_k(s, s', O) - \log Z(O, \lambda). \tag{4.9}$$

For applications in NLP, (John Lafferty and Pereira, 2001) show that CRFs can obtain better results in a POS tagging task when compared to both HMMs and MEMMs.

To use CRFs to model morphology generation, we model the morphology as the states, and the stems as the observations. Figure 4.5 gives an example of how this works for a stem plus POS-tag observation sequence, ‘tulevaisuus NOUN’ ‘enemmän ADVERB’ ‘perustuslaillinen ADJ’ ‘rakenne NOUN’ (‘future’ ‘more’ ‘constitutional’ ‘structure’). For each stem observation, a morphology prediction is made. Then the stem plus morphology prediction forms are used to arrive at the inflected surface form ‘tulevaisuudessa enemmän perustuslaillista rakennetta’ (‘more in the future of constitutional structure’).

4.2 Factored Translation, Segmented Translation, and CRF Post-Processing Morphology Prediction - Model Comparison

The translation models examined herein use the same segmented corpora, but they use the morphological information in the corpora very differently. To get a better intuition about how the models stack up against each other and their different strengths, we now include a brief comparison discussion.

4.2.1 Factored Model vs. CRF Post-Processing Model

A main distinction between the factored and post-processing models is how they produce morphology for stems. They are both based on the same basic phrase-based framework for

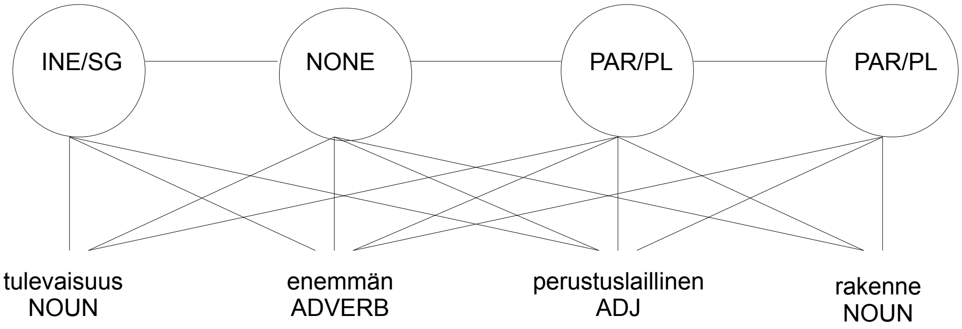


Figure 4.5: CRF for Morphology Prediction. In this example, the observations are the stems with POS-tags, and the states are the morphology tags.

translation, both using the following log-linear translation model:

$$p(t|s) = \frac{1}{Z} \exp \sum_{i=1}^{n} \lambda_i h_i(t,s), \tag{4.10}$$

where h_i are feature functions of the data. For both models, these can be word and phrase translation probabilities, language model probabilities, and phrase-length and reordering penalties.

However, in the factored model, the probabilities for a word are decomposed over each of its individual factors, i.e., lemma, POS, and morphology. This decomposition affects how the model selects the target translation word, including the stem. In the post-processing model, the morphology is outside the translation model and cannot effect how the target word or stem is chosen, only how it is inflected later. Both have their respective advantages; while some morphological information may be beneficial to the translation model, that morphology which has no source-side reflexes might be better captured by the target-side generation model, and only hinder the translation model.

To get a better idea of how these differences work let us return to the translation model feature functions mentioned earlier. As we saw in Chapter 2, for both the factored and basic phrase-based models, the translation feature functions for a sentence pair are decomposed into phrase translations covering the sentence pair, which are given a scoring function, given by:

$$h_\tau(t,s) = \sum_j \tau(\bar{s_j}, \bar{t_j}). \tag{4.11}$$

The factored model has the addition of separate feature functions corresponding to the multiple translation and generation steps. These feature functions can include factor-for-factor probabilities (instead of lexical translation probabilities, these can be POS-to-POS etc), as well as additional language models over the different factors. The generation feature functions are defined as scoring functions over the words (rather than phrases) summed over the sentence on the target side only:

$$h_\gamma(t,s) = \sum_k \gamma(\bar{t_k}). \tag{4.12}$$

These scoring functions capture, in the task at hand, the likelihood of a morph occurring with a stem or a stem and morph occurring with a surface word, based upon the stem-morph co-occurrences seen in the training data, as well as the language model over the stems or over the morphs. For example, generation functions may indicate that 'SG|people' is less likely than 'PL|people', or, in the case where the target side is annotated with POS as a factor, generation functions can capture that the sequence of POS tags 'NN|DT, ADJ', may be more probable than 'DT|DT, ADJ' (using the POS-factor language model). We see that in the factored model, the morphology generation for any given word depends upon the translation probability of the stem, which controls what morphs may be generated from it by means of the generation features for the stem, and any translation between source side factors with the target morphology for that word.

We saw in the example in Section 2.3 how factored translation models may come up with a correctly inflected word even in the case of unseen input. The success of this was predicated on the availability of a suitable lemma for which to generate morphology when forced to take the alternate decoding path in the case of an unseen word. The post-processing model has access to these stems as well, but not the surface forms or multiple decoding paths. This opens up the possibility that the CRF post-processing model is then impoverished by remaining ignorant of inflected forms, some of which do indeed have translations without the necessity of stemming. It also risks that the translation model may try to find lexical stem equivalents for items (like English prepositions, for example) which frequently have clear morphological reflexes in the target, thus adding in unnecessary words.

Translation model aside, the major differences between these two models for incorporating morphological information can be characterized as follows. Within their respective stages for morphological generation (either during or post-translation), the factored model generates suffixes on a per-word basis only, whereas the post-processing model takes into account long-distance dependencies between morphemes and between morphemes and stems. In the generation step, the prediction model has access to the same features that inform the factored model's morphology generation, with the addition of non-local features.

This brings us to the main drawback of factored models in comparison with the post-processing prediction model is the local stem-based morphology generation. As a simple example of how this might work in a phrase, consider translating the English phrase 'good|ADJ times|NNS/PL' into a morphologically complex target which enforces agreement between nouns and adjectives. The POS tags available to the factored model could help it to predict the plural morphology on the noun correctly in the target, but not so for the adjective, as these are not marked for number in English. Therefore the generation step must use the local stem probabilities; if the singular form of the adjective is more probable in the target text, that may incorrectly get picked. In addition, it may only choose from those morphs seen with that given stem in the training text, forbidding the production of novel inflected forms. As we saw in Section 4.1.2, the CRF model's prediction will instead be predicated on features over the entire input sequences as well as other predictions, allowing it to consider features such as those capturing agreement in addition to the local stem-morph co-occurrence probabilities.

4.2.2 Factored Model and Post-Processing Model vs. Segmented Model

Factored Model vs. Segmented Model

Between the factored and the segmented models, the major differences come down to how much access the model has to unsegmented word forms, how much morphological productivity is allowed, and how much morphology can be used in translation.

The factored model is aware that a word and its stem are related and has the option of translating on the word basis, whereas, the segmented model has no knowledge of the original surface form of a segmented word. On the other hand, the factored model's phrase table has no entries that combine stems and morphology, whereas the segmented model may do so.

Since the factored model decomposes the word probability into the probability of its factors, we can say that the model takes morphemes into some consideration in translation. However, it is important to note that this is only in the limited sense of being local to the word for which it is a factor. In addition, the factored model has no way of translating any content of the morphology, only guessing the grammatical class it corresponds to (unless we were translating between two morphologically rich languages, in which case we could use source-side morphology as a translation factor). This is due to the formalism of the factored model which allows morphological information to be represented only in a hierarchical relationship to the word, from which it cannot be translated separately on a lexical level without abandoning the word-aligned model.

CRF Post-Processing Model vs. Segmented Model

The important differences between the post-processing and the segmented models come down to whether morphology is used in translation and the amount and subtlety of information used to apply morphology to stems.

Like the segmented model, the post-processing model is able to combine stems with new morphs in ways not seen in the training data. However, like the factored model, the post-processing model lacks the segmented model's capacity to translate morphs, so is not able to make use of informative morphs with lexical correspondences.

In terms of how they select morphology the two models are fundamentally different; the segmented model makes no distinction between stems and morphology. In the post-processing model, morphology is generated for each on the basis of feature functions capturing various aspects of the stem and morphology context in the sentence.

4.3 Using CRFs for Morphology Generation

In this approach, morphology is generated for word stems in a post-processing step outside the translation model. Using this technique for MT presupposes that the translation model trained on stems or some form of morphologically simplified output, rather than on full words, performs better than a word-based model. Even a prediction mechanism that performs with perfect accuracy in post-processing is of little use to the MT task if the translated output on to which it is applied is not of high quality in the first place.

After segmentation pre-processing, the MT model is trained on the morphologically simplified training data. The output from the MT system is then used as input to the CRF morphology prediction model, which generates the missing inflection. In using the CRF model for the task at hand, the set of observation symbols are all the word stems generated by the segmentation model used (either supervised or unsupervised) in the training data; the corresponding labels are the morphology set from which a morph is chosen to inflect that stem in its given context. The prediction model is trained on some intermediate representation of the morphs so as to be able to capture the more general patterns of morphological distribution. From here, we use an additional language model post-processing step in order to recover the fully inflected surface forms. Figure 4.6 shows the full pipeline.

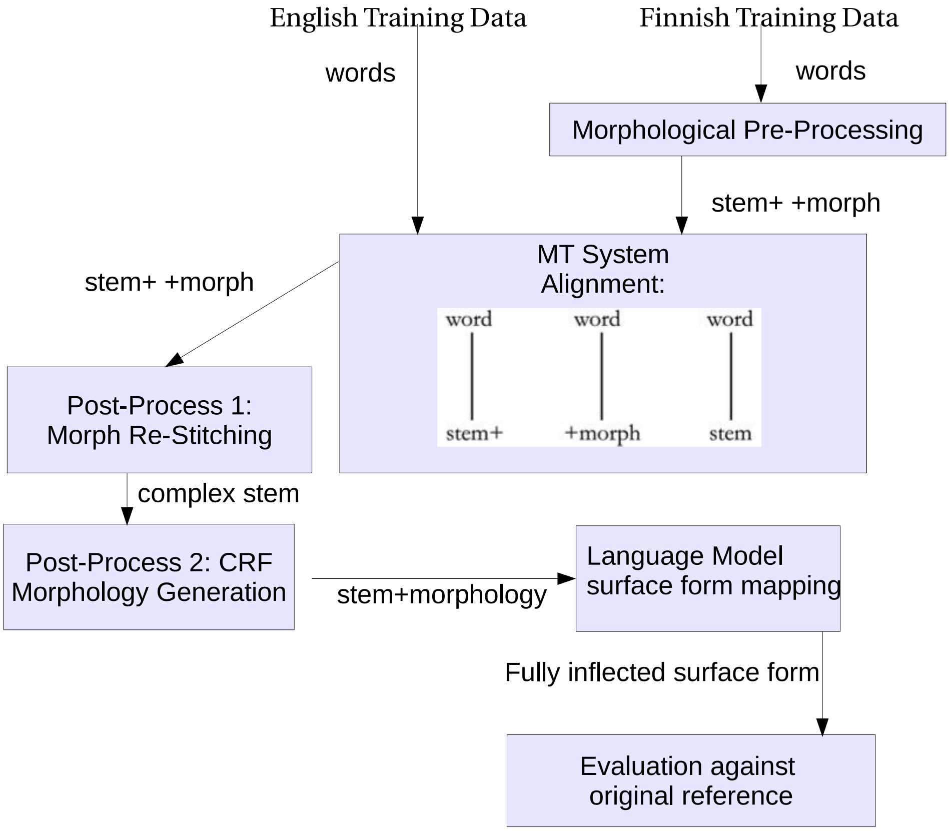


Figure 4.6: Post-Processing Model Translation-Generation Pipeline

4.4 CRF Morphology Prediction Using Supervised Segmentation

4.4.1 Experimental Setup

To derive the stem and morphological information, we ran the Omorfi FST morphological analyzer on the data, from which we extracted for each word its lemma, part-of-speech (POS) tag, and morphologically decomposed inflection sequence in the form of morphological category and value tags. When no analysis was generated, we retained the unstemmed word in the data with null feature values, that its surface form might still be used by the uninflected language model. We used the FST analyzer in hopes that it might be more effective at yielding productive morphological patterns in languages with fusional tendencies with a great degree of morphophonemic alternation that make straightforward segmentation difficult. We therefore used the concatenation of morphological tags available from the analyzer rather than a segmented parse.

For the lexical features, we considered up to trigrams of the previous and next lemmas and POS-tags. For the morphological features, we considered unigrams and bigrams of the current and previous states, giving the morphology predictions; the feature templates can be seen in Figure 4.1. In theory it is possible to use long-distance features for this model, however in the case of the supervised prediction experiments, this simply provides us with a convenient independence assumption. Rather than using long-distance features in this experiment, we restricted our features to a closer subset because the complexity of the multiple prediction feature values caused a combinatorial explosion preventing model training over more distant features. We represented each of the series of morphological tags making up the inflection prediction set as a vector of feature values corresponding to each morphological category, using the categories of number, case, and person. This represents a restricted selection of Finnish morphological categories returned by the analyzer, since it was not computationally feasible for the CRF implementation to include all the categories. These categories were selected on the basis of being the most prevalent and exhibiting the most morphologically dependent behavior. Table 4.1 shows the feature set for this model.

To recover the fully inflected word forms corresponding the the CRF output lemma plus inflection tag sequence, we used a word-based language model to predict which inflected word should correspond to an ambiguous lemma plus inflection tag sequence. The goal

Feature Categories	Instantiations
Lexical	
Word Stem	$s_{t-2}, s_{t-1}, s_t, s_{t+1}, s_{t+2}$
f: POS	$f(s_{t-2}), f(s_{t-1}), f(s_t), f(s_{t+1}), f(s_{t+2})$
Morphological	
f: number, case, person	$f(y_{t-2}), f(y_{t-1}), f(y_t)$

Table 4.1: Supervised Prediction Model Feature Set

of this approach was to capitalize on the large amounts of unannotated monolingual data to train the language model for disambiguation in surface form prediction. For this, we trained a 5-gram language model on the original 1.3 million sentence Europarl Finnish data set, unedited for sentence length or parallel corpus alignment.

The translation output of the MT system trained on supervised-segmentation-generated lemmas scored 22.28 BLEU against a lemmatized reference.

4.4.2 Experiments

The CRF model was trained on a 315,247 Finnish sentences, consisting of 6.28 million tokens; the 2000 sentence Europarl test set consisted of 41,434 stem tokens. From the training set, the CRF model extracted 628,366 features for 46 possible suffix outputs. In the test data, each word was represented as the Omorfi-generated word lemma along with its Omorfi-generated POS tag.

4.4.3 Results and Analysis

We first performed an intrinsic evaluation of the prediction model by testing it on the reference translation. In this evaluation, the CRF module achieved accuracy rates of 77.18 percent in predicting the morphology tag sequences. To give a frame of reference, these results are comparable to a similar task performed using log-linear models on Russian and Arabic, languages with simpler morphological systems than Finnish (Minkov, Toutanova, and Suzuki, 2007).

However, as shown in Table 4.2, after the application of the language model to disambiguate the corresponding surface form, the evaluation scores on the MT output were lower than the word-based baseline.

Model	BLEU Score	WER	TER
Baseline	14.39±0.74	74.96	72.42
CRF-Sup	10.09±0.67	78.22	76.29

Table 4.2: Supervised Prediction Model Scores

A limitation of this model is that it must predict the entire inflection set for each stem as an atomic unit, rather than individually predicting its composite components. In other morphologically complex languages with more fusional morphological systems, such as those used in the related work, this may not present as great an obstacle, as all the morphological information may be carried in a single irreducible affix. However, for a language like Finnish with agglutinative properties, the morphological information may be more useful if it is unpacked into discrete morphs. However, this type of prediction model is ill-suited to predict a sequence of morphs for each input.

4.5 CRF Morphology Prediction Using Unsupervised Segmented Translation

In this section, we combine the CRF morphology prediction approach with segmented translation. We do this in consideration of the inherent disadvantages in using supervised methods to obtain a segmentation and the good results achieved by the segmented translation method. The improvements in the segmentation model's performance over the baseline 3 indicate that the segmentation of the training data provides useful information to the translation model. In addition, for many languages such as Finnish that exhibit widespread vowel harmony and consonant gradation recovery of surface forms after morphology prediction is quite difficult, particularly in light of the necessity of using an intermediate morphology representation, rather than the surface morphs. Thus, we look to combining the morphology prediction model with the segmented translation MT model; the goal is to maximize the benefit of the segmentation to the translation model as well as the strengths of a CRF model for morphology prediction after translation.

4.5.1 Experimental Setup

As mentioned earlier, using the post-processing prediction model requires good MT output in the form of stem-like units on which to make inflection predictions. However, when using unsupervised segmentation-trained system output there is no linguistically motivated 'stem' available from the segmentation, making this requirement less straightforward to meet. We therefore experimented with different segmentations to see how the stemmed version of the text they yielded performed with the MT system.

For training an MT system for use with the prediction model, we tested the segmentation that had achieved the best BLEU scores without additional preprocessing, as well as an oversegmenting model and a segmentation model that underwent additional preprocessing for suffix splitting as described in Chapter 3. For this last model, we removed suffixes from words not segmented by the segmentation model according to the longest-match criterion that achieved the best translation results in Chapter 3. With each of these models, to derive the morphologically stripped forms for MT system training, we looked at all word-final suffixes available from the segmentation; these would be stripped from the training corpus and used as morphology prediction outputs.

We found that the granularity of the segmentation made a significant difference in the quality of the resulting stem-only model. For coarsely segmenting models, training MT models on the resulting stems yielded poor quality translations, even when evaluated against a reference text that had also been stripped of suffixes. We found that using the same more coarsely segmenting model that achieved the best performance in Ch.3 to provide a 'stem' for MT model training and 'suffix' for prediction later actually performed more poorly than the word-based baseline model. We expect that this is due to the fact that coarsely segmenting models are more likely to break a word into two large segments, wherein the suffix may be a conflation of a complex morpheme sequence, the removal of such making the training data less informative by way of removing too much information from the word, and the sentence as a whole. Removing only word-final suffixes from a finer-grained model, on the other hand, may both allow the translation model to retain a useful amount of morphological awareness, while still creating a smaller vocabulary of more general forms. For example, the best-performing segmentation model segments 'käytettävä' as käy+ +tettävä, while the oversegmenting model segments it as, käy+ +t+ +et+ +tä+ +vä, leaving more for the translation model after stripping the final suffix.

Segmentation	BLEU Score
Seg5k PPL 30	10.55±0.59
FullSeg PPL 100	14.79±0.76
FullSeg PPL 100 L-match	15.73±0.77

Table 4.3: Unsupervised Stemmed MT Model BLEU Scores Against a Stemmed Reference. Seg5k refers to the segmentation trained on the 5,000 most frequent words in the training data; fullSeg means that it was trained on the full data set. PPL denotes the perplexity threshold used.

Table 4.3 shows the comparative performance of MT models trained on various stems-only segmentations against a stemmed form of the reference. By *stems*-only we mean those models trained on corpora that have had some form of the morphological segmentation removed, such as omitting the final suffix segment from those words with available analyses from the segmentation model. As we might expect, some of these morphologically simplified models perform better than their more complex counterparts, but it is important to note that these systems are evaluated against a stemmed version of the reference. The value of these simplified models is for use with the post-processing prediction module; having obtained a high-scoring stemmed model, we can now use this output to apply our CRF morphology prediction model.

4.5.2 Experiments

In light of their performance over the 5,000-most-frequent-words segmentation model, we chose the FullSeg and FullSeg-longest match on which to test our suffix prediction model. Before suffix stripping, these oversegmenting models performed more poorly than the word-based baseline for MT training and had a ratio of fewer words per sentence in the source than in the target. For use with the CRF, we preprocessed the Morfessor segmentation output to remove the only final suffixes from words with available segmentations; thus in the case of words segmented into more than two pieces, the preceding segmented forms were left intact.

Due to the CRF implementation constraints, this approach requires limiting the number of possible prediction outputs. We therefore did not use the entire set of suffixes created by Morfessor for the training data, but instead limited this number by a threshold of the number of distinct stems the suffix is seen with in the training data. Akin to the threshold heuristic imposed in 3, this requirement seeks to use the prediction model to inflect morphologically stripped forms with the more highly marked of the most productively combinable suffixes. We were able to train the CRF with the suffix set resulting from setting the suffix productivity threshold at 150 stems, giving the CRF model a total of 44 possible label outputs. This suffix productivity threshold was calculated after collapsing certain vowels into equivalence classes corresponding to Finnish vowel harmony patterns. Thus variants -kö and -ko become vowel-generic enclitic particle -kO, and variants -ssä and -ssa become the vowel-generic inessive case marker -ssA, et cetera. These suffix transformations to their equivalence classes prevent morphophonemic variants of the same morpheme from competing against each other in the prediction model.

Using the same training, development and testing data as the supervised CRF experiments, the CRF was trained on monolingual features of the segmented text sentence context for suffix prediction, as shown in Table 4.4. With this more simplified feature set than that used with the supervised segmentation prediction model, we were able to use features over longer distances, resulting in a total of 1,137,228 model features.

Feature Categories	Instantiations
Word Stem	$s_{t-n}, .., s_t, .., s_{t+n}$
Morph Prediction	y_{t-2}, y_{t-1}, y_t

Table 4.4: Unsupervised Prediction Model Feature Set

For the unsupervised experiments, each word in the test data was represented as the bare word ‘stem’ alone. After prediction, we use a bigram language model trained on a full segmented version on the training data to recover the original vowels. We used bigrams only as the suffix vowel harmony alternation depends only upon the preceding phonemes in the word from which it was segmented.

4.5.3 Results

As with the previous morphology prediction experiments, we first show the accuracy of the CRF models alone, before moving on to extrinsic evaluation on MT output. These results test the CRF models on the reference translation after being stripped of the suffixes in the prediction set.

CRF-Unsup Model	All	Predictions Only
FullSeg	95.61	77.57
FullSeg L-match	82.99	74.73

Table 4.5: Model Accuracy

As Table 4.5 shows, the model trained on the regular segmentation split model fared far better than longest-match extra suffix splitting model. However, this is in part due to how much more frequently the latter model is called upon to positively predict suffixes.

CRF Model	Positive Predictions	Percent
FullSeg	6,961	16.80
FullSeg L-match	28,058	67.72

Table 4.6: Model Predictions

Table 4.6 shows how often the models make a null suffix prediction. Out of the total 41,434 tokens in the reference translation, the CRF trained on the regular segmentation model only made positive prediction quite infrequently compared to the model that had undergone preprocessing for additional suffix splitting.

To get an upper bound on how well this post-processing model can do, we considered performing an oracle experiment. Such an experiment would recognize all possible surface forms for each stem as correct matches in evaluation of the post-processing model output, in order to capture how well a perfectly predicting model would score. However, in accounting for all the surface form combinations, a brute force matching would require over 150 million different comparisons, making it too time consuming to perform within the scope of this thesis. Another measure, the stemmed MT output before morphology prediction evaluated against a stemmed reference, can give us an approximate estimate of the model's upper bound, but this measures slightly different behavior and is therefore not a precise upper bound for the post-processing model.

Table 4.7 shows the evaluation scores for the prediction models on MT system output compared to the word-based baseline. While the best performing prediction model showed only modest improvement in BLEU, WER, and TER scores over the baseline, a closer look

Model	BLEU Score	p-value	WER	TER
Baseline	14.39±0.74		74.96	72.42
CRF	14.55±0.76	0.150	73.71	71.15
CRF with L-match	14.33±0.77	0.120	73.91	71.40

Table 4.7: Unsupervised Prediction Model Scores

at the prediction system's output translation indicates that there may be translation fluency improvements not reflected by the evaluation measures.

4.5.4 Morphological Fluency Analysis

To get an idea of how well the post-processing prediction model was doing at getting morphology right in comparison the word-based model, we examined several aspects of agreement phenomena and other patterns of morphological behavior. While we wish to explore minimally supervised morphological MT models, and include in these models as little language specific information as possible, we do want to use linguistic analysis on the output of our system to see how our models fare at capturing essential morphological information in the target language (in this case, Finnish). So, to do this we took the baseline system output, the prediction model output, and the reference translation, and ran them all through the supervised morphological analyzer. For each word it can recognize in the translations, the Omorfi supervised morphological analyzer outputs the lemma, POS, and morphological categories inflecting the word. If the analyzer encounters a word that is morphologically ill-formed or is based on a lemma not in its lexicon, then it returns no analysis for that word.

Translation	Analyzed Tokens	Total Tokens	Percent
Reference	62,330	64,035	97.34
Baseline	60,958	62,563	97.44
Post-Processing	59,486	61,382	96.91

Table 4.8: Translation Tokens with Morphological Analyses

Figure 4.8 shows the analyzed token counts for the reference translation, the baseline output, and the post-processing system output. We can see that the post-processing model has the highest proportion of unanalyzed tokens. However, it is unknown whether this is because it is creating more illegal form or because it is using more words with stems outside of the analyzer's lexicon. It is worth noting, though, that despite generating inflected forms by productively combining stems and morphs, it still achieves close to the same proportion of analyzed words as the baseline model, which output only morphological forms already seen in the training data.

Using this supervised segmentation, we proceeded to look at a variety of linguistic constructions that might be reveal patterns in morphological behavior. The constructions we examined were: noun-adjective case agreement, subject-verb person/number agreement, transitive object case marking, prevalence of explicitly marked noun forms, interrogatives, postpositions, and possession. In each of these categories, we are interested in which translation model comes the closest to the reference value.

Construction	Frequency	Baseline	L-match	Post-Processing	Reference
Unmarked Nouns	5.5145	35.96	32.32	**32.02**	30.98
Trans Obj	1.0022	47.07	47.75	**48.69**	77.94
Noun-Adj Agr	0.6508	76.11	**80.11**	75.48	80.10
Subj-Verb Agr	0.4250	69.83	68.68	69.73	67.22
Postpositions	0.1138	70.00	83.27	73.45	81.37
Possession	0.0287	48.68	51.35	**54.17**	82.14

Table 4.9: Model Accuracy: Morphological Constructions. Frequency refers to the construction's average number of occurrences per sentence (correctly or otherwise), also averaged over the various translations. The constructions are listed in descending order of their frequency in the texts. The highlighted value in each column is the most accurate with respect to the reference value.

Figure 4.9 gives a summary of the comparative performance of the baseline and post-processing models on the various morphological constructions we examined. In all but two categories, the post-processing model outperforms the baseline, as we explain below. We also include results for the best of the segmented translation models for comparison. In general, the post-processing prediction model comes closest to the reference most frequently, but both this model and the segmented model do better than the baseline.

There was little evidence of a difference between the baseline and the post-processing models in two agreement categories we examined, starting with noun-adjective case agreement and subject-verb person/number agreement. In Finnish, adjectives must be marked with the same case as their head noun. We saw nearly the same proportion of case agreement between adjectives and nouns in the baseline output and the CRF prediction model, with the baseline's percentage slightly better. However, for this construction, the segmented model performed the closest to the reference.

In addition, Finnish is default Subject-Verb-Object word order and generally requires verbs to match the person and number of their subject. Again, we saw nearly the same proportion in the baseline and the prediction model of person/number agreement between a noun the verb that follows it, though both were higher than the reference ratio. However, looking at the noun immediately preceding a verb is a rough diagnostic for the syntactic subject construction; without a syntactic parse of the translations based on looking just at we do not know that the noun is in fact the syntactic subject of the verb. We conclude that this diagnostic for that particular construction is too rough to be an informative indicator.

However, the CRF model output diverges from the baseline in some other aspects of morphological behavior. Finnish generally marks direct objects of verbs with the accusative or the partitive case, and we observed a higher percentage of accusative/partitive-marked nouns following verbs in the CRF output, with 48.69%, versus 47.07% in the baseline. The translations below illustrate this phenomenon. While neither translation picks the correct verb for the input 'clarify,' the CRF-output arrives at a paraphrase thereof by using a grammatical construction of the transitive verb for 'give' followed by a noun phrase inflected with the accusative case, in both the adjective and the noun with which it agrees. The baseline translation, on the other hand follows 'give' with a direct object in the nominative case, 'visible,' morphologically derived from the verb 'see.'

(12) Input: 'the charter we are to approve today both strengthens and gives visible shape to the common fundamental rights and values our community is to be based upon.'

a. Reference translation: perusoikeuskirja , jonka tänään aiomme hyväksyä , sekä vahvistaa että **selventää niitä** yhteisiä perusoikeuksia ja -arvoja , joiden on oltava yhteisömme perusta.
selventää/VERB/ACT/INF/SG/LAT-clarify
se/PRONOUN/PL/PAR-them
Back-translation: 'Charter of Fundamental Rights, which today we are going to accept that clarify and strengthen the common fundamental rights and values, which must be community based.'

b. Baseline: perusoikeuskirja me hyvksymme tnn molemmat vahvistaa ja **antaa näkyvä** muokata yhteistä perusoikeuksia ja arvoja on perustuttava.
antaa/VERB/INF/SG/LAT-give
näkyä/VERB/ACT/PCP/SG/NOM-visible
Back-translation: 'Charter today, we accept both confirm and modify to make a visible and common values, fundamental rights must be based.'

c. CRF: perusoikeuskirja on hyväksytty tänään , sekä vahvistaa ja **antaa konkreettisen muodon** yhteisiä perusoikeuksia ja perusarvoja , yhteisön on perustuttava.
antaa/VERB/ACT/INF/SG/LAT-give
konkreettinen/ADJECTIVE/SG/GEN,ACC-concrete
muoto/NOUN/SG/GEN,ACC-shape
Back-translation: 'Charter has been approved today, and to strengthen and give concrete shape to the common basic rights and fundamental values, the Community must be based.'

In this example, the prediction model has correctly captured the transitive construction 'give shape,' contributing to its fluency. To help clarify the constructions in question, we have used Google Translate to provide back-translations of our MT output into English; this is not the work of a human translator. In order to view these back-translations in the proper context, we have provided Google's back-translation of the reference.

Another general pattern is that the baseline tends to prefer the least marked form for noun cases (corresponding to the nominative) more than the reference or the CRF model. In the reference, only 30.98% of nouns are unmarked for case, and thus in the nominative. The baseline significantly increases this ratio with 35.96% of all nouns in the (unmarked) nominative, but the CRF model does better with 32.02% nominative, so it seems to fare somewhat better at using explicitly marked forms, rather than defaulting to the dominant unmarked form.

The use of postpositions shows another difference between the models. Finnish postpositions require the preceding noun to be in the genitive or sometimes partitive case. This happens correctly 70% of the time in the baseline, but is a somewhat better in the prediction model, which correctly marks the preceding noun 72.52% of the time. The following is an example of variants of this construction between the models for the input ‘with the basque nationalists’:

(13) Reference:

baskimaan kansallismielisten kanssa
basque-SG/NOM+land-SG/GEN,ACC

nationalists-PL/GEN with-POST

(14) Baseline:

baskimaan
basque-SG/NOM-+land-SG/GEN,ACC
kansallismieliset kanssa
kansallismielinen-PL/NOM,ACC-nationalists POST-with

(15) CRF:

kansallismielisten baskien kanssa
nationalists-PL/GEN basques-PL/GEN with-POST

In the above example, all three translations roughly correspond to the same English text. ‘Baskimaan’ is a compound form, composed of ‘basque+land,’ while ‘baskien’ simply means ‘basques.’ However, the CRF output is more grammatical than the baseline, in the sense that not only do the adjective and noun agree for case, but the noun ‘baskien’ to which the postposition ‘kanssa’ belongs is marked with the correct genitive case. However, this aspect of well-formedness is not rewarded by the BLEU evaluation measure, because ‘baskien’ does not match the reference ‘baskimaan.’ Interestingly, in the L-match segmented translation model, the translation marked nouns in the genitive more frequently than was warranted by the reference translation.

We also looked at the morphological reflexes of possession. Finnish may expresses possession using case marking alone, but it has another construction for expressing possession; this can be used to disambiguate an otherwise ambiguous clause. This alternate genitive construction uses pronoun in the genitive case followed by a noun marked with a possessive suffix. We found that in this construction, the prediction model correctly marks the noun with the possessive 54.17% of the time, while the baseline gets it right 48.68%. That this pattern crosses intervening adjectives, numerals, and conjunctions would seem to indicate the strength of the CRF model in using long-distance features to capture morphological dependencies.

(16) Input: ‘and in this respect we should value the latest measures from commissioner fischler , the results of **his trip to morocco on the 26th of last month** and the high level meetings that took place, including the one with the king himself’

a. Reference translation: ja tässä mielessä osaamme myös arvostaa komission jäsen fischlerin viimeisimpiä toimia , jotka ovat **hänen marokkoon 26 lokakuuta tekemns matkan** ja korkean tason kokousten jopa itsensä kuninkaan kanssa tulosta
Back-translation: ‘and in this sense we can also appreciate the Commissioner Fischler’s latest actions, which are **his to Morocco 26 October trip** to high-level meetings and even the king himself with the result/

hänen/GEN-his ... tekemänsä/POSS-his matkan/GEN-tour

b. CRF: ja tässä yhteydessä meidän olisi lisäarvoa viimeistä toimenpiteitä kuin komission jäsen fischler , että **hänen kokemuksensa marokolle** viime kuun 26 ja korkean tason tapaamiset järjestettiin, kuninkaan kanssa
Back-translation: ‘and in this context, we should value the last measures as the Commissioner Fischler, that **his experience in Morocco** has on the 26th and high-level meetings took place, including with the king.’

hänen/GEN-his kokemuksensa/POSS-experience marokolle-Moroccan

c. BASELINE: ja tässä yhteydessä olisi arvoa viimeisin toimia komission jäsen fischler , tulokset monitulkintaisia **marokon yhteydessä** , ja viime kuussa pidettiin korkean tason kokouksissa , mukaan luettuna kuninkaan kanssa
Back-translation: ‘and in this context would be the value of the last act, Commissioner Fischler, the results of **the Moroccan context**, ambiguous, and last month held high level meetings, including with the king’

yhteydess/INE-connection

While neither model correctly translates ‘matkan’ (‘trip’), the baseline’s output attributes the inessive ‘yhteydess’ (‘connection’) as belonging to ‘tulokset’ (‘results’), and misses marking the possession linking it to ‘Commissioner Fischler’.

One last morphological difference between the prediction model and the baseline (not included in Table 4.9) can be seen with interrogative sentences. Rather than relying on punctuation, Finnish can use a word-final morpheme to express that a sentence is a question. This morpheme occurs 86 times in the reference, 89 times in the prediction model, and 94 times in the baseline. In a 2,000 sentence test set, this is a small difference, but it is noted herein because mismarking a declarative sentence as a question seems to be an error that would matter in human evaluation. In contrast, in the segmented translation, this interrogative morpheme was not used at all.

Factors such as those described above contribute to fluency but operate across sentences on the subword level and are therefore not measured by the BLEU evaluation measure.

4.6 Chapter Summary

This chapter gives an account of CRFs and their model family, after contrasting the main approaches towards using morphological information in SMT. The factored model approach allows access to both lexical and morphological information, but is better suited for use with supervised segmentation methods, and moreover does not allow for the productive combination of inflectional morphology across phrase boundaries. The segmented translation approach can improve the accuracy of the MT system, but is highly dependent on the segmentation used, which is quite prone to either undersegmentation or overfitting. The morphology generation approach can have high accuracy rates for correct morphology prediction, but is applied in post-processing only and does not allow the translation model to benefit from that morphology which may correspond to lexical content.

We next detailed several experiments using a CRF morphology prediction model on MT output. We found that the supervised segmentation, with its conflation of morphological categories, made fully inflected surface forms difficult. However, we found that using an unsupervised segmentation-trained segmented translation model in conjunction with the

CRF post-processing prediction model slightly increased BLEU scores and increased several aspects of translation fluency on the sub-word level.

Chapter 5

Conclusion

In this thesis, we examined the main approaches to using morphological information in SMT, and produced and tested systems for each approach.

5.1 Thesis Summary

Chapter 2 described a means of using factored models to incorporate morphological information into the MT task. We described our experiments using factored models with morphological analysis from unsupervised models.

In Chapter 3, we discussed the challenge of finding the unsupervised segmentation type that yields best results for MT, and we described the Morfessor model for performing unsupervised morphological analysis. We then described a series of experiments using the Morfessor-derived segmentations to train the MT system, and evaluated their results using the BLEU measure against whole-word and segmented forms of the reference and output translations.

In Chapter 4, we outlined the use of a CRF-based classifier to predict morph endings as a post-process. We made a comparison of the translation model using the CRF for morphology generation to the previous approaches seen in Chapters 2 and 3, to analyze how the different models may be used for and affect the MT task for morphologically complex languages. We then described our experiments using the CRF morphology generation model in conjunction with the unsupervised segmented translation model. We evaluated these results against a word-based baseline and analyzed how sub-word factors may be influencing the relative fluency of the systems' output.

Table 5.1 gives a summary of the best translation scores of the each of the models explored in this thesis.

Model	BLEU-Portage	BLEU-WMT	WER	TER
Baseline	14.39	14.68	74.96	72.42
Factored	13.98	14.22	76.68	74.15
Unsup L-match	**15.15**	**15.46**	74.46	71.78
CRF-Unsup	**14.55**	**14.87**	**73.71**	**71.15**

Table 5.1: Summary Model Scores: two versions of BLEU, with WER and TER.

5.2 Contribution

The main contributions of this thesis can be summarized thus:

- We show that using morphological segmentation in the translation model can improve output translation scores. We also demonstrate that for this language pair, phrase-based MT benefits more from allowing the translation model access to morphological segmentation on the same level as the lexical level, rather than in a hierarchical relationship to the surface form or in translation to other non-lexical source-side factors.
- We discover that using a CRF post-processing morphology generation model can improve translation fluency on a sub-word level, in a manner that is not captured by the BLEU word-based evaluation measure.

5.3 Future Research

There are many ways to approach using morphology in phrase-based MT. This begins with where the morphological information comes from, and the examination what type of segmentation model is best suited for the task. While supervised techniques are inherently limited for this task, a possible drawback of the unsupervised segmentation used herein is that for languages with highly complex morphological systems, certain segments are concatenated morphs, while others represent a single morphological category. A possible consequence of this is that morph forms representing overlapping information (NOM versus NOM-PL versus PL) may be competing with each other in translation or generation. Linguistically motivated segmentation, on the other hand, can be used to structure segmentation based on a priori notions of morphological category, rather than substring distributional statistics alone. To this end, in the future we would be interested in examining the use of semi-supervised segmentation techniques that extrapolate a segmentation model from small amount of hand-annotated data, in order to combine coverage and extensibility with more sophisticated representations of morphological category.

Segmentation model aside, it is then an interesting question how to incorporate this into the translation model. In this thesis, we found that the model that predicted morphology in post-processing gained translation output fluency on the subword level, but this had little effect on the word-based evaluation scores. In order to get an upper bound on just how much it is possible for this technique to increase evaluation scores we would like to design an oracle experiment capturing this information precisely.

We also found that simple segmented translation models can outperform word-based models. Thus, we are interested in examining whether discontinuous phrase-based segmented translation may be used to improve further their effectiveness in this task. With discontinuous morph phrases that skip stems, discontinuous segmented translation may be able to use morpheme phrases across words, thus in effect capturing, say, suffix agreement phenomena.

In addition, we hypothesized from the experiments herein that the factored models may be less than ideal for the particular task because they disallow morphological recombination and predicate all morphology generation on the local stem. We would therefore like to examine the use of segmented factored translation models, wherein morphs are treated as the surface form factor, and additional factors would be more general morphological forms, using automatically derived morph classes in order to accomplish this while still using unsupervised morphological analysis.

Another type of translation model that we have not yet explored with the use of morphology is syntax-based MT. Like the discontinuous phrase-based models noted above, syntax-based MT can translate discontinuous units. In addition, these models translate on the basis of hierarchically structured constituents, and therefore promise to be a way of capturing dependencies between morphological constituents within the translation model, as well as how grammatical morphology might express word-order-based syntactic relations in non-inflecting languages.

An issue that begs further investigation is that of evaluation measures and tuning. Not only does the standard measure ignore sub-word information, but since the model weights are tuned to optimizing this measure, it may have reflexes in the translation model itself. Therefore, it would be useful to design a measure (for evaluation and as a tuning drop-in) that considers translations on both the word and segmented level, though these two representations might be weighted differently.

Finally, we would like to test the techniques developed herein, as well as those described above, on other languages with rich morphological systems besides the language pair used in this thesis.

Chapter 6

List of Morphological Abbreviations

2P	second person
ABL	ablative case
ACC	accusative case
ACT	active voice
ADE	adessive
ADJ	adjective
AFF	affirmative
AUX	auxiliary
COND	conditional mood
ESS	essive case
GEN	genitive case
ILL	illative case
INE	inessive case
INF	infinitive
INT	interrogative case
LAT	lative case
LOC	locative case
NN	singular nominal
NNS	plural nominal
NON	non-morpheme
NOUN	noun
PAR	partitive
PAST	past tense
PCP	participle
PL	plural
POSS	possessive
POST	postposition
REL	relational
SG	singular
SG1	first person singular
STM	stem
SUF	suffix
TRA	translative

References

Avramidis, Eleftherios and Philipp Koehn. 2008. Enriching morphologically poor languages for statistical machine translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 763–770, Columbus, Ohio, USA. Association for Computational Linguistics.

Banchs, Rafael and Haizhou Li. 2008. Exploring Spanish-morphology in effects on Chinese-Spanish SMT. In *Proceedings of the Workshop on Mixing Approaches to Machine Translation*, pages 49–54, Donostia, Spain.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Cer, Daniel, Christopher Manning, and Daniel Jurafsky. 2010. The best lexical metric for phrase-based statistical MT system optimization. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 555–563, Los Angeles, California, USA. Association for Computational Linguistics.

Chang, Pi-Chuan, Michel Galley, and Christopher Manning. 2008. Optimizing Chinese word segmentation for machine translation performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 224–232, Columbus, Ohio, USA. Association for Computational Linguistics.

Chiang, David and Kevin Knight. 2009. 11,001 new features for statistical machine translation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 218–226, Boulder, Colorado. Association for Computational Linguistics.

Chung, Tagyoung and Daniel Gildea. 2009. Unsupervised tokenization for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 718–726, Singapore. Association for Computational Linguistics.

Creutz, Mathias and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR'05)*, pages 106–113, Espoo, Finland.

Dasgupta, Sajib and Vincent Ng. 2007. High-performance, language-independent morphological segmentation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 155–163, Rochester, New York, USA. Association for Computational Linguistics.

de Gispert, Adriá and José Mariño. 2008. On the impact of morphology in English to Spanish statistical MT. *Speech Communication*, 50(11-12).

de Gispert, Adriá, José Mariño, and Josep Crego. 2005. Improving statistical machine translation by classifying and generalizing inflected verb forms. In *Proceedings of the*

Ninth European Conference of Speech Communication and Technology, Interspeech'05, pages 3193–3196, Lisbon, Portugal. International Speech Communication Association.

Dyer, Chris. 2009. Using a maximum entropy model to build segmentation lattices for MT. In *Proceedings of the Joint Conference of Human Language Technologies and the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 406–414, Boulder, Colorado, USA. Association for Computational Linguistics.

Galley, Michel and Christopher Manning. 2010. Accurate non-hierarchical phrase-based translation. In *Proceedings of The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 966–974, Los Angeles, California. Association for Computational Linguistics.

Goldwater, Sharon and David McClosky. 2005. Improving statistical MT through morphological analysis. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 676–683, Vancouver, B.C., Canada. Association for Computational Linguistics.

Habash, Nizar and Fatiha Sadat. 2006. Arabic preprocessing schemes for statistical machine translation. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 49–52, New York, New York, USA. Association for Computational Linguistics.

John Lafferty, Andrew McCallum and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, San Francisco, California, USA. Association for Computing Machinery.

Knight, Kevin. 1999. A statistical MT tutorial workbook. http://www.isi.edu/natural-language/mt/wkbk.rtf.

Koehn, Philipp. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the Empirical Methods in Natural Language Processing (EMNLP)*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Koehn, Philipp. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X*, pages 79–86, Phuket, Thailand. Association for Computational Linguistics.

Koehn, Philipp, Marcello Federico, Wade Shen, Nicola Bertoldi, Ondrej Bojar, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Richard Zens, Alexandra Constantin, Christine Corbett Moran, and Evan Herbst. 2007. Open source toolkit for statistical machine translation: factored translation models and confusion network decoding. Technical report, Johns Hopkins University.

Koehn, Philipp and Hieu Hoang. 2007. Factored translation models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 868–876, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL '07: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–108, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, Philipp and Kevin Knight. 2003. Empirical methods for compound splitting. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 187–193, Budapest, Hungary. Association for Computational Linguistics.

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

Lee, Young-Suk. 2004. Morphological analysis for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pages 57–60, Boston, Massachusetts. Association for Computational Linguistics.

Lindén, Krister and Jussi Tuovila. 2009. Corpus-based paradigm selection for morphological entries. In *Proceedings of the 17th Nordic Conference on Computational Linguistics (NODALIDA)*, pages 96–102, Odense, Denmark.

Ma, Yanjun and Andy Way. 2009. Bilingually motivated word segmentation for statistical machine translation. *ACM Transactions on Asian Language Information Processing*, 8(2):1–24.

Martin, Joel, Howard Johnson, Benoit Farley, and Anna Maclachlan. 2003. Aligning and using an English Inuktitut parallel corpus. In *Proceedings of the HLT-NAACL Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 115–118, Edmonton, Alberta, Canada. Association for Computational Linguistics.

Matusov, Evgeny, Richard Zens, and Hermann Ney. 2004. Symmetric word alignments for statistical machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 219–225, University of Geneva, Switzerland. Association for Computational Linguistics.

McCallum, Andrew, Dayne Freitag, and Fernando Pereira. 2000. Maximum entropy markov models for information extraction and segmentation. In *Proceedings of the 17th International Conference on Machine Learning*, pages 591–598, Stanford University, Palo Alto, California, USA. Association for Computing Machinery.

Minkov, Einat, Kristina Toutanova, and Hisami Suzuki. 2007. Generating complex morphology for machine translation. In *In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL07)*, pages 128–135, Prague, Czech Republic. Association for Computational Linguistics.

2005. Inuktitut morphological analzer. http://www.inuktitutcomputing.ca/Uqailaut/en/-IMA.html.

Neißen, Sonja, Franz Joseph Och, and Hermann Ney. 2000. An evaluation tool for machine translation: Fast evaluation for MT research. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC)*, pages 39–45, Athens, Greece. European Language Resources Association (ELRA).

Och, Franz Josef. 1999. An efficient method for determining bilingual word classes. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, pages 71–76, University of Bergen, Bergen, Norway. Association for Computational Linguistics.

Och, Franz Josef. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of 2003 Conference of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan. Association for Computational Linguistics.

Och, Franz Joseph and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong. Association for Computational Linguistics.

Oflazer, Kemal and Ilknur Durgar El-Kahlout. 2007. Exploring different representational units in english-to-turkish statistical machine translation. In *Proceedings of the Statistical Machine Translation Workshop at ACL 2007*, pages 25–32, Prague, Czech Republic. Association for Computational Linguistics.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics ACL*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Pirinen, Tommi and Inari Listenmaa. 2007. Omorfi morphological analzer. http://wiki.apertium.org/wiki/Omorfi.

Popoviç, Maja and Hermann Ney. 2004. Towards the use of word stems and suffixes for statistical machine translation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, pages 1585–1588, Lisbon, Portugal. European Language Resources Association (ELRA).

Rabiner, Lawrence. 1989. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Ramanathan, Ananthakrishnan, Hansraj Choudhary, Avishek Ghosh, and Pushpak Bhattacharyya. 2009. Case markers and morphology: Addressing the crux of the fluency problem in English-Hindi SMT. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 800–808, Suntec, Singapore. Association for Computational Linguistics.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2000. A study of translation edit rate with targeted human annotation. In *Proceedings of the Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts.

Snyder, Benjamin and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 737–745, Columbus, Ohio, USA. Association for Computational Linguistics.

Stolcke, Andreas. 2002. Srilm – an extensible language modeling toolkit. *7th International Conference on Spoken Language Processing*, 3:901–904.

Sulkala, Helena and Merja Karjalainen. 2008. *Finnish*. Routledge.

Toutanova, Kristina, Dan Klein, Christopher Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180, Edmonton, Alberta, Canada. Association for Computational Linguistics.

Toutanova, Kristina, Hisami Suzuki, and Achim Ruopp. 2008. Applying morphology generation models to machine translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 514–522, Columbus, Ohio, USA. Association for Computational Linguistics.

Tsuruoka, Yoshimasa and Jun'ichi Tsujii. 2005. Bidirectional inference with the easiest-first strategy for tagging sequence data. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 467–474, Vancouver, B.C., Canada. Association for Computational Linguistics.

Väyrynen, Jaakko, Mathias Creutz, and Markus Sadeniemi. 2007. Morphology-aware statistical machine translation based on morphs induced in an unsupervised manner. In *Proceedings of the Machine Translation Summit XI*, pages 491–498, Copenhagen, Denmark. Association for Computational Linguistics.

Wintner, Shuly. 2002. Morphology. Course Notes in Computational Linguistics, University of Haifa.

Xu, Jia, Richard Zens, and Hermann Ney. 2004. Do we need chinese word segmentation for statistical machine translation? In *Proceedings of the Third SIGHAN Workshop on Chinese Language Learning*, pages 122–128, Barcelona, Spain. Association for Computational Linguistics.

Yang, Mei and Katrin Kirchhoff. 2006. Phrase-based backoff models for machine translation of highly inflected languages. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, pages 41–48, Trento, Italy. Association for Computational Linguistics.

Yang, Mei, Jing Zheng, and Andreas Kathol. 2007. A semi-supervised learning approach for morpheme segmentation for an arabic dialect. In *Proceedings of Interspeech*, pages 1501–1504, Antwerp, Belgium. International Speech Communication Association.

Zhang, Ruiqiang, Keiji Yasuda, and Eiichiro Sumita. 2008. Chinese word segmentation and statistical machine translation. *ACM Transactions of Speech and Language Processing*, 5(2):1–19.