Bilingual Word and Chunk Alignment: A Hybrid System for Amharic and English

Saba Amsalu Teserra

A thesis submitted in fulfillment of

the requirements for the degree of

Doctor of Philosophy

Fakultät für Linguistik und Literaturwissenschaft

Universität Bielefeld

June 2007

1. Examiner: Prof. Dr. Dafydd Gibbon

2. Examiner: Prof. Dr. Dieter Metzing

Declaration

This thesis is submitted to the University of Bielefeld in fulfillment of the requirements for the degree of Doctor of Philosophy in Computational Lingusitics. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Saba Amsalu Teserra

June, 2007

Dedication

To my father

Acknowledgements

The research that has gone into this thesis has benefited greatly from the interaction that I have had with my supervisors and colleagues, as well as friends.

I feel very privileged to have worked with my supervisors, Dafydd Gibbon, and Dieter Metzing. To each of them I owe a great debt of gratitude for their patience, inspiration and friendship. Dafydd, I specially thank you for giving me this chance to come all the way to Germany and take this step in my career. I have greatly benefited from your excellence in producing ideas instantly and your all-round deep knowledge in computational linguistics.

My regard for Dieter Metzing is enormous. Thank you for your experienced help in directing me on how to approach some of the problems that I had in my thesis. Thank you very much for the time you spent with me as a friend and as a supervisor. You have also shared all the complex administrative and financial problems I had.

I would also like to thank Henrike Wanke for her diligent support and patience whenever I asked for it. Many thanks Ms. Wanke!

Many thanks go to my friends who are a bunch of nice people with whom

I had lots of fun. With you guys, managing all the stress was much easier.

The Deutsche Forschungsgemeinschaft (DFG) was very generous in giving me a two year scholarship. Bielefeld University also provided me with a one year scholarship to finish this thesis. The Faculty of Linguistics and Literature has provided an excellent environment for my research by financing my German courses.

Finally, I thank my family who have been extremely understanding and supportive of my studies. Ribka, I particularly thank you for not being weary of having to go through my emotional ups and downs during this time of study. Ababa, I thank you very much for your patience, diligence and perseverance in having to teach me when I was a little girl. I would have not made it without you. You are also a very good father.

Contents

List of Tables

List of Figures

Abstract

This thesis presents efficient word alignment algorithms for sentence-aligned parallel corpora. For the most part, procedures for statistical translation modelling are employed that make use of measures of term distribution as the basis for finding correlations between terms. Linguistic rules of morphology and syntax have also been used to complement the statistical methods. Two models have been developed which are briefly described as follows:

Alignment Model I

For this first model a statistical *global alignment* method has been designed in which the entire target document is searched for the translation of a source word . The term in the target language that has the highest similarity in distribution with the source term is taken as the best translation. The output of this algorithm is a 1:1 alignment of a complex Amharic word with simple English words.

In reality, one word in one language is not necessarily translated into a single word in the other language and vice versa. This phenomenon is even more pronounced in disparate languages such as English and Amharic. Therefore, an enhancement method, *relaxing routine*, that would scale up the

1:1 alignments into 1:m alignments is devised. This approach that synthesises English chunks that are equivalent to Amharic words from parallel corpora is also described in this study. The procedure allows several words in the simpler language to be brought together and form a chunk equivalent to the complex word in the other language.

The relaxing procedure may resolve the shortcomings of a 1:1 alignment but it does not solve the distortion in the statistics of words created by morphological variants, hence *finite-state shallow stemmers* that strip salient affixes in both languages have also been developed.

Alignment Model II

Model II performs *local alignment* of a source word in a source sentence in the source language to a word in the target sentence in the target language. The search for a translation of a word in a sentence is only limited to the corresponding sentences instead of the entire document. This is a step towards achieving an increased recall, which is vital when dealing with languages that have scarcity of translation texts. This procedure, however, results in a drop in precision. To improve the diminished precision, two procedures have been integrated into it:

1. Reuse of the lexicon from model I, that is, known translations are excluded from the search space, leaving a limited number of words from which to choose the most likely translation; and

2. a pattern recognition approach for recognising morphological and syntactic features that allows the guessing of translations in sentences has also been developed.

A comparative study of the performance of Model I across Amharic, English, Hebrew and German was also part of the study. The impact of the complexities and typological disparities on the performance of the alignment method has been observed.

Another attempt to exploit translation texts that has been made in the course of this research was an attempt to recognise nouns in Amharic by transfer from German translation. Since nouns in German are recognised by their initial capital, aligning nouns leads to the recognition of nouns in Amharic, which do not have special features that distinguish them from words in other word classes.

All the components of the system have been evaluated on text aligned at sentence level. On the same data, a comparison with the IBM alignment model implementation (GIZA++) has also been made.

Chapter 1

Introduction

Lexical resources are useful for diverse natural language processing and language engineering applications. In particular, bilingual lexica are invaluable for coping with the escalating need to access information in various languages. Hence, the need to have cross-lingual lexical resources is pressing. Bilingual applications such as machine translation and cross-language information retrieval systems are dependent on machine readable lexical resources in multiple languages. Coming up with these lexica, however, is very difficult. Such resources used to be hand coded in earlier days. Today, they are explicitly built automatically or semi-automatically. Manually built lexica are of high quality but are time consuming, labour intensive and costly. They are also relatively static, and not so easy to upgrade for new domains. Automatic methods of bilingual lexicon acquisition, however, promise to provide fast, cheap and dynamic alternatives though they require huge amounts of data and efficient tools to produce high quality results. In addition corpora

allow us to generate not only word lexica but also multistring lexica such as phrasal lexica. Special expressions, newly introduced terms and domain specific terms are easier to capture in corpora than in any other way.

1.1 Data-driven bilingual lexicon compilation

Automatic methods of building lexica are primarily based on existing data. For bilingual lexicon compilation, one needs to have texts from which one can extract data in two languages. One major data source for such applications is existing translations. Translation texts also called *parallel texts* are composed of text in one language and its translations in one or more other languages.

Human translations consist of text interpretations which reflect the cultural, social and linguistic typologies of the languages to which translations are made. Parallel texts are a valuable source of a kind of linguistic meta-knowledge used for diverse purposes. Their usefulness has been stated over and over again by different scholars in the past two decades [Kay, 2000, Isabelle, 1992, Somers, 2001, Vèronis, 2000].

[Kay, 2000] asserts that parallel corpora are without parallel in history as a source of data for terminology banks and bilingual dictionaries. Addressing the problems of human translators, [Isabelle, 1992] also stated existing translations provide more solutions for translators than any other resource. No other resource is rich enough in providing translations in different contexts, specific to certain topics or on recently introduced terms. Parallel texts are also praised for being excellent sources for language learning, machine-aided

human translation, machine translation systems, word sense disambiguation, etc. [Melamed, 2001, Brown et al., 1993, Koehn, 2004a]. The knowledge of the benefits of translation texts has, hence, led to floods of research on methods of exploiting them. Some historical accounts on the directions and progress of studies in the area are presented below.

1.1.1 History

The idea of using translation texts for obtaining translation information has been around since the seventies as mentioned by [Melby, 1981, Kay, 1980]. [Melby, 1981] made a proposal to store past translations electronically for bilingual concordancing. [Debili et al., 1988, 1989] made an analysis on the use of different kinds of reformulations used in information retrieval systems where full text databases are accessed through natural language queries. It was also at this time that Harris coined the term *bitexts* to refer to a text in one language and its translation in a second language [Harris, 1988a,b].

But the first attempt to align these translation texts was made by Martin Kay and Martin Röscheisen in 1987 [Kay and Röscheisen, 1988]. At the same time, Gale and Church were working on an alternative approach for sentence alignment [Gale and Church, 1991]. Initial proposals On memory-based machine translation systems were made in 1990 by [Sato and Nagao, 1990] which was a major step in the direction of example-based machine translation today. [Church and Gale, 1991] made bilingual concordances available for lexicographers the following year. They also developed a program for aligning sentences [Gale and Church, 1991]. In 1992 [Simard et al., 1992]

introduced the use of cognates to align sentences. One of the well known alignment systems *Char_align*, which aligns texts at the character level, was then developed by Church in 1993 [Church, 1993]. During that time, Isabelle studied using translation texts as an aid to machine-aided human translation [Isabelle, 1992, 1993]. More sophisticated studies on using linguistic rules together with statistical methods for aligning sub-sentence text units such as phrases and clauses were developed in the following years [Kupiec, 1993, Smadja et al., 1996a,b, McEnery et al., 1997, Och and Ney, 2002, Melamed, 1996c, 2000, 2001, Ahrenberg et al., 1998, Boutsis and Piperidis, 1998a,b].

1.1.2 Text alignment

Extracting bilingual lexica from parallel corpora involves the difficult task of aligning text units. This is a non-trivial task for several reasons which may be related to the properties of the languages considered or to the common problems that come with translated documents such as deletion, insertion, splitting, merging, etc. The problem gets even more challenging when the languages considered are disparate. Often other tools, such as morphological analysers and taggers which may not be available for resource-poor languages are required.

Aligning lexical units often follows a top down approach of aligning bigger chunks of text such as paragraphs and sentences first and then progressively aligning the smaller units such as words and phrases in the chunks which are also at times iterative where aligned smaller units could be used to align larger units (See Figure 1.1). The part enclosed by the circle with broken

lines is the focus of this work.

Alignment algorithms that have been developed so far use probabilistic models and/or rule based models that integrate linguistic information. Probabilistic models are highly dependent on the statistics of words in texts to determine relation of words and expressions. They are more commonly used on language pairs that have high similarity and also on those that have a relatively less complex morphological structure. Linguistic information of morphology, syntax, cognateness, synonymy, word classes etc. have also been used to identify equivalent lexical entities. Linguistic approaches are predominantly used to process disparate languages where statistical information could not help much.

1.1.3 The problem of text alignment

Despite these efforts, there are problems that are not yet addressed by existing alignment algorithms. These include:

1. Most of the methods used are very suitable for use on major languages such as English and French and haven't yet been proved to be applicable to other languages;

2. Alignment methods have also focused on historically related languages that share a large proportion of their vocabulary and that are typologically similar;

3. Not much has been done on simple-complex language pairs;

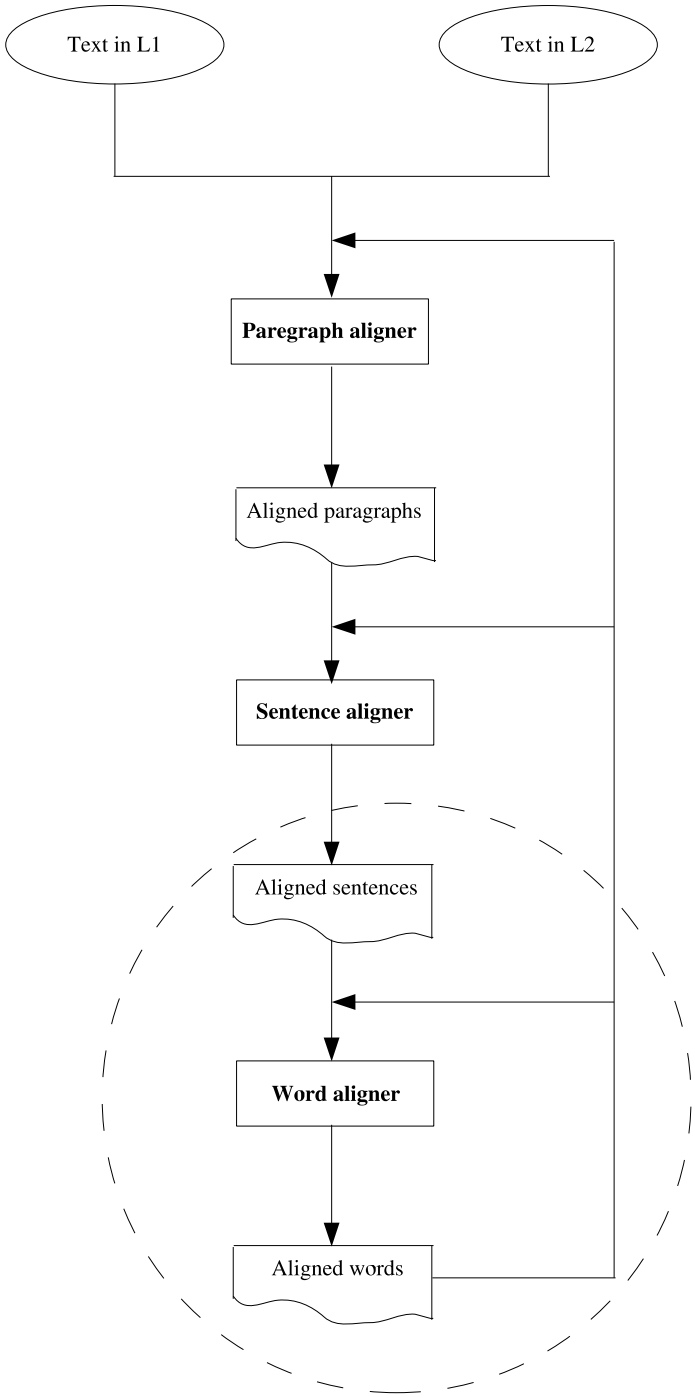


Figure 1.1: Iterative alignment.

4. Not much has been done with languages for which there exists a limited number of translation texts;

5. Languages that are not computationally exploited and hence have scarce or no linguistic tools haven't been addressed;

6. Most alignment algorithms only give outputs of partial matches which makes their applicability far from clear;

7. Most algorithms depend on external knowledge sources such as machine readable dictionaries, but it is not clear why one would try to gain dictionaries from dictionaries;

8. It is not very clear how far one can go without the use of linguistic tools such as morphological analysers and parsers.

1.2 Contribution

In this thesis, an attempt to address the basic problems mentioned above has been made.

1. This thesis is an experiment on complex-simple language pairs in which smaller translation units are difficult to extract. In complex languages, the base forms of words are inflected in complex ways. A single word consists of semantic plus grammatical components. When trying to align such words to simple words which either are lexical or grammatical words, the relations are difficult to model.

2. Amharic and English do not share much of their vocabulary. This results in the exclusion of cognate information from being an easy tool to use.

3. Scarcity of linguistic tools is a bottle neck for many natural language processing applications. In text alignment tools such as morphological analysers, part-of-speech taggers and parsers are of great value. For many languages, including Amharic, such tools are not available. For complex languages, the problem is two fold because without these tools, understanding sentence and word structures is impossible.

4. Scarcity of translation data is also a major problem. Statistical methods of alignment depend on the availability of huge quantities of data. For languages where there are hundreds or even thousands of variants of words, the use of statistical methods requires a data size which could cope with these variants even more so than simpler languages.

1.3 Organisation of the thesis

The thesis is organised into 9 chapters.

Chapter 2 presents a detailed account of the state of the art in procedures for word alignment. Methods, data and results of major studies in the area are presented. An emphasis on studies of word alignment is made. Statistical translation model development algorithms that have been developed for related and, to some extent, unrelated languages is presented. The use of

linguistic tools and resources for aligning text units is also presented together with some experimental results.

Chapter 3 reports a detailed description of the first component of Model I in our system. This part of the system performs global statistical word alignment to gain improved precision alignment. In the global alignment system a 1:1 alignment of complex Amharic words to simple English words is made. This system is called a global alignment system because even though sentences are already aligned and the words in these sentences have their translation in the target sentences, a search for word translations is made in the target sentence and throughout the whole target document.

Chapter 4 discusses a technique of generating 1:m alignments from the 1:1 alignment lexicon in the global alignment procedure. This basically is a chunking system that makes shallow parsing of English sentences in search for chunks equivalent to complex Amharic words.

Chapter 5 presents a description of the morphology of Amharic words. Major word formation rules of simple and derived word forms of the various word classes are discussed.

Chapter 6 describes the development and implementation of a shallow finite-state morphological analyser for Amharic and English texts. Inflectional and derivational variants of words are reduced into a common canonical form. This is aimed at improving the precision of the statistics of words in the documents.

Chapter 7 describes Model II of our system which is basically a procedure for maximum likelihood local alignment of words in parallel sentences. It is called a local alignment system because it takes two translation sentences and aligns the words within them in contrast to Model I where the words in the source sentence can be aligned with words that do not exist in the target sentence but elsewhere in the target document. It is a procedure of maximising recall at a cost of precision. Enhancing methods for Model II with the aim of improving the recall are also discussed. Reusing the lexicon produced by Model I and pattern recognition approaches to obtaining prior alignments before running Model II are the main components of this chapter.

Chapter 8 presents the evaluation methods used and the results obtained. Both Model I and Model II as well as the accompanying chunker, shallow stemmer and enhancement methods are evaluated on data taken from the bible.

An experiment made to test the performance of Model I on four different languages is also presented. The goal is to compare what the impact of the difference in typologies would have on the performance of a statistical word alignment system. The languages considered are Amharic, English, Hebrew and German.

The performance of Model I in enriching monolingual text is also tested. This test presents a system that maps German nouns to their translations in Amharic. The spelling of German nouns which always begin with a capital letter is taken as an information to identify the nouns in Amharic which do not have special marking. The goal is spotting Amharic nouns in text.

Chapter 9 discusses overall conclusions that have been drawn from the studies reported in the preceding chapters. Based on the results obtained, areas open to future research are also identified, as well as possibly better methods which have not been investigated within the scope of this work.

Appendix contains source code for different text processing and alignment algorithms, each of which is implemented in C++. The source code for each algorithm includes documentation that gives a guide on how to install it, the input and output format of data and files, as well as how to use the interface and command line to achieve the required output. Lexicon of roots of verbs and finite state code, as well as miscellaneous code, are also included in this section.

Chapter 2

Describing Word Alignment Algorithms

This chapter discusses the state of the art techniques for aligning texts, with greater emphasis to word alignment. A review of the different statistical and linguistic as well as hybrid approaches is presented. A brief account of types of data sources used, filtering and refining procedures and results obtained is also provided.

2.1 Objectives

The need for exploiting parallel corpora has urged many scientists to come up with methods of aligning translation text. As a result, several algorithms of matching paragraphs, sentences, words, etc. have been developed. Among others, studies by [Kay and Röscheisen, 1988, 1993, Klavans and Tzoukermann, 1990, Gale and Church, 1991, Simard et al., 1992, Brown et al.,

1990, 1993, Chen, 1993, Church, 1993, Church and Helfman, 1993, Dagan and Church, 1994, Fung and Church, 1994, Och and Ney, 2002] are some of the leading works. These studies were directed in extracting lexical relations relevant for various language engineering research. Some of these applications are: machine translation [Brown et al., 1990, 1993, Sato and Nagao, 1990]; terminology and translation aids [Isabelle, 1992, Dagan et al., 1993]; word sense disambiguation [Brown et al., 1991a, Tufis et al., 2004] and bilingual lexicography [Klavans and Tzoukermann, 1990]. Each of these and other applications require lexicon of their own art.

2.2 Applications

The applications of aligned texts are extremely diverse, and include compiling translation memories, deriving dictionaries and bilingual terminology lists, extracting knowledge for cross-language information retrieval, extracting examples for machine translation and computer assisted teaching or contrastive linguists, etc. It is these applications that encouraged researchers to invest more effort on the area. A review of the need and relevance of aligned texts for some of the application areas is presented.

2.2.1 Machine translation

One of the major applications for which data-driven bilingual lexica are used are machine translation systems. Machine translation systems require bilingual lexica from which to get translations of terms. Todate the well known

data-driven approaches to machine translation are basically two: Example based machine translation (EBMT) [Nagao, 1984, Sato and Nagao, 1990, Sadler, 1989a,b, Sumita and Tsutsumi, 1988, Sumita et al., 1990] and statistical machine translation (SMT) [Arad, 1991, Brown et al., 1992]. The basis for SMT are basically word translations. Lately the use of bigram or trigram models [Koehn et al., 2003, Koehn, 2004a,b] has been practised because translations are not always one to one correspondence of words and also because such sequences of strings or phrases lead to sentences faster than the word to word translations. In example based translation systems, translation data of bigger chunks are used. The idea behind example based translation system is a kind of translation gathering made by a second language learner where one translates an entire expression, phrases or clauses heard or read before to construct new translations [Nagao, 1984]. The type of bilingual lexicon required by machine translation systems is thus a database of translation units of words or phrases or clauses that were used in previous translations. Efficient algorithms that do align sentences or subsentence chunks of any level are thus very important tools to achieve such structures from parallel corpora.

2.2.2 Bilingual lexicography

Word alignment can be used to create bilingual translation dictionaries of the vocabulary included in parallel corpora. Bilingual dictionaries generated from translation corpora are of utmost importance in providing translation of terms in different contexts. Domain specific dictionaries without doubt

are built exhaustively and precisely from translation documents in certain domains. There are usually a large number of alignment alternatives for most of the automatically aligned words and phrases. That is a word in a source document may be aligned with several alternative words in the target document. It is believed that processing these alignment alternatives may lead to the automatic discovery of morphological and semantic relations between them. The alternative words are often either inflectional variants of each other, that share their semantics, or that they are translations of homonymic or polysemous words.

Apparently, lexicographers discovered parallel data as a valuable resource even for monolingual investigations. The approach described in [Tiedemann, 2001] uses simple filters with different settings in order to find relational categories between alignment alternatives.

2.2.3 Computer-assisted language learning

Computer-assisted Language learning (CALL) refers to programs designed to help people learn foreign languages. CALL is an innovative approach of second language acquisition. Natural language processing has been enlisted in several ways in CALL: including to carry out bilingual text alignment so that a learner who encounters an unknown word in a second language can see how it was rendered in translation [Nerbonne, 2000, 2002]. Parallel texts together with glosses lay bare the grammatical patterns of a language in a way which is valuable to adult language learners. At advanced levels of language instruction it is common to find courses in comparative structure

which are aimed at informing language instruction through the comparative examination of the grammar. Bilingual concordance and collocation aids from corpora also provide words and their translations including the contexts in which they are used [Chang and Chang, 2004].

2.2.4 Machine-aided human translation

Existing translations are extremely valuable resources that can be exploited with software systems to improve the efficiency of human translation. A human translator can learn how certain source language expressions have been translated from existing translations for improving translation, and improving consistency of translation. Translation memory, dictionaries with multi-word units and non-compositional compounds are useful resources for such purposes [Melamed, 1997]. Translation texts are provide such tools readily.

On the other hand, in translation omissions occur quite frequently and proof-reading is costly. Automatic detection methods avoid costly human proof-reading Melamed [Melamed, 1996a,b, Dagan et al., 1993].

Corpus-based methods start from translations that have already been produced by humans and seek to discover their structure, completely or partially. This analysis-oriented perspective lends itself naturally to the development of translator's aids because in Machine-aided human translation (MAHT) the machine is not expected to produce the translations, but rather to understand enough about them to become helpful [Isabelle, 1993].

2.2.5 Cross-language information retrieval

Information retrieval systems that retrieve documents from more than one languages can use bilingual lexica by which query words are translated and the search is carried out in different languages. Corpus-based bilingual term substitution thesaurus is proved to substantially outperform traditional dictionaries [Hull and Grefenstate, 1996, Ballestros and Croft, 1997, Yang et al., 1998]. Such systems also outperformed the cross-language versions of the Generalised Vector Space Model (GVSM) and Latent Semantic Indexing (LSI) techniques. Domain specific bilingual lexica, particularly, provide very useful support in getting the sense of words in a specified context. Such kind of bilingual dictionaries are simply generated by searching repeated co-occurrence.

To use the automatically extracted dictionary for information retrieval, each of the words in the original query are substituted by the possible translations into a new query in the other language. This new query is then used for monolingual retrieval in the document collection. Retrieval performance can be enhanced by giving weights to the more likely translations of each term as obtained in the frequency information in the bilingual dictionary. Such approaches are supposed to have a performance approaching monolingual accuracy [Brown et al., 2000].

2.2.6 Word sense disambiguation

The task of word sense disambiguation (WSD) is to determine the correct meaning, or sense of a word in context. It is a fundamental problem in natu-

ral language processing (NLP). The ability to disambiguate word sense accurately is important for applications such as machine translation, information retrieval, etc. Corpus-based supervised machine learning methods have been used to tackle the WSD task [Ng et al., 2003, Márton Mihált, 2006]. Among the various approaches to WSD, the supervised learning approach is the most successful todate. One source to look for potential training data for WSD are parallel texts [Resnik and Yarowsky, 1997, Diab and Resnik, 2002]. Given a word-aligned parallel corpus, the different translations in a target language serve as the sense-tags of an ambiguous word in the source language. Different dictionaries define a different sense inventory. Tying sense distinction to the different translations in a target language, introduces a data-oriented view to sense distinction and serves to add an element of objectivity to sense definition. The outcome of word sense disambiguation of a source language word is the selection of a target word, which directly corresponds to word selection in machine translation.

2.2.7 Paraphrasing

One of the difficulties in Natural Language Processing is the fact that there are many ways to express the same message. Most studies in automatic generation of paraphrases have examined the use of monolingual parallel corpora for paraphrase extraction [Barzilay and McKeown, 2001, Barzilay and Lee, 2003, Pang et al., 2003, Ibrahim et al., 2003]. Recent studies show that by way of alignment techniques for phrase-based statistical machine translation, paraphrases in one language can be identified using a phrase in

another language as a pivot [Bannard and Callison-Burch, 2005].

So much about the applications of aligned bilingual translation texts, a lot has been done to come up with efficient techniques and methods of aligning existing translation texts without which the application described are simply theoretical. As a result different approaches of aligning translation texts have been suggested. These approaches are discussed in detail in the next section.

2.3 Alignment algorithms

Alignment, can be made at different levels of text ranging from course grained alignment of paragraphs and sentences or simply chunks of text to fine grained alignments of phrases and words. The shallower the granularity of the text units to be aligned the more difficult the task. Yet, smaller units remain to be of prime importance for many applications translation data can be used for.

2.3.1 Paragraph and sentence alignment

For a word alignment system the texts are first segmented into smaller units which are themselves aligned. This helps to determine the similarity of the distribution of terms across these segments. Texts can be segmented into paragraphs, sentences or simply chunks of text with a certain byte length. A word alignment system would naturally be more precise with segments of smaller size. This is basically because the smaller the segment size the fewer the number of words in the segments and hence fewer possibilities of

alignment. Of course these segments should be correctly aligned with their counter part. Otherwise it would result in faulty estimation of parameters. Given the segments of texts, distribution of words across these segments is compared.

Paragraphs are often aligned sequentially, i.e. first paragraph to first paragraph and so on. This might not be always the case, however. Since there could be splitting and merging and even there could be different orders of translating paragraphs. Other problems also come due to deletions and insertions. Paragraph markers sometimes may not be present in the corpus especially in scanned documents which makes the task even more complicated. For structured text with clear paragraph boundaries, the postion and length of paragraphs are the basic criteria of alignment. The use of cognates and collocations is also used to recognise translation paragraphs.

Aligned paragraphs are further segmented into sentences. Sentence alignment is not trivial because translators do not always translate one sentence in the input into one sentence in the output. Another problem is that of crossing dependencies, where the order of sentences are changed in the translation. There are several sentence alignment algorithms [Gale and Church, 1991, 1994, Brown et al., 1991b, Kay and Röscheisen, 1988, 1993] being some of them. The main approaches of sentence alignment in the state of the art studies are:

1. *Length-Based Approaches*: Sentence length methods are based on the intuition that the length of a translated sentence is likely to be similar to that of the source sentence. Sentence length (measured in characters

or words) are used to evaluate how likely an alignment of some number of sentences in L1 is with some number of sentences in L2. Brown, Lai and Mercer [Brown et al., 1991b] used word count as the sentence length, whereas Gale and Church [Gale and Church, 1991] used character count. Brown, Lai and Mercer assumed prior alignment of paragraphs. The method performs well on related languages and clean text where noisy optical character recognition (OCR) or unknown markup conventions are not used.

2. *Offset Alignment by Signal Processing Techniques*: These approaches do not attempt to align beads of sentences but rather just to align position offsets in the two parallel texts [Fung and McKeown, 1994]. They induce an alignment by using cognates (words that are similar across languages) at the level of character sequences. The method consists of building a dot-plot, i.e., the source and translated text are concatenated and then a square graph is made with this text on both axes. A dot is placed at (x,y) when there is a match. Signal processing methods are then used to compress the resulting plot. The interesting part in a dot-plot are called the *bitext maps*. These maps show the correspondence between the two languages.

   In the bitext maps, roughly straight diagonals corresponding to cognates can be found. A heuristic search along this diagonal provides an alignment in terms of offsets in the two texts. The algorithm works without having found sentence boundaries. For each word, a signal is produced, as an arrival vector of integer numbers giving the number of

words between each occurrence of the word at hand.

3. *Lexical Methods*: Use lexical information to align beads of sentences [Kay, 1991, Kay and Röscheisen, 1993]. By assuming that the first and last sentences of the texts align (the initial anchors), find pairs of source and target sentences which contain many possible lexical correspondences.

The most reliable of these pairs are used to induce a set of partial alignments which will be part of the final result. The best alignment is the one that maximises the likelihood of generating the corpus given the translation model.

2.3.2 Word alignment

There are several word-alignment strategies devised by computational linguists for major languages such as English and French [Dagan et al., 1993, Simard et al., 1992, Dagan and Church, 1994, Gale and Church, 1994, Melamed, 2000, Kay and Röscheisen, 1993]. Considerable effort has also been made to align English-Chinese translation texts [Fung and Church, 1994, Wu and Xia, 1994]. The task of aligning words has been dominated mostly by statistical approaches based on the distribution of words in text. The assumption behind using the statistics of words as an indication of possible association between terms is hinged on the assumption that *translation words are comparably distributed in parallel texts.* In practice, word alignment is much more difficult than sentence alignment. Main algorithms are discussed below.

2.4 Statistical approach

In general statistical alignment algorithms try to use word distribution information in text to find some relationship between possible translations. There are different word distribution measuring metrics. Each of them use one or more of the information on the frequency of words, where in the text terms exist, and how far repeated appearances of a term are. To determine these values texts should first go into a serious of segmenting, preprocessing and archiving activities. Then parameter estimation and comparing similarities are involved in determining the distribution relations.

a. Preprocessing

For use in statistical methods, texts need to be preprocessed to remove punctuation marks and numbers from texts. Some preprocessing efforts may also include semi-automatic removal of known deletions and insertions. Uppercase characters are also converted to lowercase.

b. Measuring term distribution similarities

There are different methods of measuring similarities in distribution. The most prominent of these are conditional probability measures, mutual information measure and Dice' similarity measures.

i. Conditional probability measures

The conditional probability of a target term t given a source term s is measured using Bayes theorem:

$P(t|s) = P(t, s)/P(s)$

Table 2.1: Source-target term distribution

	S_1	S_2	S_3	S_4	S_5
s	0	1	1	0	0
t	0	1	1	0	1

If s and t are distributed in two texts divided into five segments each as in Table 2.1. 0 indicates the absence of a term in a segment and 1 the appearance. $P(t|s)$ would be $2/2 = 1.0$. One needs to notice that $P(t|s)$ is different from $P(s|t)$ which is $2/3 = 0.67$. From the values one could say the presence of s indicates the presence of t but not the other way round. In a sense the values of the conditional probabilities should somehow be combined to indicate mutual similarities.

ii. Mutual information measure

Formally, the mutual information of two distributions S and T can be defined as:

$$I(S;T) = -\sum_{i=0}^{n}\sum_{i=0}^{n} p(s,t)log(p(s,t)/p(s)p(t))$$

where $t \in T$, $s \in S$, $p(s,t)$ is the joint probability distribution functions of S and T, and $p(s)$ and $p(t)$ are the marginal probability distribution functions of S and T respectively. The distributions in Table 2.1 can be represented using joint probability distributions as in Table 2.2

$p(s,t) = 2/5$, $p(s) = 2/5$, $p(t) = 3/5$

$I(S;T) = -p(s,t)\ log\ (p(s,t)/p(s)p(t) = 0.63)$ bit

Table 2.2: Joint probability distribution

	s	-
t	2	3
-	2	2

The value of I goes closer to 0 for distributions not related and for the more similar to 1.

iii. Dice' Similarity Coefficient

Dice' coefficient is a term similarity measure whereby the similarity measure is defined as twice the number of common occurrences of terms divided by the total number of terms in both tested entities. The coefficient result of 1 indicates identical vectors; whereas a 0 indicates orthogonal vectors. Taking the distributions in Table 2.1, the similarity measure would be 2/5. If there had been a different distribution such that a word can appear more than one time in a sentence, which is not uncommon the score by all of the above methods will be still the same.

c. Matching terms

The first work on statistical alignment is an attempt made by Kay and Röscheisen to align words in English-German translation texts [Kay and Röscheisen, 1988, 1993]. Kay and Röschesen, came up with a strategy of mining translation units relying merely on evidence they get from the corpus itself without using external knowledge. They start by roughly aligning

sentences in the texts, taking points in the diagonal of a Cartesian plane consisting of the sentences of the source and target documents in the X and Y axis. An initial set of candidate word alignments are then extracted from possibly aligned sentences. Association between words is computed using Dice's similarity coefficient [Rijsbergen, 1979] obtained from vectors of words with binary values indicating presence or absence of the terms in the aligned sentences.

$$Similarity = \frac{2c}{N_A(v) + N_B(w)}$$

where c is the number of corresponding positions, and $N_A(v)$ and $N_B(w)$ are the number of occurrences of the word v and w in the source and target documents respectively. Multiple appearance of a word in a sentence does not affect the numerator c since binary weights are used. The distribution of a word in the source language is compared with the distribution of every word in the target language in search for the word with the highest similarity.

One of the challenges statistical methods are short of solving is the presence of variants of the same word for different grammatical functions, while still they maintain the semantics. Kay and Röscheisen, still depended on corpus evidence to resolve this problem. They wrote an algorithm that tries to find a breaking point that divides a word into two components, each of which might be an affix or a stem. This point is determined by how many times a stem or an affix exist in text as free or bound morphemes. After identifying the two parts of the word, the longest part is used as a canonical form of the word. If both parts are of equal size then the prefix (the first

part of the word) is considered the canonical form. This initial work opened also other doors for new approaches.

Shortly after, another well known algorithm for text alignment,K-vec, was developed by [Fung and Church, 1994]. K-vec is designed to avoid ambiguities created in identifying sentence and paragraph boundaries which my be difficult to obtain for certain data such as scanned documents. Therefore, K-vec, instead of segmenting text into paragraphs or sentences it divides the two texts into K pieces with equal size measured in bytes. The distribution of each word in the pieces is then denoted by a K-dimensional binary vector. The joint probability distribution of each word in a source language and a word in the target language is represented in a contingency table as in Table 2.3,

Table 2.3: Contingency table

	French	-
English	a	b
-	c	d

Where a is the number of pieces where both the English and French word are found, b the number of pieces where just the English word is found, c the number of pieces where the French word is found, and d the number of pieces where neither word is found. From the contingency table mutual information and t-scores are used as measurements on all pairs of K-vecs,

$$I(x,y) = log_2 \frac{P(x,y)}{P(x)P(y)}$$

where $P(x,y)$ is the probability that the words x and y occur in corresponding pieces, and $P(x)$ and $P(y)$ are the probabilities that x and y respectively occur in the text. The probabilities are estimated by using absolute frequency numbers. $P(x,y)$ is $freq(x,y)/K$, $P(x)$ and $P(y)$ are $freq(x)/K$ and $freq(y)/K$, where $freq(x,y)$ is the frequency that x and y occur together and $freq(x)$ and $freq(y)$ are the number of occurrences of x and y. The t-score is used to filter out insignificant relations. Thus, K-vec is based on the assumption that if two words are translations of each other, they are more likely to occur in the same segments than two words which are not. Except for the scoring scheme they used, if the size of a piece in a text corresponds to a sentence, K-vec would be similar to the approach by Kay and Röscheisen. This approach was used for historically related languages French-English of the Canadian Hansard in which case the authors claim high performance. But was not as good for distant languages such as Japanese/English or Chinese/English.

A modified version of K-vec was later proposed to be used for less related languages belonging to different language groups [Fung and McKeown, 1994]. This version is called Dynamic K-vec or simply DK-vec. DK-vec captures recency information in addition to frequency and position of words as features for matching word pairs. From a vector of a term with a frequency *f*, a recency vector with *f-1* entries can be generated. For example for a word w with a frequency 5 occurring at positions (8, 20, 31, 75, 160), the recency vector

would be of length 4 (12, 11, 44, 95). The recency vectors of translation words may be of different length. Therefore, a matching technique for comparing vectors of variable length (Dynamic Time Warping) is used.

[Choueka et al., 2000] used DK-vec to align Hebrew-English translation text. They used DK-vec to generate a rough initial alignment which is given as an input to word_align, an extension of Model 2 in the IBM statistical translation model [Brown et al., 1990].

One of the well known translation system based on parameter estimation obtained from parallel corpora is the IBM statistical translation model [Brown et al., 1990, 1993]. They developed their model using expectation maximisation (EM) algorithm to find the most likely source translation sentence given a target sentence. The parameters of their translation model are a set of translation probabilities $Pr(f|e)$, one for each element f of the French vocabulary and each member e of the English vocabulary; a set of fertility probabilities $Pr(n|e)$ for each English word e describing how many French words it produces during alignment; and a set of distortion probabilities $Pr(i|j,l)$ for each target prosition i, source position j, and target sentence length l showing the relative positions of French and English words in the source and target sentences. Thus, the probability of alignment for two terms f and e is calculated as a product of the translation probability, the fertility probability and the distortion probability.

[Dagan and Church, 1994] developed a tool, *Termight*, for helping professional translators and terminologists identify technical terms and their translations. The tool makes use of part-of-speech tagging and word_align

to extract candidate terms and their translations. They exclude stop-words in standard stop-wordlist from their system. Termight provides a list of candidate translations sorted descending based on frequency of occurrence. It also provides a concordance for each candidate. The user then is allowed to choose the correct translation among the candidates by crosschecking to the concordance whenever they need. They report evaluation results showing the bilingual component of termight in translating a glossary of 192 terms found in the English and German versions of a technical manual. The correct answer was the first candidate in 40% of the cases and the second choice in 7% of the cases. For the remaining 53% of the terms, the correct answer was always somewhere in the concordance.

An exploration on Amharic by [Alemu et al., 2004] deals with an attempt to extract noun translations from the bible. Yet, nouns are relatively minimally inflected and not a problem to align in Amharic, specially when the bible is the data source.

2.5 Linguistic alignment methods

Linguistic alignment methods are basically highly dependent on linguistic knowledge such as cognates [Church, 1993, McEnery and Oakes, 1996] and external knowledge base such as machine readable dictionaries or glossaries as in SIMR [Melamed, 2001]. These methods perform highly for related languages that share the same vocabulary to a large extent. In cases where cognates are insignificant subset of the languages considered, they certainly

would fail. [Klavans and Tzoukermann, 1996] combine what they call structured but incomplete information in dictionaries to unstructured but more complete lexical information available in English-French bilingual corpora. They deal only on verbs that describe the action of movement. They are supported with linguistic information of parts-of-speech and verb types. Kupiec [Kupiec, 1993] also aligned noun phrases in English and French bilingual corpus, using part-of-speech taggers for each half of the texts.

- **Morphology**: Morphological analysers help to distinguish between lexical components of words which are accountable for the semantics of the words and grammatical words. During text alignment this helps to align grammatical components and lexical components separately. Hence the use of effective morphological analysers can particularly be useful to assist statistical alignment systems. Specially, this is true for languages that are highly inflected. [Choueka et al., 2000] have used a fullfledged morphological analyser for Hebrew when aligning Hebrew-English translation texts. [Alemu et al., 2004] have also used a stemmer when aligning English-Amharic nouns.

- **Part of Speech**: Part of speech information is important during text alignment and later for disambiguating ambiguous translations. In translation sentences if a word with a certain part of speech is recognised, then one can project that the words with other part of speech may not be its translation.

- **Syntax**: One approach of aligning words/phrases of a source sentence

to that of the target is by projecting the parse tree of one to the other. When such tools are found in both languages successful alignments can be obtained to a large extent [Yamada and Knight, 2001, Niessen and Ney, 2004, Smith and Eisner, 2006, Zhang and Gildea, 2004, Gildea, 2003].

- **Dictionaries**: Dictionary based alignment algorithms try to recognise words in sentences as given in dictionaries [Klavans and Tzoukermann, 1996, Melamed, 2000, 2001]. Such alignment methods may not necessarily depend on complete dictionaries but with some entries that would allow recognition of anchor words which would reduce the number of remaining alignments to be made. Tree-based approaches to alignment, model translation as a sequence of probabilistic operations transforming the syntactic parse tree of a sentence in one language into that of the other [Zhang and Gildea, 2004, Gildea, 2003]. The trees may be learned directly from parallel corpora [Wu, 1997], or provided by a parser trained on hand-annotated treebanks [Yamada and Knight, 2001].

- **Cognates**: Cognates in historically related languages have been used to align texts [Melamed, 2000, 2001]. This alignment method has produced good results on highly related languages. [Melamed, 2001] argues if orthographic cognates cannot be recognised on languages that do have different writing system phonetic cognates can be used. But again how exhaustive these cognates are in comparison with word types

in a language is a question that may not have a positive answer.

More recent studies use hybrid methods [de Gispert et al., 2006, Niessen and Ney, 2000, Yamada and Knight, 2001, Och and Ney, 2000, Gildea, 2003]. [de Gispert et al., 2006] presents a wide range of statistical word alignment experiments incorporating morphosyntactic information. By means of parallel corpus transformations according to information of POS-tagging, lemmatisation or stemming, they explore which linguistic information helps improve alignment error rates. They reported improvements due to introducing morphosyntactic information are bigger in case of data scarcity, but significant improvement is also achieved in a large data task, meaning that certain linguistic knowledge is relevant even in situations of large data availability.

2.6 Filtering

Alignment algorithms produce alignments of words with their first most likely translation or the first N-best translations. Such lists are not all true translations. On the other hand using a lexicon that has also false translations within produces erroneous use by applications that make use of the lexicon. Apparently, to reduce these false alignments from the result set different filtering mechanisms are used. Hence, filters of removing unlikely translations are made by introducing filters at different levels:

- **Score filter**: Alignment systems produce the possible translations in ranked order from which those ranked higher are taken as the best

translations. But the absolute value of the scores may still be low. If the evaluation metric gives scores of 1.0 as the highest score and 0.0 as the lowest score. obviously low scoring translations are not likely to be true translations. Hence one way of filtering the translations would be to remove those candidates with low score from the translation list.

- **Frequency filter**: Particularly statistical methods that rely on the statistics of words require as many instances for better precision. Those words with low frequency, even if they may give high score, it could be a matter of coincidence and not necessarily because they are true translations. To avoid such cases introducing a frequency filter that removes words of low frequency from the translation list is necessary.

- **Part of speech filter**: The part of speech filter removes every translation candidate with different parts of speech in the source and the target language. Here the usage of different tags for different languages is something one should note.

- **Machine-readable bilingual dictionary filter**: Available, machine-readable bilingual dictionaries are used as a lookup, where priority is given to the alternative that exists in the dictionary.

2.7 Evaluation

There are growing number of research in the area and various results have been reported in the literature. However, these results are often evaluated

on different corpora that makes it difficult to compare them [Vèronis and Langlais, 2000] (See also ARCADE: http://aune.lpl.univ-aix.fr/projects/arcade/index-en.html). Precise methods of evaluating alignment systems is invaluable for developing better systems and also for users who would like to use efficient methods for the applications they are interested in.

Alignment systems can be evaluated to compare:

- The performance of different systems on the same reference corpus of two given languages
- The performance of systems across different various language pairs.

Of course even for the same language pairs, alignment systems do not work equally well on all kinds of documents. Hence, corpora that are built on principles of balance and representativeness are of crucial importance. On the other hand the availability and accessibility of parallel texts is often limited to certain domains, at least for many languages (See the Linguistic Data Consortium page for some parallel corpora: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC95T20). The Canadian Hansard (http://en.wikipedia.org/wiki/Hansard) and Europarl [Koehn, nd.] are two of the most used such corpora. The Hansard Corpus consists of parallel texts in English and Canadian French, drawn from official records of the proceedings of the Canadian Parliament. The content is therefore limited to legislative discourse. It consists of over 20 million words per language. Europarl is extracted from the proceedings of the European Parliament. It includes versions in 11 European languages. It has a size of

up to 28 million words per language. These corpora are always updated and their size increases every time.

Aligned text units are compared to manually annotated alignments. Precision and recall measures are then computed. Precision is simply the ratio of the number of correctly aligned units to the number of all aligned units by the system and recall is a measure of correctly aligned units to the total number of correctly aligned units in the corpus. To resolve the tradeoff between recall and precision, measures such as F-measure (The weighted harmonic mean of precision and recall) $= 2 * (Precision * Recall)/(Precision + Recall)$ are used.

In this project work the principles of existing statistical alignment methods are used to get an initial alignment based on which new methods of alignment and enhancement are projected.

Part I

Model I

Chapter 3

Global Alignment

This chapter describes a simple approach of statistical translation modelling for bilingual lexicon acquisition. The model is tested on Amharic-English parallel corpora. The translation lexicon contains matches of Amharic lexemes to weekly inflected English words. Purely statistical measures of term distribution are used as the basis for finding correlations between terms. For low frequency terms a two step procedure of: first a rough alignment; and then an automatic filtering to sift the output and improve the precision is made.

A brief exemplary account of the grammatical characteristics of Amharic with relevance to corpus-based lexical acquisition is presented in Section 3.1. In Section 3.1.3, the orthography of Amharic is introduced. In Section 3.2, methodological aspects of how the problem is approached are discussed. Evaluation results are reported in Section 3.2.3.

3.1 Data: Amharic - English parallel texts

One of the goals of this study is to develop a system that works well for languages that are disparate and where there are scarcities of data and tools. Therefore, the system is evaluated on Amharic-English translation texts. Eventhough, a detailed description of Amharic morphology is presented in Chapter 5, A basic comparisons of Amharic and English texts is provided here for the reader to understand this first part of alignment Model I and the outputs it produces.

3.1.1 Morphology

Amharic and English differ substantially in their morphology, syntax and the writing system they use. As a result various methods of alignment that work for other languages do not apply for them. A description of the grammar of Amharic that suffices the relevance to text alignment is subsequently presented.

Amharic is a Semitic language that has a complex morphology in which words are consisted of discontinuous consonantal roots with vowel intercalation [Amsalu and Gibbon, 2005a, Fissaha and Haller, 2003, Bayou, 2000], an inherent Semitic property. Articles, prepositions, conjunctions and personal pronouns are often inflectional components of other parts of speech and can only seldom occur as disengaged morphemes. Apparently, sentences in Amharic are often short in terms of the number words they are consisted of. For the reader to assimilate the flavour of the problem, just picking the

first sentence in the bible in Amharic and English, a ratio of 1:2 words is obtained(Figure 3.1).

በመጀመሪያ አግዚአብሔር ሰማያትንና ምድርን ፈጠረ

In the begining God created the heavens and the earth

Figure 3.1: Translation sentences

This is a common case as far as the two languages are concerned. The texts that are used in the experiment presented in this chapter have a ratio of 22179 : 36733, which is approximately 1 Amharic word to 1.7 English words. But considering morphemic substratum a different result is observed. In Figure 3.2, a projection at nearly morpheme level is presented.

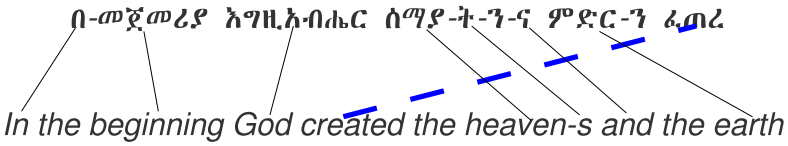


Figure 3.2: Morphemic alignment.

The definite article '*the*' that occurs three times in the English sentence in Figure 3.2 is in all cases implicit in the Amharic translation. Hence, there are floating words in the English side that are not aligned. Definiteness in Amharic is not necessarily explicitly represented. It is often left to be understood contextually. When it is explicit the definite article is realized as a suffix and rarely the indefinite article is expressed with a number coming before the noun such as '*and säw*', literally it means '*one man*', parallel to

the English '*a man*'. The object marker'ን' in Amharic also does not exist in English. The conjunction *and* is also a suffix in Amharic. For now a detailed account of Amharic morphology is not given; better treatments are given in Chapter 5 and other studies by [Yimam, 1994, 1999, Bender and Fulas, 1978, Berhane, 1992, Dawkins, 1960, Amare, 1997, Markos, 1991, Amsalu and Gibbon, 2005a]. Chapter 5 also gives a general description of Amharic structure.

3.1.2 Syntax

Syntactically, Amharic is an SOV language. It does not have free order as in other Semitic languages due to the Cushitic influence on the language. The generalisation given by [Choueka et al., 2000] about the free word order for Semitic languages does not hold for Amharic. Taking their own example,

The boy ***ate*** *the apple* (English)

the correct representation in Amharic is:

The boy the apple ***ate***

This forbids a linear alignment of Amharic words with their English equivalents which are revealed in SVO order. The broken line in Figure 3.2 shows a cross-over alignment that accommodates this discord in syntax. In a two dimensional Cartesian plane of alignments between source and target texts one does not expect a linear path, rather it would be skewed at the position of inversion of the verb and object. See the chart in Figure 3.3 for the portray of the mapping of the example sentences.

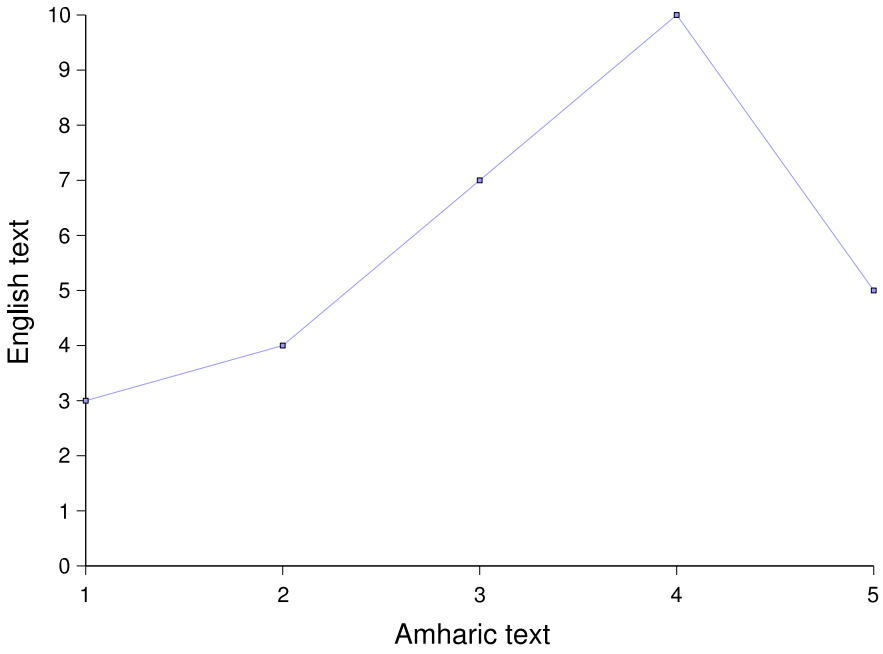


Figure 3.3: Non-linear alignment.

3.1.3 Orthography

Amharic uses a syllabary script called *Fidel*, with graphemes denoting consonants with an inherent following vowel, which are consistently modified to indicate other vowels or, in some cases, the lack of a vowel. There are no different representations of upper and lower cases, hence there is no special marking of the beginning of a sentence and first letter of names or acronyms. Words are separated by white space. The streams of characters, however, are written left-to-right deviating from its relatives Hebrew and Arabic.

Amharic and English do not share many words such as, say, English and German do, for us to use cognate alignments even with phonetic alignment; though it might be possible for scientific words, technical words and names

of places in very specified texts. Possibilities of identifying nouns based on their spelling like in German nouns or English common nouns is also not possible due to the absence of special markings for them.

3.2 Alignment method

The alignment task is began by first trying to generate an initial lexicon of rough 1:1 alignment which is then improved by subsequent modules. To produce such aligned lexicon statistical measures of frequency are used for parameter estimation.

Statistical alignment methods are in general based on the assumption that translation terms in translation texts are distributed comparably. For example for a word s that occurs 30 times in a source document, its translation t is not expected to occur 3 times in the target document. Neither does one expect the translation of s which occurs only in the first half of the source document to be t occurring on the second half of the target document. Hence, how often and where in the document terms occur is an important measure of the distribution of the terms.

3.2.1 Capturing term distribution data

The distribution of a term is simply a measure of how frequently and where in the document it occurs. Texts are often divided into smaller segments inorder to decrease the amount of search space and consequently have limited options. In the case of this work the small segments are sentences. Three

parameters are used to describe the distribution of each term in the segments and in the texts as a whole:

1. *Global-frequency*: Frequency of occurrence in the corpus;

2. *Local-frequency*: Frequency of occurrence in a segment; and

3. *Placement*: Position of occurrence in the corpus (i.e. segment ID).

Each of the translation texts is mapped into a two dimensional matrix with sentences as columns and words as rows. A document that consists of five sentences could for example be mapped as in Table 3.1,

Table 3.1: Document mapping into a matrix.

	S_1	S_2	S_3	S_4	S_5
$Word_1$	0	2	1	0	0
$Word_2$	1	0	0	3	0
$Word_3$	0	0	0	0	1
...	...	...	...	...	...

The entries in the matrix stand for local frequencies of the words in the corresponding sentences. Thus $Word_1$ occurs in sentence 2 (S_2) twice and one time in sentences 3 (S_3). One can also observe that each word in a document is mapped into a weighted vector, where the weights are the local frequencies. The idea of weighted vectors as opposed to binary vectors used by [Kay and Röscheisen, 1988, 1993, Fung and Church, 1994] is introduced

because it contradistinguishes terms that appear only once in sentence from those that appear multiple times. Function words that are likely to occur more than once in sentences, particularly in English sentences are addressed easily this way. The target document is also mapped into such a matrix.

Hence, there are an $n\ x\ s$ and an $m\ x\ s$ matrices; where n and m are the number of unique terms in Amharic and English respectively and s is the number of segments in either of the texts. The values in the matrix are local frequencies. Therefore, each word is a weighted vector of its distribution; where the weight is its local frequency in the respective segment.

3.2.2 Comparing distributions

In this module a global search for the best translation is made. The idea is that the distribution of each term in the source document is compared with the distribution of all the terms in the target document. The target word that gives the highest similarity is then taken as the possible translation. Second best or third best translations are not considered at this point.

If D_a is the set of distributions in the Amharic text and D_e the set of distributions in the English text and let T_a be the set of terms in Amharic text and T_e be the set of terms in English text, then if an Amharic word $Word_j \in T_a$ has a distribution $D_j \in D_a$ and an English term $Word_k \in T_e$ has a distribution $D_k \in D_e$, then the score of the translation candidates $Word_j$ and $Word_k$ is a measure of the degree of similarity between the distributions D_j and D_k.

To measure similarities there needs to be a metric that gives a figurative

value of similarity that helps us to compare the different candidates. For this a novel scheme that gives categorical scores for each distinct pair of distributions and favours those that are distributed similarly is formularised. The scoring scheme also handles function words robustly. Basically the scoring scheme takes the minimum of the two vectors compared and computes the sum of the entries on the nominator side. On the denominator side a vector which is a sum of the two vectors is generated. The entries of this vector are then summed. Hence, for an Amharic term vector $Word_j$ and English term vector $Word_k$,

$$Score_{(j,k)} = \frac{2 \cdot \sum_{i=0}^{n} (Word_j \wedge Word_k)_i}{\sum_{i=0}^{n} (Word_j + Word_k)_i}$$

where $_i$ denotes the i^{th} entry of a vector and n stands for the number of segments.

If for example $Word_j = (0, 2, 1, 0, 0)$ and $Word_k = (0, 1, 1, 0, 1)$. Then,

$$Word_j \wedge Word_k = (0, 2, 1, 0, 0) \wedge (0, 1, 1, 0, 1) = (0, 1, 1, 0, 0)$$

,

$$Word_j + Word_k = (0, 2, 1, 0, 0) + (0, 1, 1, 0, 1) = (0, 3, 2, 0, 1)$$

$$\sum_{i=0}^{n} (Word_j \wedge Word_k)_i = 2$$

$$\sum_{i=0}^{n} (Word_j + Word_k)_i = 6$$

$$Score_{(j,k)} = \frac{2 \cdot 2}{6} \approx 0.67$$

If instead there are pairs of $Word_j = (0, 1, 1, 0, 0)$ and $Word_k = (0, 1, 1, 0, 1)$,

$$Word_j \wedge Word_k = (0, 1, 1, 0, 0),$$

$$Word_j + Word_k = (0, 2, 2, 0, 1)$$

$$\sum_{i=0}^{n} (Word_j \wedge Word_k)_i = 2$$

$$\sum_{i=0}^{n} (Word_j + Word_k)_i = 5$$

$$Score_{(j,k)} = \frac{2 \cdot 2}{5} = 0.8$$

again, for $Word_j = (0, 2, 1, 0, 0)$ and $Word_k = (0, 2, 1, 0, 1)$,

$$Word_j \wedge Word_k = (0, 2, 1, 0, 0),$$

$$Word_j + Word_k = (0, 4, 2, 0, 1)$$

$$\sum_{i=0}^{n} (Word_j \wedge Word_k)_i = 3$$

$$\sum_{i=0}^{n} (Word_j + Word_k)_i = 7$$

$$Score_{(j,k)} = \frac{2 \cdot 3}{7} \approx 0.86$$

The constant 2 in the numerator is algebraised to normalise the scores to range between 0.0 (for disjoint vectors) and 1.0 (for identical vectors), which otherwise would have been in the range of 0.0 to 0.5.

The scores are more discriminative than the scores obtained using the scoring method of [Kay and Röscheisen, 1993]. The difference lies in that Kay & Röscheisen give binary weights (0 and 1 for absence and presence

respectively) to terms in segments. Binary weights fail to address the difference between words that appear multiple times in a single segment from those that appear only once. Giving non-binary weights is very powerful in dealing with function words. K_vec also gives binary values for terms in the K-pieces. Using binary weights the similarity scores for the terms with distribution $Word_j = (0, 2, 1, 0, 0)$ and $Word_k = (0, 1, 1, 0, 1)$ in equation (3.1) which given binary weights would be represented as $(0, 1, 1, 0, 0)$ and $(0, 1, 1, 0, 1)$ respectively is,

$$Word_j \wedge Word_k = (0, 1, 1, 0, 0) \wedge (0, 1, 1, 0, 1) = (0, 1, 1, 0, 0)$$

,

$$Word_j + Word_k = (0, 1, 1, 0, 0) + (0, 1, 1, 0, 1) = (0, 2, 2, 0, 1)$$

$$\sum_{i=0}^{n} (Word_j \wedge Word_k)_i = 2$$

$$\sum_{i=0}^{n} (Word_j + Word_k)_i = 5$$

$$Score_{(j,k)} = \frac{2 \cdot 2}{5} \approx 0.8$$

Equation (3.2) would have the same score because anyway each word occurs only one time in each sentence. But equation (3.3) would have another result. In binary weights $Word_j = (0, 2, 1, 0, 0)$ and $Word_k = (0, 2, 1, 0, 1)$ would simply be $Word_j = (0, 1, 1, 0, 0)$ and $Word_k = (0, 1, 1, 0, 1)$ respectively. Hence,

$$Word_j \wedge Word_k = (0, 1, 1, 0, 0),$$

$$Word_j + Word_k = (0, 2, 2, 0, 1)$$

$$\sum_{i=0}^{n}(Word_j \wedge Word_k)_i = 2$$

$$\sum_{i=0}^{n}(Word_j + Word_k)_i = 5$$

$$Score_{(j,k)} = \frac{2 \cdot 2}{5} = 0.8$$

Now it is clear that if the distribution of the words gives us some incite of which words could be translations, then words that occur quite a few number of times cannot give us reliable distribution information. To coupe with that often using bigger size of corpora i.e. increase the chances of reoccurrence of rare terms is one of the methods used. Zipf's law, however, works against this assumption. Zipf's law is an empirical law which states that, in a corpus of natural language utterances, the frequency of any word is roughly inversely proportional to its rank in the frequency table. Zipfian distributions are commonly observed, in many kinds of phenomena. The fact that Zipfian distributions arise in randomly-generated texts with no linguistic structure suggests that the law as applied to languages may in part be a statistical artifact. Over fairly wide ranges, and to a fairly good approximation, many natural phenomena obey Zipf's law. Mathematically Zipf's law is stated as

$$f(k; s, N) = \frac{1/k^s}{\sum_{n=1}^{N} 1/n^s}$$

where N is the number of elements, k is their rank, and s is the exponent characterising the distribution. In the classic version of Zipf's law, the exponent s is 1. Zipf's law is most easily observed by scatterplotting the data, with the axes being log(rank order) and log(frequency).

The idea is that even if the data size is increased to increase the chances of reoccuring for low frequency terms, new words are introduced at the same time into the document. On the other hand during evaluation the recall and precision may remain more or less the same with big or small data. But it can be seen from a different perspective; that is language coverage. If a language is a set of words L, and if T is the set of words in L that are part of the lexicon generated and if a term t occurs n times in the text n being a small number, when the data size is increased the chances for n to increase is higher. Which in other words means that the chances for t to be a member of T increases. As T becomes larger then the coverage of the language becomes higher too.

Given Amharic-English texts the goal at this stage is that high frequency Amharic words will be aligned. The high frequency Amharic words are different from high frequency English words (function words) since this word forms rarely occur as free morphemes. Hence these high frequency Amharic words would be lexical words. These lexical words cannot be aligned to high frequency English words because the high frequency function words in English are very frequently distributed and their distribution is in general overall the text. In the end complex lexical words of Amharic and weakly inflected lexical words in English are left. The alignment algorithm does not exclude function words from computation rather the scoring scheme which is discussed in Section 3.2.2 distills them by keeping their scores low. From the Amharic side a significant proportion of the words have a high probability of being included in the lexicon (except for low frequency words), while

in the English side there will be floating words which would in many cases be function words. A demonstration on alignment of the exemplary bitext segment is presented in Figure 3.4.

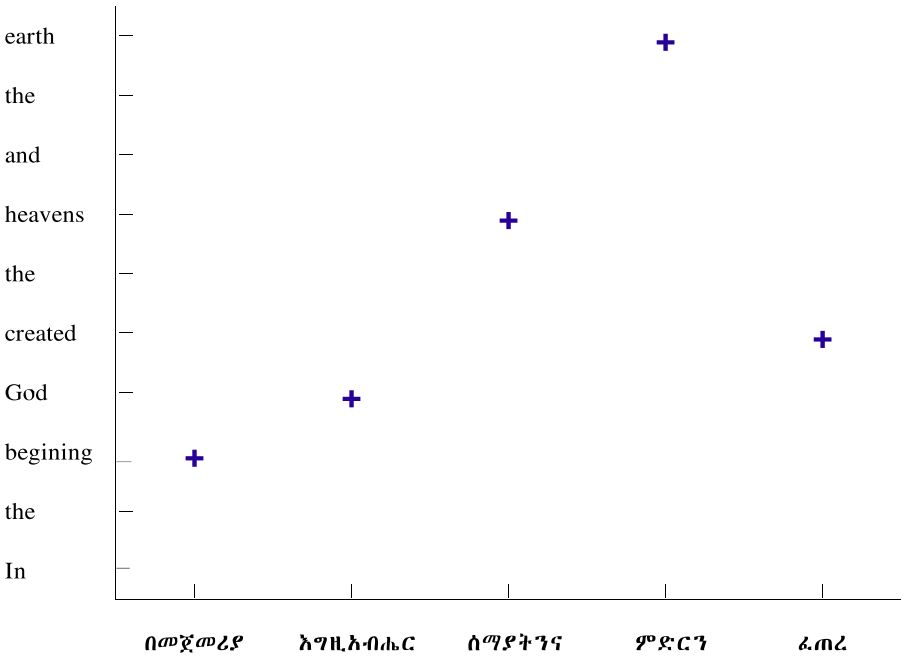


Figure 3.4: Aligned words.

Gaps for non-aligned words and crossing alignments to overcome syntactic differences are extant. Details of the alignment heuristics are discussed in subsequent subsections.

3.2.3 Preprocessing

The statistical translation model developed is evaluated on a dataset of 20,347 Amharic and 36,537 English tokens, which encompass 6867 and 2613 types in Amharic and English respectively. Since the concern is with align-

ing words in sentence aligned translation texts, the bible is taken as a data source. When generating the parallel sentences of the initial corpus, some preprocessing actions were conducted. These preprocessing steps are :

- removal of punctuation marks in both texts
- decapitalisation of the English.
- splitting and combining of sentences

There is no capitalisation in Amharic, so there was no need to perform decapitalisation on the Amharic text. Sentences represented in one sentence in the Amharic and still numbered as more than one verses (for example, verse 4-6 - which means the verses 4, 5 & 6 are contained in one long sentence) have been split. If these combinations were uniform across the two languages, it would have not been a problem, but when it is only a phenomenon of one in one position and the other in another position, it makes the alignment procedure complex. We did not want to deal with this problem at this stage, hence we split them manually. The other preprocessing steps were performed automatically, with the help of a preprocessing software developed as part of this project.

3.2.4 Data structure

Preprocessed text are segmented into sentences and stored a sentence per line in a data structure that makes them easy to access. In our case the structure in Table 3.2 is designed to accommodate all. The Id is an integer

value assigned to identify each sentence in the corpus. This identification is unique for each sentences in each half of the translation texts.

Table 3.2: Data structure of aligned sentences.

$Sent_{Id}$	$Chap_{num}$	$Verse_num$	$Sent.$

The text for each language are stored separately, but the corresponding sentences have identical IDs. Now, to map these documents into a matrix, tasks of tokenisation, computing of type frequencies, sorting etc. are involved. The tokeniser takes each sentences and produces tokens (Figure 3.5). With each word, the Id of the sentence goes with it, since it is useful for generating the document matrix later. From this word list token types and their frequencies are generated (Figure 3.6). Given these two lists of tokens with sentence Ids and type with frequency the document matrix is constructed (Figure 3.7). Using the same procedure the target document is also mapped into a matrix. Every term vector in the source document is then compared with every term vector in the target document. The one that gives the highest score is then taken as the best possible translation.

Figure 3.8 describes a summary of intermediate outputs from the source document. Table a) in Figure 3.8 is the aligned parallel sentences. Table b) has terms and their global frequency, Table c)compresses terms in Table b) to unique terms and their global frequencies. Words in c) are sorted descending by frequency and ascending by word which makes it simple for focusing on high or low frequency words separately during analyses of results. Table d) is a matrix with rows of unique terms in the corpus and columns of unique

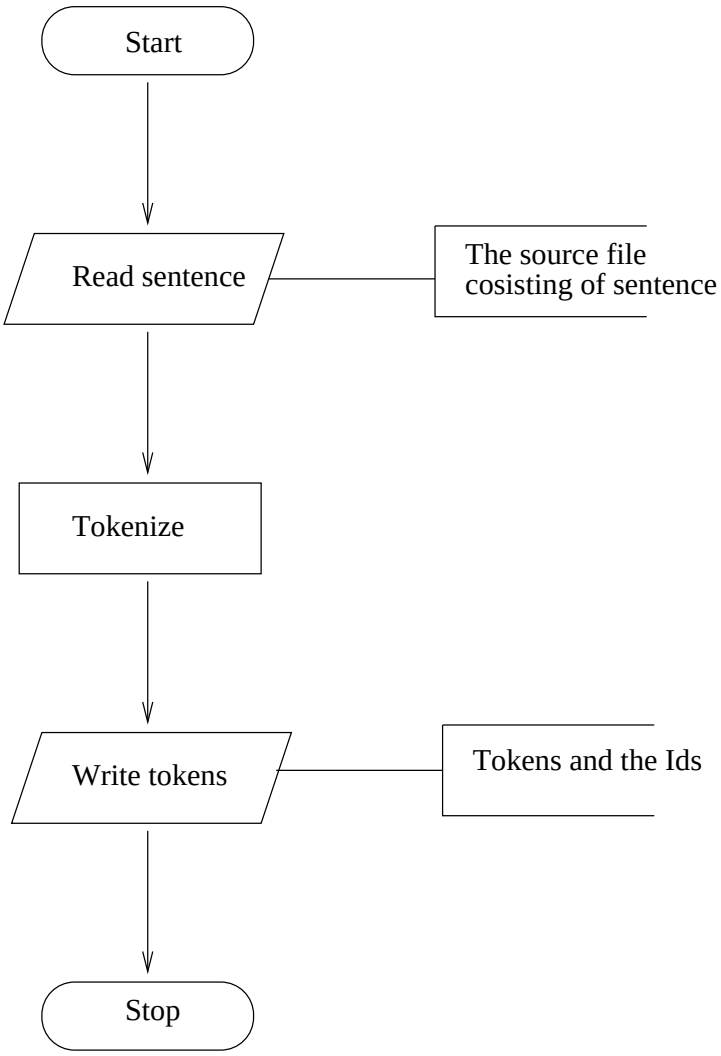


Figure 3.5: Tokeniser.

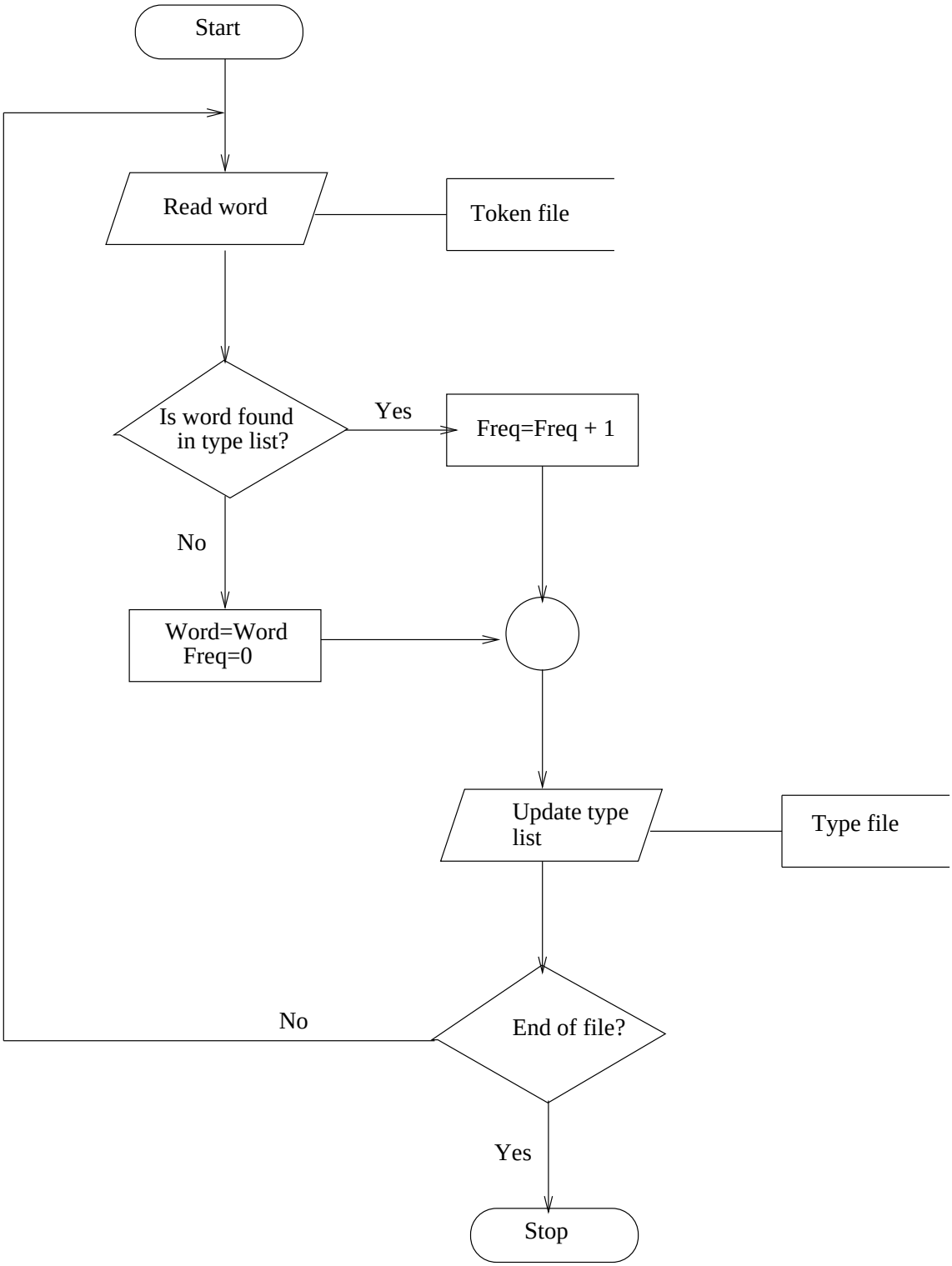


Figure 3.6: Generate type list with frequency.

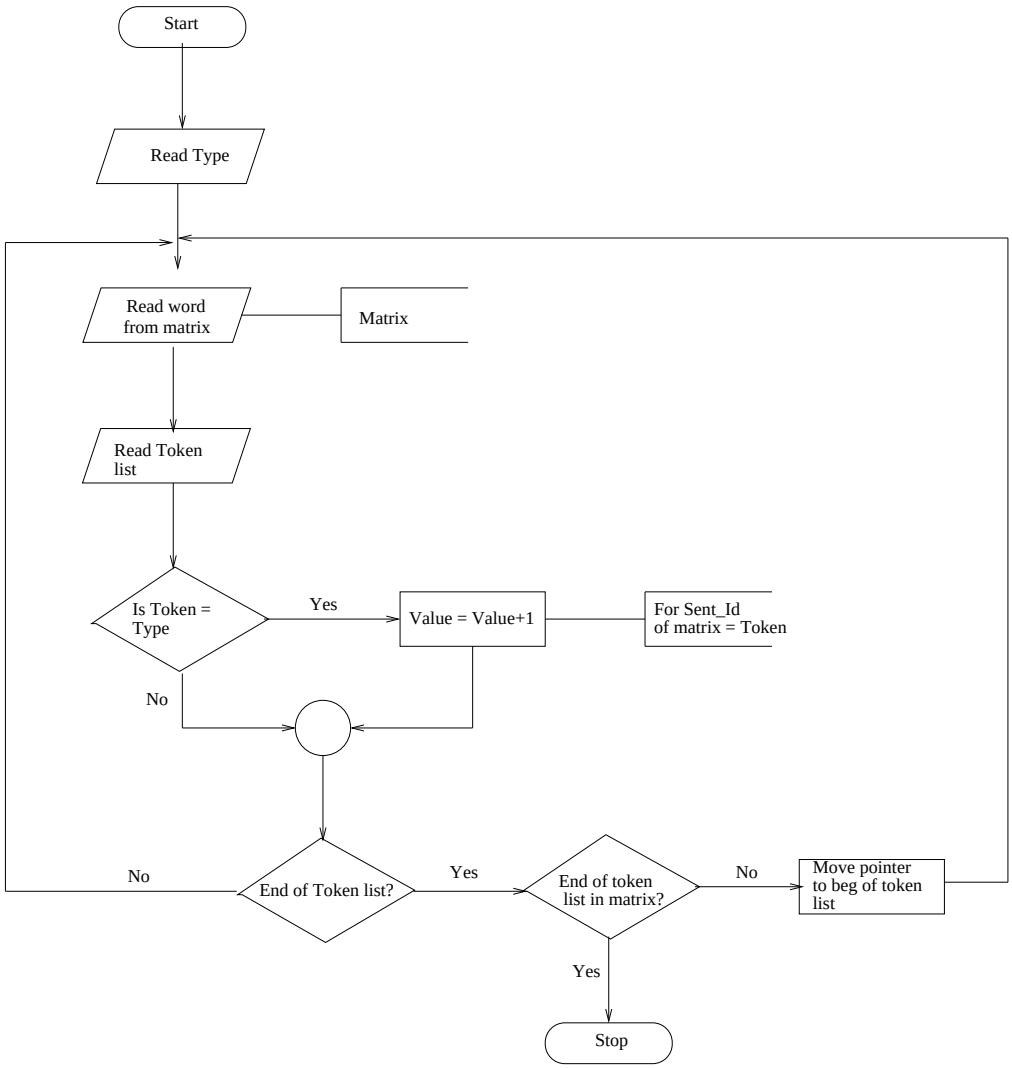


Figure 3.7: Document matrix generator.

sentences in the corpus. The values in the matrix and the local frequencies. Table e), the English counter part is generated with the same procedure as the Amharic.

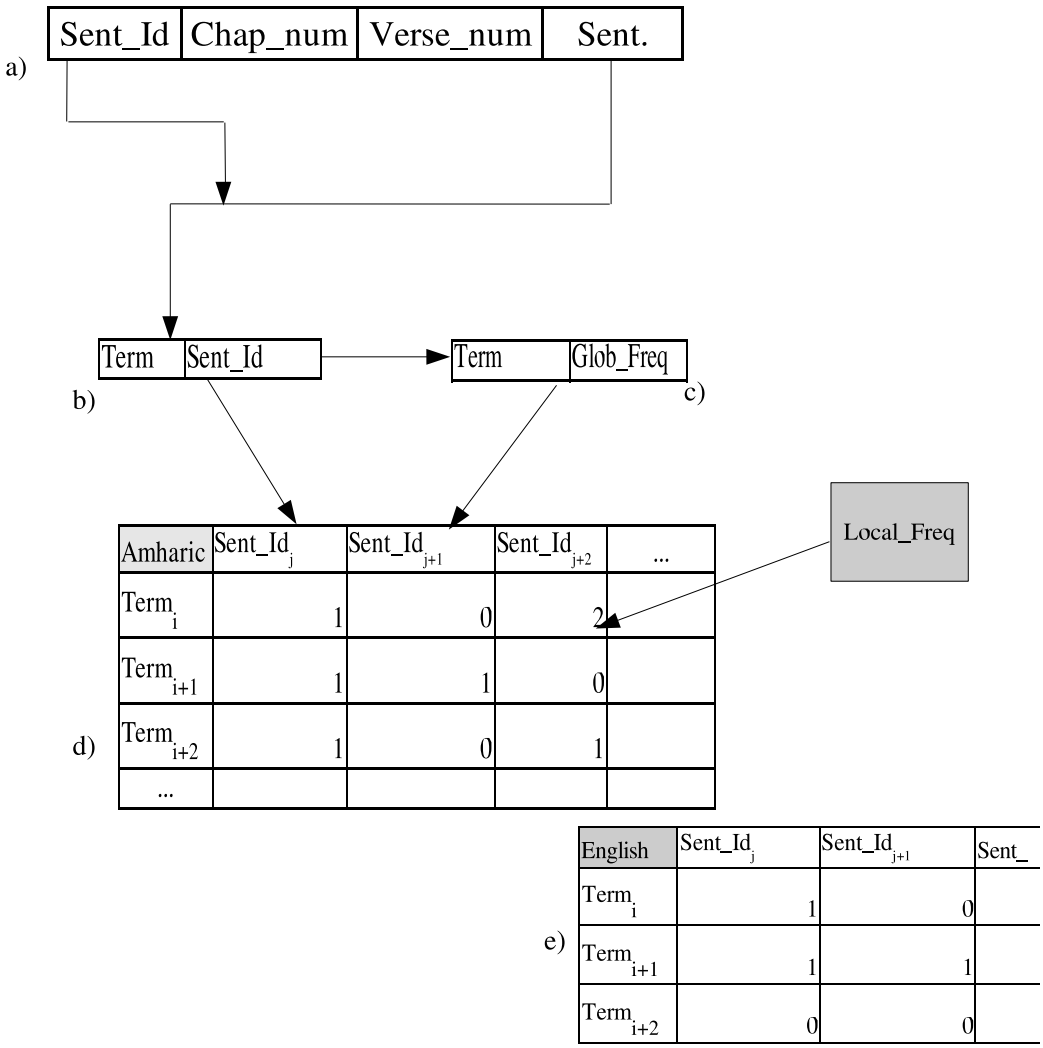


Figure 3.8: Intermediate outputs of distribution analysis.

Another kind of information produced as a byproduct is a translation memory, consisting of entries consisting of translation equivalents in the two languages (Figure 3.9).

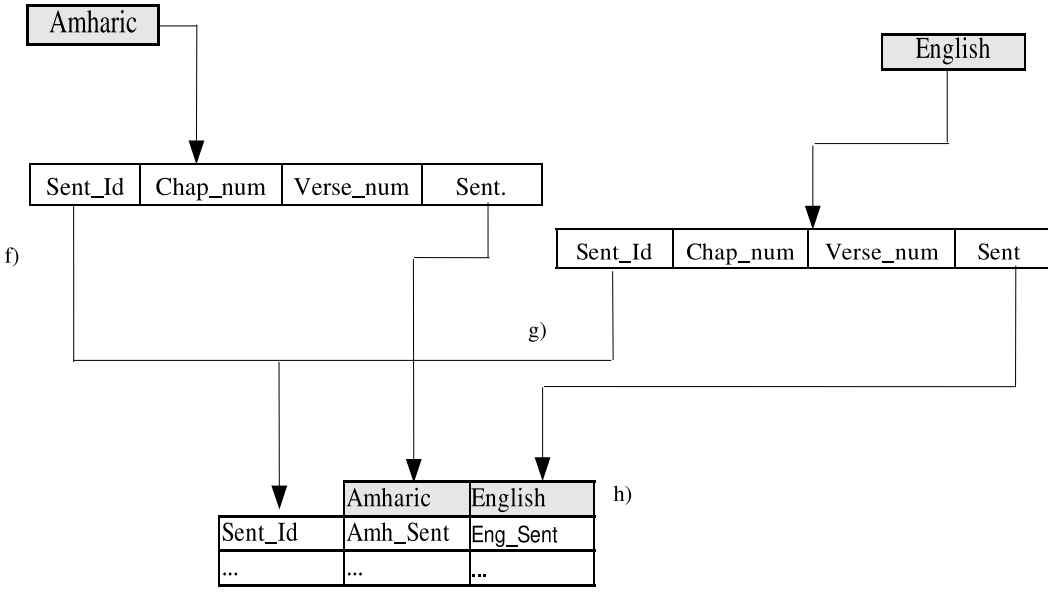


Figure 3.9: Translation memory.

Tables a) and b) in Figure 3.9 are the English sources of corpora. In c) only the sentences and the Id associated with them is extracted.

3.2.5 Aligning the words

Having the distribution information, the next step is to align possible translation equivalents at a word level. The assumption here is that equivalent terms are distrubuted comparable throughout the corpus. Of course their are factors that make this assumption less strong. These could be, the fact that when using such analysis:

- synonyms are considered to be different from each other and
- inflectional variants are also regarded different from each other.

For a language like English, which is rich in vocabulary synonyms are really many. A small system was developed to test if this is true and for an experiment conducted in a small monolingual parallel corpora of the English bible by different translators, a similarity score of 0.64 was obtained by using Dice's similarity coefficient, giving a binary weights to terms. For sets X and Y of words used in monolingual parallel texts, Dice's similarity coefficient may be defined as:

$$s = \frac{2|X \cap Y|}{|X| + |Y|}$$

The procedure goes like this: for each document

1. Clean both texts from punctuation marks
2. Generate type list
3. Count the number of words that are found in both type lists
4. Compute the sum of the number of words in both lists
5. Compute Dice's similarity coefficient

A score of 0.64 shows that the two documents shared 64% of the words. In other words the rest of the words in one document are replaced by other terms which are semantically equivalent to them. The implication of this to the alignment system is that one word in the Amharic document is translated by more than one words (alternative words) in the English. For example the words *construct* and *build* may have been translated into one word in Amharic.

On the other hand for a language such as Amharic where the inflectional variants are really numerous [Amsalu and Gibbon, 2005a]; one cannot expect one variant to occur as often as its English equivalent may.

3.2.6 Thresholds

Obviously, candidates with low score are bad candidates. But the question is, what values of score are low? To determine this cutting point different thresholds of score above which candidates could be true translation were tested on the corpus and the one that gives reasonably good translation pairs is selected. But again not all candidates with high score are true translations. In fact, for a small size of corpus many of the candidates with a score of 1.0 are low frequency words. Hence, to control this a second threshold for frequencies is also set (Figure 3.11).

3.2.7 Filtering mechanism

In statistical methods of alignment, the words that can most likely be correctly aligned are high frequency words. This is because there are many instances of these words that enable them to survive from coincidental co-occurrence with false translations. But for low frequency words, it is highly likely that just by chance they could co-occur with words that are not their equivalents. Specially, when the test is made on a small size of corpora, low frequency words are too many and often coincide with several other low frequency words.

One commonly used method of avoiding such coincidences is to amputate

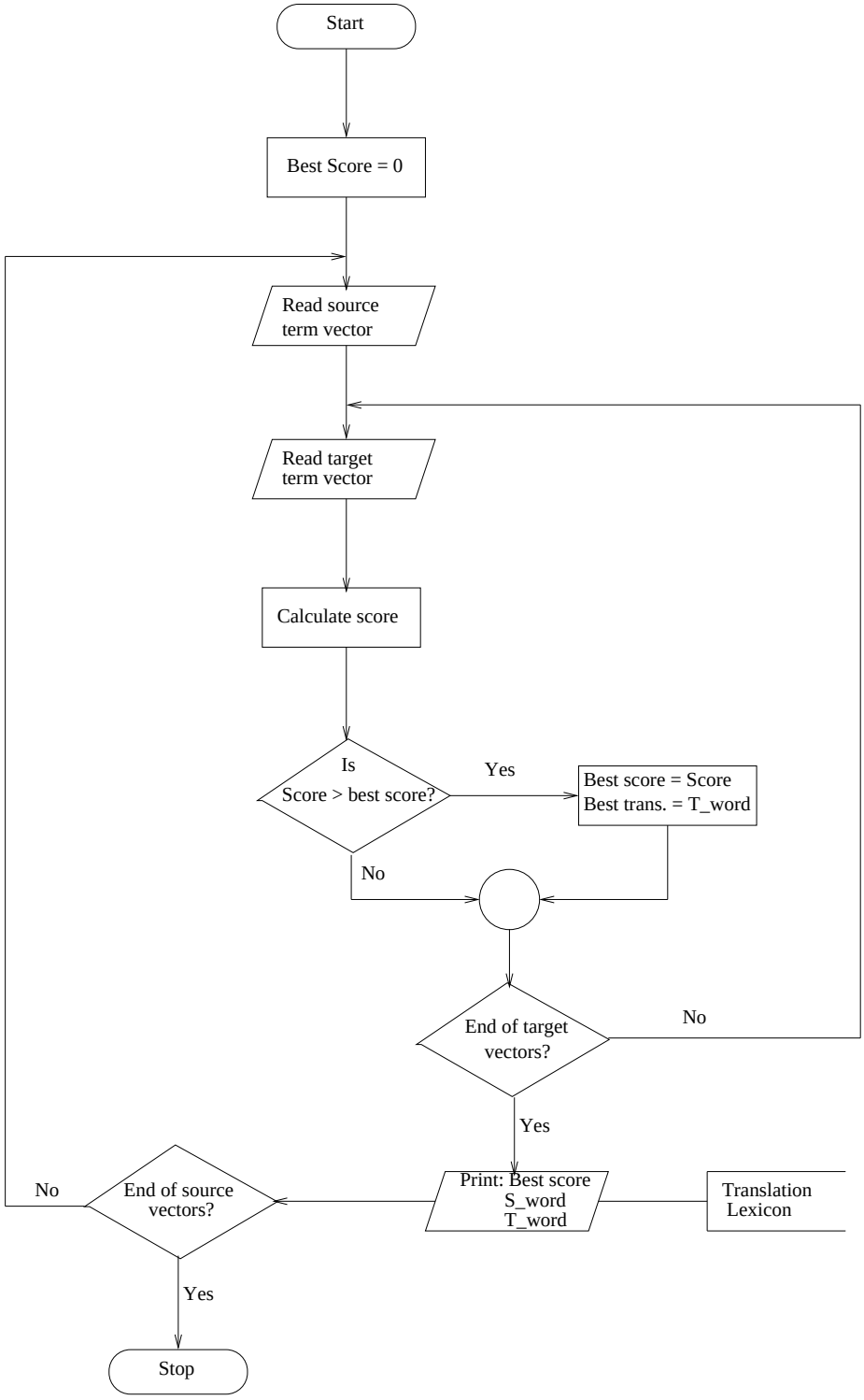


Figure 3.10: Translation lexicon.

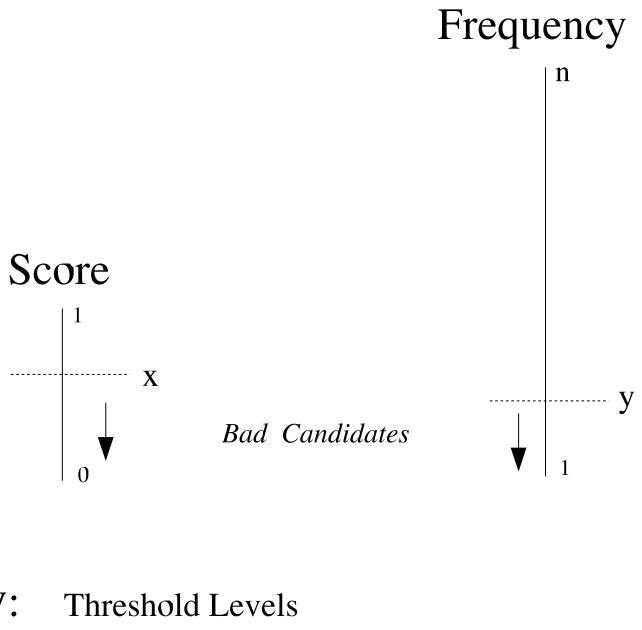


Figure 3.11: Determining threshold Levels

low frequency words from evaluation set. Other methods of filtering are looking into knowledge sources such as the parts of speech of aligned texts, machine readable dictionaries, cognate heuristics, etc. [Melamed, 1995]. In this paper a simple operation of annihilating those words that are aligned with equal score to different words is made. The results of the first attempt to screen the candidates with higher score and high frequency is presented in Table 3.3.

For score >= 0.7 and $\Sigma(Word_j + Word_k)_i > 5$, among the 38 errors, 30 of them are due to candidates with $\Sigma(Word_j + Word_k)_i$ between $6 - 9$. Hence, the threshold for frequency is set to > 9. Again, keeping the frequency threshold fixed the score is lowered until 0.55. For scores below 0.55 the accuracies went below 80%.

Table 3.3: Candidates of high score and high frequency.

Score	$\Sigma(Word_j + Word_k)_i$	Correct	Compounds	Wrong	Total	% Correct
$>= 0.7$	> 5	123	16	38	177	**69.49%**
$>= 0.6$	> 9	134	8	20	162	**82.72%**
$>= 0.55$	> 9	172	9	24	205	**83.90%**

To exploit the low frequency words, a two step analysis is assembled. First a higher threshold is set for them, second a filtering algorithm is designed to screen those words with multiple equal score translations. For $\Sigma(Word_j + Word_k)_i$ between $6-9$ with score $>= 0.8$, 64.71% precision has been obtained before filtering and 82.35% after filtering.

The filter for one and two frequency words, selects all words that match with a score of 1.0 with one and only one word. After filtering, precisions of 51.61%and 43.55% for two and one frequency words (i.e. $\Sigma(Word_j+Word_k)_i$ equal to 4 and 2) respectively is achieved.

3.3 Analysis of the results

The score threshold level for which a good percentage was found before filtering is 0.55. This means the distributions of translation candidates need only overlap in almost half of the case. This is a good news for inflectional

variants of Amharic that fail to align quite well with their counterpart.

Surprisingly enough our method works well even with low frequency words. The translation pairs need to have a frequency sum > 9. This means that each word on average needs to appear in the text only 4.5 times. This is without filtering. With filtering words of frequency 3 also give good results. Most other existing systems use a higher frequency threshold [Sahlgren and Karlgren, 2005, Fung and McKeown, 1994].

The weakness of this system lies on the inability to handle multiword compounds. Verbal compounds as well as many nominal compounds are written as two separate words in Amharic[Amsalu and Gibbon, 2005a]. Split compound alignments are reckoned as wrong matches. Excluding them from the result set, the accuracy of the experimentation increases to 87.76%.

Lets give an exemplary explanation of the case, for more clarity of the facts. The analogue for the word *'disciple'* in Amharic is *'dăkă mäzmur'*. The constituent words always come together. Nevertheless, a statistical alignment system knows them to be two separate words. Yet, since they always appear as a unit, each one of them are likely to match with every word in the English text with equal score. Suppose we have,

Score <disciple , dăkă>= 0.7 and

Score <disciple , mäzmur>=0.7

It is easy to excavate them from the result set by simply setting a conditional rule that if a word is aligned with a value which is its best score with two terms, then accouple the two terms as strings of a compound and align the single word to them, i.e.,

$$Score < disciple\ ,\ d\ddot{a}k\ddot{a}\ m\ddot{a}zmur > = 0.7$$

Corpus data can be used to find which string comes first. But there are two problems that block us from using their score as a measure of their association. First, compounds could be inflected. Inflection may alter either or both of the elements. If the compound takes a prefix, the first element will be affected. If the compound takes a suffix the second element will be changed. This will mess up the scores. The second problem arises for the reason that in most cases, the second part of the compound can exist unbound. And when it occurs independently it has alltogether another meaning. In the example multiword compound, the second part *'mäzmur'* means *'song'*.

The best plausible solution would possibly be to mark compounds as one word right from the beginning. That way, even if they are inflected they will only be affected like any other word would. In an attempt to excerpt compounds from the corpus, bigram distributions of words were generated. The procedure followed is,

1. Take monolingual text

2. Generate bigram word list

   Given a text $word_i\ word_{i+1}\ word_{i+2}\ word_{i+3}....word_{n-1}\ word_n$, a list of,

   $word_i\ word_{i+1}$

   $word_{i+1}\ word_{i+2}$

   $word_{i+2}\ word_{i+3}$

   .

   .

.

$word_{n-1}$ $word_n$

3. Generate type and frequency for each bigram token

4. Extract the high frequency bigrams

Perhaps because the document size was small there were many non-compound bigrams that occurred as frequently as the compounds.

3.4 Summary

The work described in this chapter demonstrates that alignment of disparate languages using statistical methods is viable. It is also possible to gain good translation matches even for low frequency words with the assistance of simple filtering measures.

There are two problems that need to be solved, however. The first problem is that the alignment system gives 1:1 alignments while that should not be the case always. In Chapter 4 a way of achieving 1:m alignments from Amharic to English is discussed. The second problem is that such methods of alignment are regrettably short of achieving high recall. In general, the use of statistical methods is to achieve high precision at a cost of recall. However, given a data size of text, we would like to have a lexicon which would be a good coverage. If not, the lexicon generated would be of minimal use for natural language processing systems and tools it is deemed to serve. This surely is a serious problem, particularly for languages with scarcity of data.

In Chapter 7 a method of maximising recall is discussed.

Chapter 4

Scaling up from Word to Chunk Alignments

In this chapter an algorithm that synthesises English chunks that are equivalent to Amharic words from parallel corpora is described. Often, when aligning bilingual corpora we may not get a 1:1 alignment of words. This is more so in disparate languages pairs. To resolve this problem the widely used solution is to break down more complex words to their underlining components. In this paper an approach that works the other way round where several words in the simpler language are rather brought together to form a phrase equivalent to the complex words in the other language is reported.

4.1 Introduction

Natural languages are made of words that are fully formed by rules of generation that vary across languages. In morphologically complex languages

a word might be inflected to contain lots of information in it, while in simpler languages the same information might be expressed using several words. This leads into a situation where, given complex-simple bilingual text pairs, we often have word alignments of 1:m which are not easy to extract using automated systems. Amharic-English bitexts fall into this category. In this paper an attempt to upgrade the 1:1 alignment output in Chapter 3 to 1:m alignments is reported.

In Section 4.2 Amharic morphosyntax is briefly discussed. In Section 4.3 the results of 1:1 alignments of words using a statistical method of alignment is presented. Section 4.4 describes methods used to cluster the correct chunks of English words. Evaluation of the results obtained is made in Section 4.5.

4.2 Amharic morphosyntax

Amharic is a morphologically complex Semitic language whose basic units are mostly consonantal roots. All classes of words and particularly verbs are highly inflected. Prepositions, articles, pronouns and conjunctions are often or always bound to other classes of words. On the other hand, English is a weekly inflected language with simple morphology. This results in the fact that Amharic texts are made of a few number of words in comparison to English; and Amharic texts are consisted of relatively many low frequency words in general. Table 4.1 presents the statistics of words in the dataset used.

By the same analogy, Amharic sentences are normally shorter than En-

Table 4.1: Word statistics.

Language	Tokens	Types
Amharic	20347	6867
English	36537	2613

glish sentences. For clarity, examplenary description of the case is presented. Taking the first sentence in Amharic bible and its counter part in English a 1:2 ratio in number of words is observed:

በመጀመሪያ እግዚአብሔር ሰማያትንና ምድርን ፈጠረ

In the beginning God created the heavens and the earth

The Amharic version has 5 words and the English version has 10 words. The difference in length comes from the difference in the morphological rules of the two languages. In this particular example, the counter parts of the grammatical words *In*, *the* & *and* in Amharic are not free morphemes. The preposition *in* and the conjunction *and* are affixes and the definite article *the* (which may also occur as a suffix) is omitted in this instance. If these grammatical words are removed, we also have 5 words in the English sentence. On the other hand if Amharic words decomposed, the number of morphemes will be increased:

በ-መጀመሪያ እግዚአብሔር ሰማያ-ት-ን-ና ምድር-ን ፈጠረ

The number of morphemes increased to 10. The numbers are identical, but the decomposed morphemes do not really exactly match to those in English.

በ = the preposition *in*
አት = plural marker
ን = accusative marker (does not exist in English)
ና = the conjunction *and*

Using an affix stripper, it is possible to easily remove some of the affixes. However, it is quite often difficult to get the borders where the affixes get in contact with the stem. One reason is because Amharic has syllabic writing system, as a result whenever an affix that begins with a vowel gets in contact with the consonant in the stem there is a change in grapheme. Second, vowel clusters formed during contact undergo alternation process which then make it difficult to identify the affixes. That is observed in the example on the word ሰማያት (pl). The stem of this word is ሰማይ (sg). When the suffix which begins in a vowel is in contact with the last syllable either simply a consonant or a vowel there occurs a change in the grapheme. The modification in the last symbol of the stem which becomes the penultimate in the inflected word is observed on the leg of the symbol.

ሰማይ + አት(at) = ሰማያት

Note that there are symbols for vowels which are written when they occur at the beginning of a word or morpheme as in the affix in the example (vowels seldom occur in the middle of the word when they are pronounced not in conjunction with the consonant that precedes them).

There are various statistical methods proposed for identifying complex chunks of expression in a language [Church and Hanks, 1990, Smadja and McKeown, 1990, Brown et al., 2005]. However, purely statistical methods

often fail for various reasons. In our case even if an expression in Amharic could occur several times, a slight difference in inflection causes a miss on the target. The rareness of such chunks also makes them unidentifiable. An attempt to generate bi-gram and tri-gram chunks was made to see the possibilities of capturing compounds that are separate words. The results showed that other chunks that are not compounds or even expressions that do not make sense occurred as frequently as the real chunks (see also [Dunning, 1993]). As a result other linguistic approaches alone or with combination of statistical methods were developed. These approaches were mainly based on pattern or template recognition [Jacquemin, 1991, Bourigault, 1992, Smadja, 1993, Daille, 1994]. These techniques have been applied to bilingual text alignment later [Daille et al., 1994, Smadja et al., 1996b, McEnery et al., 1997].

Considering the complexities in Amharic and the scarcity of linguistic tools for dealing with such structures and most importantly the increased usability that can be obtained from rather aligning inflected words with chunks of English words for systems such as machine translation systems, we decided to construct the chunks of English text equivalent to the inflected forms by developing a chunk parser. The previous example sentence would then be aligned in the way in Figure 4.1.

But first it is necessary to obtain 1:1 alignments, because even if it is possible to construct the chunks we do not know which chunk should be aligned to which word. So, first rough 1:1 alignments are generated and then relaxed the English words to obtain the correct chunks.

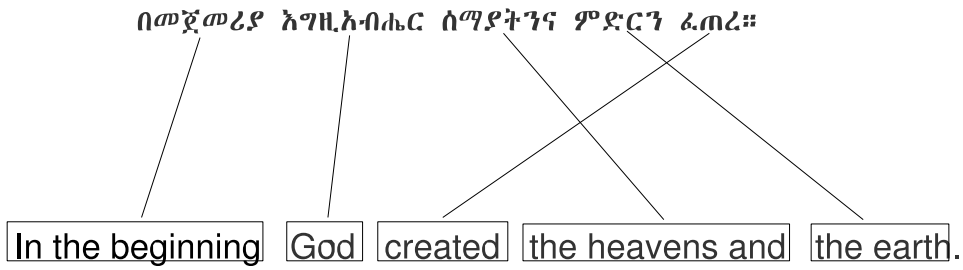


Figure 4.1: Model of 1:m alignment.

4.3 1:1 alignment

The 1:1 alignment in Chapter 3 align weakly inflected content bearing words in English to the highly inflected Amharic words which gave us alignments of the type in Figure 4.2.

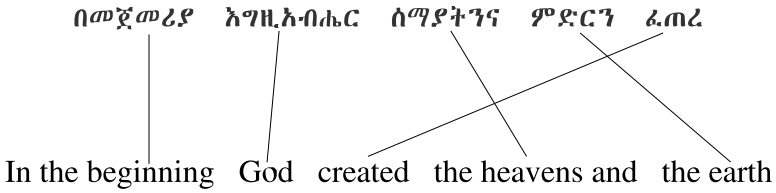


Figure 4.2: Alignment with gap.

In the 83.9% precision alignment in Chapter 3, a lexicon of 205 correctly aligned words are obtained. Among these aligned words, 42% of them are rough 1:1 alignments of inflected Amharic words with non-inflected English words. Therefore, after obtaining the first rough 1:1 alignment, we attempted to incorporate function words to the lexical word which in the Amharic case is the lemma (approximately) of the inflected words.

4.4 Constructing word cluster: 1:m alignment

To help us understand what kind of clusters need to be formed and how to recognise them we take an excerpt of some of the cases in the 1:1 match output set. Table 4.2 presents Amharic words together with their gloss and the corresponding English translations.

Amharic	**Gloss**	**English**
ለቄሣር	*for Caesar*	Caesar
መረባቸውን	*their nets*	nets
ሙሽራው	*the bridegroom*	bridegroom
ሰይፍና	*swords and*	swords
ሰገዱለት	*they worshipped him*	worshipped
ቅፍርናሆምም	*Capernaum also*	capernaum
በልዩ	*with various*	various
በመሸም	*in the evening*	evening
ወደማይጠፋበት	*cannot be quenched*	quenched
አትጨነቁ	*do not worry*	worry
በርባንን	*Barabbas(acc)*	barabbas
ይፈርዱበታል	*they will condemn him*	condemn
ጠገቡ	*they are satisfied*	satisfied

Table 4.2: Clusters.

In Table 4.2 one can see that prepositions, articles, pronouns, negation markers and conjunction happen to be bound with the word that is equivalent to the English lexicon, which is in many cases the content bearing word. The word order in which the function words occur in relative to the content bearing words is in many case regular, which gives us an opportunity to identify the word clusters needed. The most salient instances are discussed

in brief subsequently.

4.4.1 Articles

Articles or the function of articles in Amharic is not represented in the same way as in English. The definite article comes either as a suffix *-u* attached to the noun it modifies or is omitted. In the sentences below, (2) is grammatically correct, but the form used both in written and spoken language is the form in (3) omitting the article.

(1) *wonbär-u* (the chair)

(2) *midr-it-u molala nat* (*-it* feminine)

(3) *midr molala nat*

(*midr* = The earth, *nat* = is, *molala* = spherical)

The use of the indefinite articles is not necessary in Amharic texts in general. When there is a need to stress singularity the term *and* (one) is placed before the noun it modifies. In cases where there are other words in between for example an adjective, such as,

The red ball

The Amharic counter part has the definiteness marker attached to the adjective. Thus, the chunk *The red* would be equivalent to *qäy-u* and *ball* would be equivalent to *kwas*.

qäy-u kwas (*qäy*=red, *kwas*=ball).

4.4.2 Subject and object markers in verbs

In Amharic verbs are inflected in agreement with the subject in the sentence. This is always the case whether the subject is explicitly indicated or not. This marker varies with number, gender and person:

säbär-ä (he broke)

säbär-äc(she broke) etc.

If the object is definite, an object marker suffix is also concatenated next to the subject marker.

wonbär säbär-ä (he broke a chair)

wonbär-u-n säbär-ä-w (he broke the chair).

-e on the verb is the subject marker (3rd person, singular, masculine) and *-w* is an object marker (3rd person, singular, masculine). The noun (the object) is also inflected for definiteness and *-n* is an accusative marker. On the other hand in English the verb does not accommodate such markers. The subject and object necessarily exist independently. Syntactically, the subject comes first followed by the verb followed by the object in simplest cases. There are cases that cannot be handled robustly, however. The problem comes in cases when pronouns have the same forms in nominative and accusative cases or accusative and possessive cases such as:

It dropped first.

He broke *it*.

In the first case it is a subject in the second case it is the object of the

verb. When it is subject it comes before the verb and when it is object it comes after the verb. This creates confusion as to whether to take it with the word following it or the word preceding it. The same is true for her in the accusative and when used to indicate possession:

He fired *her*.

Her pencil is red.

4.4.3 Negation

Negation in Amharic is represented by the prefix *al-* and the suffix *-m*. For example:

al-säbär-ku-t-m (I did not break it)

In this example the whole English sentence should be one chunk, because we have the subject and object markers as well as the negative marker together with the verb.

4.4.4 Prepositions

Prepositions are prefixed to nouns, pronouns and adjectives. Many of them have the form CV (one symbol when written in Amharic writing system). But in cases where the preposition is longer than one symbol such as *wädä* (*to* as used indicating direction as in *He went to school*), they are written as separate words. Their position is just before the object they modify similar to the pronouns in English.

It is evident that these words which are bound in Amharic are found

around the content word. Hence it is necessary to develop a parser that could produce these chunks correctly. The English clusters generated are similar to what [Abney, 1991] calls chunks. Abney, compares chunks to prosodic patterns. His goal was to generate clusters which would consist of content words surrounded by function words. In this sense we are also doing the same thing, except in our case the chunk should match with an Amharic word. Therefore, we are constrained by the morphology of Amharic words.

Apparently, a parser that traps chunks parallel to Amharic words has been designed. The parser is fundamentally a window that can perform three basic operations: *relaxing*, *contracting* and *sliding*. The window starts at the beginning of an English sentence and relaxes to the right until it finds a chunk that is saturated, i.e. something equivalent to an Amharic word . Often it has to relax one step further to check if the next word is part of the chuck, if it is not it contracts one step backwards. A padding symbol is placed to mark the end of the chunk and the window slides to search for the next chunk. What happens in the end is that fewer units of text than the number of words in an English text are formed. Ideally the number of units in the English text will be equal to the number of words in the corresponding Amharic text.

These structures are purely language dependent and the algorithm is governed by the morpho-syntactic rules of both languages. Since we already have the 1:1 aligned lexicon, there was no need to deal with the whole document. Instead, we identified the chunks for the words in the translation lexicon. The procedure used is a bit more complex than searching for a sin-

gle window. This is because the words in the initial lexicon have multiple occurrences in the documents. So all the co-occurrences of the translation equivalents throughout the text are searched and then the most frequently used form is considered as the most likely match.

4.5 Evaluation

Very strict evaluations were made in that even if the chunk constructed may have the necessary adjuncts, if it contained any single additional word which in many cases is also a function word, it is considered as a wrong alignment.

The output shows that 72% of chunks were correctly constructed. An attempt to compute the damage caused on alignments that were correct 1:1 alignments has also been made but the attempt to construct the chunks resulted in wrong 1:m alignments. Note that all words in the lexicon were passed through this second procedure, there is no way of automatically knowing whether the translation pairs are perfect 1:1 translations or not so the procedure itself is needed to identify the correct alignments. The precision obtained after all was 97%. Thus there was a 3% damage.

4.6 Summary

A chunking algorithms for English words which are equivalent with Amharic words has been developed in this chapter. This chunks allow to make 1:m alignments from Amharic to English respectively. The morphology and syntax of both languages were taken into consideration in the process. The

results obtained are encouraging, however, we do not claim that we have considered all possibilities. More investigation on the relative morphosyntactic rules of the two languages needs to be done. An attempt to construct chunks in English sentences has been made but not in Amharic sentences. Equal treatment of both languages would be fair and possibly improve the results.

On the other hand, generating chunks does not solve the difference in frequency of translations created due to inflection of one of the languages. For that it is necessary to develop a morphological analyser for at least the complex language is necessary. Successive chapters discuss attempts made to develop a shallow morphological analyser for Amharic. For equal treatment a simple stripping of the most salient affixes from the English words is also performed.

Chapter 5

Amharic Morphology

This chapter describes the morphology of Amharic words. A design of a morphological is also presented. Amharic is a Semitic language with an influence from Cushitic languages which is more visible on the syntax. Amharic verbs exhibit the typical Semitic non-linear word formation with intercalation (interdigitation) of consonantal roots with vocalic patterns. This also applies to deverbal nouns and adjectives. The term *root* is used to refer to lexical morphemes consisting of consonants, *radical* for consonant constituents of roots; and *stem* for intercalated forms. We decided to describe the rules of word formation for each class of words; since there is greater regularity of the inflectional and derivational rules across parts of speech and the subclasses within them.

5.1 Verbs

Verbs are morphologically the most complex word class in Amharic, with many inflectional forms. A substantial set of words in other word classes are derived primarily from verbs. Consonantal roots carry semantic values of Amharic verbs. Stems are constructed from these consonantal roots by the intercalation of vowels governed by patterns of the type CVCVC, CVCC etc. Prefixes and suffixes are then concatenated to the stem in the regular linear way.

5.1.1 Roots

Amharic roots are constructed only from consonants. The meaning of the word lies on the consonantal roots and these consonants do not change during inflection of the word. There are different views on how many radicals exist in verb roots. Three of the most dominant views are briefly discussed in the successive paragraphs.

[Dawkins, 1960], classifies verbs in five groups based on the number of radicals they have. Those are:

1. Uncontracted three-radical (e.g. *sbr* "break"),

2. Contracted three-radical verbs with a vowel instead of the last radical (eg. *sma* "hear")

3. Contracted three-radical verbs, with a vowel instead of the penultimate radical (eg. *lak* "send", *Tes* "smoke")

4. Uncontracted four-radical (eg. *mrmr* "investigate")

5. Contracted four-radical verbs, with a vowel instead of the last radical (eg. *znga* "forget")

[Bender and Fulas, 1978], on the other hand, classify simple verbs as:

1. biliterals (*smh* - vowelled form *sma*),

2. triliterals (*sbr*, *zngh* - vowelled form *znga*) and

3. quadriliterals (*mnzr*).

They say quinqiliterals and sexiliterals do not really exist; rather they are derived from triliterals. The biliterals are what [Dawkins, 1960] calls contracted triradicals. Dawkins explains that at some time during the evolution of the language the stem has contracted, one of its radicals being lost. An example of such kind of verbs is the verb *sma* (hear). [Bender and Fulas, 1978] classify this verb as a biliteral with final *a*; the vowelled form being *sma* and the base form *smh* (triliteral base form from which it comes). The term *base form* is used by [Bender and Fulas, 1978] to mean the phonological part of a lexical entry.

[Yimam, 1999], argues that all verbal roots have uniformly three radicals in their underlining representations and that variations in the number of such radicals in surface forms is a result of extensions and/or reductions of one or more of the three radicals.

5.1.2 Stems

Most simple verbs have five verbal stems [Dawkins, 1960, Bender and Fulas, 1978]. These stems provide the basis for all tense, aspect, mood etc. They can have patterns of gemination that are of one of the types A, B or C as formulated by Dawkins. These patterns are:

- *Type A*: penultimate consonant geminates in perfect only
- *Type B*: penultimate consonant geminates throughout the conjugation
- *Type C*: penultimate consonant geminates in perfect and contingent.

Table 5.1: Conjugation of a typical triradical type A verb root *sbr* (break)

Aspect	Pattern	Stem	Description
Perfective	CVCVC	säbbär	broke
Imperfective	CVCC	säbr	break, will break
Jussive	CCVC	sbär	break! let sb. break!
Gerund	CVCC	säbr	breaking
Infinitive	CCVC	sbär	to break

One can observe in Table 5.1, that in the root *sbr* (trilateral, type A), the only vowel intercalated is *ä*. In principle this is the only vowel intercalated in other verbs too. When vowels other than the usual *ä* occur in stems, it is the result of reduction of consonants at sometime in the evolution of the language or the existence of sharp or flat consonants. Table 5.2 shows the

the vowelled form of *hwq* a verb belonging to the class trilateral, type A verb root with a lost radical. The vowel *a* occurs due to the reduction of the glide *h* in the root.

Table 5.2: Trilateral, type A verb root with a lost radical *hwq* (know)

Pattern	CV CVC	CVCC	CCVC	CVCC	CCVC
Stem	äwäq	awq	Ïwäq	awq	awäq

The vowel *o* alternatively occurs in dialects in cases where flat consonants are followed by the vowel *ä* such as $k^{\mathrm{w}}ä$, $q^{\mathrm{w}}ä$, $g^{\mathrm{w}}ä$ etc. occur to create radicals of the form *ko*, *qo*, *go*, etc. (there are about 11 of them, some are said to have disappeared). When the vowel is short it is converted to *u* instead of *o*. See Table 5.3 for an example of a verb with such alternative forms used across dialects.

Table 5.3: Trilateral, type A verb root with a flat consonant $q^{w}Tr$ (count)

Pattern	CVCVC	CVCC	CCVC	CVCC	CCVC
Stem	q^{w}äTär	q^{w}äTr	q^{w}Tär	q^{w}ätr	q^{w}Tär
Stem	qoTär	qoTr	quTä r	qotr	quTär

The vowel *e* also refers to an underlining sharp consonant such as $C^{y}ä$, $T^{y}ä$, making *Ce*, *Te*. You may see in Table 5.4, the conjugation of $T^{y}s$ which is a verb that belongs to class 8[1] of [Bender and Fulas, 1978]'s classification.

[1]See Appendix A for the classification by [Bender and Fulas, 1978]

Table 5.4: Class 8 verb root $T^y s$

Pattern	CVC	CVC	CVC	CVC	CVC
Stem	Tes	Tes	Tis	Tes	Tes

Except for the second person jussive, the stem does not stand on its own. The minimally inflected stem must at least have a subject marker. The verb may be inflected for person, gender, number, mood, and tense [Amare, 1997, Yimam, 1994]. The verb is also inflected for benefactive, malfactive, causative, transitive, passive, negative, etc [Berhane, 1992].

5.1.3 Person

The verb is inflected for person of the subject and object. This is always the case whether the subject and object are explicitly described or not. They can exist in the nominative and accusative form. These affixes have different forms in a verb of the perfective or the imperfective. Table 5.5 presents the inflections that occur on the verb for different persons and when the person is in the nominative. The inflection takes place on the perfective verb for the simple past and the imperfective for the present and future tenses.

The verb is also inflected for the object, the inflections are as listed in Table 5.6.

The object marker affix does not exist when there are suffixes indicating malfactive or benefactive functions. For example in sentences such as,

ÏnCät-u-n säbär-ä-l-N (he broke the wood for me)

Table 5.5: Inflection for person in the nominative

Person	Past Sg.	Pres/Fut Sg	Past Pl	Pres of Fut Pl
1	-ku/hu	Ï-	-n	Ïn -
2Mas	-k/h	t-	-achu	t- -u
2Fem	-x	t- -i	-achu	t- -u
3Mas	-ä	y-	-u	y- -u
3Fem	-äc	t-	-u	y- -u

Table 5.6: Inflection for person for the object

Person	Singular	Plural
1	-N	-n
2Mas	-h	-achu
2Fem	-x	-achu
3Mas	-wu	-acäw
3Fem	-at	-acäw

ÏnCät(wood)-*u* (def. article)-*n* (Accus. marker)

säbär(broke)-*ä* (3P, Masc. Sing.)-*l* (Benf.)-*N* (1P, Sing.)

The *l* in *säbär-ä-l-N* indicates that the action was done for the benefit of someone.

ÏnCät-u-n säbär-ä-b-N (he broke the wood - against my will)

The *b* indicates that he did it against the benefit of someone. In both cases

there is no any object marker on the verb indicating the object *thewood*. Had it not been for the $-l$ and $-b$ the sentence would be constructed as,

ÏnCät-u-n säbär-ä-w (the $-w$ indicating 3P, Masc., Sing.(the wood)).

5.1.4 Mood

When a sentence is constructed for different moods the verb is inflected accordingly cf. Table 5.7. In the mood of negation, when the prefix *al-* comes in contact with the affix for the subject marker of the imperfect, it makes a change that results in the affix in the brackets. Hence, for instance, for the first person singular, *al-Ï-* results in simply *al-*. In most cases, however, the negation *al-* occurs together with *-m* at the end of the verb. One can say *almäTa* (I will not come) or *almäTam* and the second one is more common and probably almost the only one used in writing.

5.1.5 Tense

The main categories of time being three, these broader categories of time can have smaller subdivisions that create six categories. These six categories are as in Table 5.8.

These tenses are expressed in two verb forms (the perfective or the imperfective) and the auxiliary verbs *alä*, *näw* and *näbär*. The remote and recent past are based on the imperfect verb form. They also take the same types of inflection. Their difference lies on the auxiliary verb they use. The remote past is constructed with the help of *näbär* which is the auxiliary verb of the

Table 5.7: Inflection for mood

Person	Statement.	Command	Question	Negation
Singular				
1	-ku/hu	-	l-	al-Ï- (al-)
2Mas	-k/h	-	-	al t- (at-)
2Fem	-x	-i	-	al-t- (at- -i)
3Mas	-ä		y-	al-y- (ay-)
3Fem	-äc		t-	al-t- (at-)
Plural				
1	-n	-	Ïn-	al- Ïn- (an-)
2	-achu	-u	-	al- t- (at-)
3	-u	-	y- -u	al- y- (ay-)

past. Whereas the the recent past is constructed with the auxiliary verb of the present *alä* (*-al* when it comes as suffix) cf. Table 5.9.

The simple past tense indicates that the action is in the past but does not indicate when in the past (how far) it was performed (See Table 5.10).

The verb in the present and future tenses is inflected in the same way. These tenses can be ambiguous unless an adverb indicating the future or the present is within the sentence. See Table 5.11.

The present continuous tense shows an action that is taking place continuously at the present. These tense is formed by the prefix *Ïyä-* and by the auxiliary verb *näw* when used with a perfective stem. An example is,

Table 5.8: Tenses in Amharic

Tense	Description
Remote past	An action that took place a long time ago
Recent past	Recently completed action
Simple past	No indication of how far in the past the action took place
Present and Future	The present and future tenses in sentences can be ambiguous unless adverbs of time are included in the sentence
Present continuous	Continuous action in the present
Past continuous	Continuous action in the past

Kasa ÏnCät Ïyä-säbärä näw (Kasa is breaking wood)

The prefix *Ïyä-* actually only shows the continuity of the action and not the period of the action. Rather the auxiliary verb *näw* tells the action is in the present. So now when constructing the past continuous tense the same prefix *Ïyä-* is used but together with the auxiliary verb of the past which is *näbär*. In the past the above sentence is written,

Kasa ÏnCät Ïyä-säbärä näbär (Kasa was breaking wood)

It can also be correct to express the past continuous with the prefix *y-*,

Kasa ÏnCät y-säbr näbär

But this statement could be ambiguous in that it can also show a habitual

Table 5.9: Inflections for remote and recent past tense

Person	Singular	Plural
1	-e	-(ä)n
2Mas	-h	-achu
2Fem	-x	-achu
3Mas	-u	-(ä)w
3Fem	-a	-(ä)w

Table 5.10: Inflection for the simple past tense

Person	Singular	Plural
1	-ku/hu	-n
2Mas	-h	-achu
2Fem	-x	-achu
3Mas	-ä	-u
3Fem	-äc	-u

action in the past. So one can only be sure of continuity when using *Ïyä-*. Note that the verb in the past continuous tense is also in the imperfective.

There are verbs that do not obey the above inflectional affixes. These verbs are what the early linguists call, the verb to have and verb to be. In Amharic there are two verbs of this type *näw* (verb to be) and *allä* (verb to have). They need to be handled separately in computation.

Table 5.11: Inflections for the present and future tenses

Person	Singular.	pl
1	t- -al-ä-hu	În- -al-ä-n
2Mas	t- -al-ä-h	t- -al -achu
2Fem	t- -al-ä-x	t- -al -achu
3Mas	y- -al	y- -al -u
3Fem	t- -al-äc	y- -al -u

5.1.6 Phonological alternations

Change of vowel occurs when vowel clusters come in sequence during affixation. The phonological alternations that occur in Amharic words are summarised in Table 5.12. The cluster is made taking an entry in the first column first followed an entry in the first raw. For example *aa* is alternated into *a*, *ae* is alternated into *aye*, ...

Another processes that takes place during word formation is palatalisation. Palatalisation causes change of sound that affect the grapheme. This occurs when a dental consonant is followed by the vowel *ä* or *i*. Table 5.13 shows the corresponding dental and palatal consonants.

The changes that occur are such that, for instance *di* gets changed to either *ji* or *j* whereas *dä* changes to only *jä*.

Table 5.12: Alterations in vowel clusters

-	a	e	i	o	u	ä
a	a	aye	ay	awo	aw	a
e	eya	eye	ey	ewa	ew	ä
i	iya	iye	i	iwo	iw	iya
o	owa	oye	oy	owo	ow	o
u	uwa	uye	-	uwo	uw	wa
ä	a	äye	-	o	äw	ä

5.1.7 Derived verbs

Verbs could also be derived from other words. But instead of being derived from words of other categories, Amharic verbs are derived from other verbs [Amare, 1997]; [Yimam, 1994]. These derived verbs can be *adragi, tädäragi asdäragi, adaragi, tädaragi, däraragi, tädärarragi, adäraragi* morphemes. Using the verbs sbr and flg, each of the forms are as presented in Table 5.14.

5.1.8 Compound verbs

A few verbs in Amharic can only exist as compound verbs. They are created by combining the words *alä* (said) or *adärägä* (did) with words that cannot stand on their own.

zm alä (he kept quite)

qäT alä (he stood straight up)

Table 5.13: Dentals that change to palatals

Dentals	palatals
d	j
t	c
T	C
z	Z
s	S
'S	C
n	N
l	y

zm adärägä (he made sb. quite)

qäT adärägä (he made sth. straight)

However, all verbs can potentially form compounds in combination with *alä* (passive) *adärägä* (active) to indicate mostly sudden happening of an event or action. Look at the forms below:

sbr alä (it broke suddenly)

sbr adärägä (he broke it suddenly)

Note that these verb compounds are written as two separate words. When they get inflected, however, they inflect as a single word would. A prefix to such compounds is attached on the left side of the first word and a suffix is concatinated on the right side of the second word.

Table 5.14: Verbal derivations

Derivation	Derived form for the root *sbr*	Gloss
adragi	säbär	he broke
tädäragi	tä-säbär	it got broken
asdäragi	as-säbär	he made sb. break sth
adaragi	a-sabär	he took part in breaking
tädaragi	t-sabär	they broke each other
däraragi	säbabär	he broke sth to several pieces
tädärarragi	tä-säbabär	it got broken into several pieces
adäraragi	as-tä-säbabär	he made sb break sth to several pieces

5.2 Nouns

Amharic nouns may be grouped into basic (or primitive) and derived nouns. Primitive nouns are nouns that cannot be formed from other words by derivation. They exist by themselves. *bet* (house), *märet* (earth) and *Ïsat* (fire) are examples of such primitive nouns. There are also nouns derived from verb roots and other parts of speech. Some examples of derived verbs are *dbq* (secretive - derived from the verb *dbq*- to hide), *dägnät* (cheerfulness - derived from the adjective *däg*- cheerful), *xumät* (post - derived from the verb *xom*-to appoint). Both basic and derived nouns are inflected for number, gender, case and definiteness.

5.2.1 Number

Most plural nouns are formed by adding a plural marker affix to the singular form. Some plurals are, however, formed by full reduplication of the singular noun. Nouns that are inherited from Geez are exceptions to these ways of making plural. It is also possible to express plurality by using a quantifier number or an adjective before the noun in a sentence while not changing the noun. Instances of each of these processes of pluralisation are discussed in the next paragraphs.

Suffix plural markers

There is a lot of regularity among nouns in forming plurals. Nouns that have a vowel ending take the suffix $-woc$ while those with a consonant ending take the suffix $-oc$ to form their plurals. Some nouns with a vowel ending may also as an alternative omit the last vowel and attach $-oc$. Table 5.15 describes this phenomena.

Table 5.15: $-oc$ and $-woc$ plural suffixes

Singular	Plural	Gloss
alga	algawoc	bed(s)
bet	betoc	house(s)
wuxa	wuxa-woc / wux-oc	dog(s)

Nouns are not the only category of words that take $-oc$ as a suffix to form plurals but adjectives too form their plurals in the same fashion. So,

one cannot consider a word to be a noun because of only its plural ending. Nouns inherited from Geez do not necessarily take these suffixes while making plurals. Table 5.16 lists some nouns inherited from Geez. Often, the mere affixation of plural markers is avoided.

Table 5.16: Plurals of singular nouns inherited from Geez

Singular	Plural	(Alternative)	Gloss
mäzgäb	mäzagbt	mäzgäboc	archive(s)
mäShaf	mäSahft	meShafoc	book(s)
kokäb	käwakbt	kokäboc	star(s)
anbässa	anabst	anbässoc	lion(s)
ngus	nägästat	ngusoc	king(s)

But some plural nouns in Geez are inherited in Amharic as singular and they take additional plural marker. Table 5.17 shows the plural suffix $-oc$ attached to an already plural noun. Note, however, that the plural does not indicate collection of the same thing but collection of similar things. For example qus-a-qus is not used for several pieces of one type of kitchen item but collection of may be dishes, forks, stove etc.

Country and tribe name plural marker

The suffix $-awi$ attached to names of countries and tribes ending in consonants yields nouns that describe citizenship/tribe one belongs to. This suffix will be $-wi$ when the name of the county or tribe is a vowel. Such derived nouns formed by affixing $-awi$ have the feminine gender indicator

Table 5.17: Plurals of plural nouns inherited from Geez

Geez pl.noun	Amharic pl.
Mekuannt	mekuanntoc
Liqawint	liqawintoc

suffix $-awit$. For plurals, these nouns have the suffix $-yan$ and drop the last $-i$ (m) or $-it$ (f) (See Table 5.18).

Table 5.18: Plurals indicating citizenship

Noun	Gloss
ityoPiya	Ethiopia
ityoPiyawi	Ethiopian (sg. masc.)
ityoPiyawit	Ethiopian (sg. fem.)
ityoPiyawyan	Ethiopian (pl.)

Using numbers or adjective quantifiers

Plurals are also formed by using a singular noun preceded by a number or an adjective which describes quantity. For example, one can say,

They both indicate plural nouns. But in the case where a non-number quantifier is used, it is more like the *quantifier + sing noun* refers to a bigger quantity really many, while the one with *quantifier + pl noun* means a few.

Table 5.19: Phrases indicating plurality

Phrase	Gloss
10 fyäloc	10 goats or
10 fyäl	10 goat
bzu fyäloc	many goats or
bzu fyäl	many goat

Uncountable nouns

Uncountable nouns on the other hand never take affixes for number. Their magnitude is described by using quantifiers the same way as in English. See Table 5.20.

Table 5.20: Magnitude in uncountable nouns

Uncountable noun	Gloss
bzu skuar	a lot of sugar
Tiqit skuar	a little amount of sugar

Collective nouns

It is common to use collective singular nouns. For instance the word *gomän* (cabbage) is a singular noun but used for several sticks or leaves of cabbage. But plurals which indicate collection of similar items (not the same kind of items) are formed by reduplication of the noun itself. For example Table 5.21,

Table 5.21: Plurals made by reduplication

Sing	Gloss	Plural	Gloss
qTäl	leaf	qTäl-a-qTäl	leaves
Tre	cereal	Tre-a-Tre	cereals
qus	kitchen item	qus-a-qus	utensils

5.2.2 Gender

Amharic has two genders: masculine and feminine. Things that are naturally male have masculine gender and things that are naturally female have also feminine gender. For things that are not either male or female naturally, the gender female is used when the thing is either small or adorable the gender male is used otherwise. This makes the situation precisely contextual and determined only by the individual using it momentarily.

Naturally masculine things are sometimes called with a feminine gender, especially in spoken language, when the speaker wants to emphasise how small or how clever or adorable the thing (also works for human beings) is. Masculine nouns do not have any gender marker to indicate their being masculine, while, feminine nouns have the feminine gender marker affix -*it* to indicate their being feminine. When the noun has a vowel ending the suffix $-yit$ is used rather than $-it$. Table 5.22 presents an examples explaining this.

If the feminine gender marker $-it$ and the definiteness marker $-u$ are removed from the above nouns, the gender is not clear. But with $-u$ as

Table 5.22: Feminine and masculine nouns

Noun Fem.	Noun Masc.	Gloss
ljitu = lj(child)-it(fem.)-u(definiteness)	lj-u	The child
dmmät-it-u	dmmät-u	The cat
znjäro-yit-u	znjäro-w	The monkey
doro-yit-u	doro-w	The hen(cock)

in $lj - u$ and $dmmet - u$, the noun refers to masculine definite noun "the child" and " the cat", respectively. When the noun has a vowel ending, however, $-w$ is attached rather than $-u$. In feminine nouns the definiteness marker $-u$ can also be be replaced by $-wa$. They serve the same purpose; $ljitu = lijitwa$. Each one of them may be used more often in certain areas. The gender marker in the noun should always agree with the gender marker of the verb. The gender marker -*äc* on verbs indicates feminine gender. Thus, only nouns that have the feminine gender marker -*it* go with verbs having the suffix -*äc*. For instance, while,

bäg-itu mot-äc "The sheep (fem.) died" ,

is acceptable because the affixes $-itu$ and -*äc* indicate feminine gender. The sentence

bäg-itu mot-ä "the sheep died",

is not acceptable since -*ä* is a masculine gender marker of the verb. Some nouns are distinct in gender lexically as shown below without marking gender

markers externally or explicitly. The examples in Table 5.23 are nouns that indicate gender without using any gender marker(s).

Table 5.23: Lexically distinct genders

Noun (masculine)	Gloss	Noun (feminine)	Gloss
bäre	ox	lam	cow
abbat	father	Ïnnat	mother

Inanimate objects are generally treated as masculine. But, occasionally, there are certain things (i.e. inanimate objects) that should be treated as feminine as in the following examples.

Table 5.24: Inanimate objects with feminine gender

Inanimate object	Gloss
Sähay	sun (f)
Cäräqa	moon (f)
märet	earth (f)

5.2.3 Case

Nouns inflect for accusative and genitive cases. In the accusative nouns change their form by simply taking the suffix $-n$ independent of number, person and gender. It is a little bit complicated in the genitive, because

because the case markers vary depending on the possessor's number and gender. The complete listing of case markers is presented in Table 5.25.

Table 5.25: Case markers for the accusative and genitive forms

Case	Features		
Nominative	Accusative	Genitive	Person
No case markers	-n	-e	1p masc./fem. sing.
	”	-h	2p masc. sing.
	”	-x	2p fem. sing.
	”	-u	3p masc. sing.
	”	-wa	3p fem. sing.
	”	-acn	1p pl.
	”	-achu	2p pl.
	”	-acäw	3p pl.

5.2.4 Definiteness

Definite noun markers are suffixes that vary depending on the gender of the noun. Their distribution is determined phonologically. For instance, singular nouns ending with consonants use the definite noun marker $-u$ for masculine gender and $-wa$ or $-itu$ for feminine genders. If the nouns end with a vowel, the definite markers change to $-w$ for masculine and $-wa$, $-yt$, $-ytu$, or $-ytwa$ for feminine. Table 5.26 shows instances.

Table 5.26: Definiteness in singular nouns

Nouns (sing).	Definite			
-	Gloss	Masculine	Feminine	Gloss
bet	House	bet-u	bet-wa	The house
wänbär	Chair	wänbär-u	wänbär-wa	The chair
tämari	Student	tämari-w	tämari-wa/ytu/yt/ytwa	The student
bäg	Sheep	bäg-u	bäg-wa/itu/it/itwa	The sheep
ahya	donkey	ahya-w	ahya-wa/yt	The donkey
ToTa	monkey	ToTa-w	-	The monkey
geta	master	geta-w	-	The master
doro	hen	doro-w	doro-wa/ytu/ytwa	the cock/hen

Plural nouns, on the other hand, take the only definite noun marker $-u$ irrespective of their gender Table 5.27. That is, definite noun markers for singular masculine nouns and plural nouns (masculine as well as feminine) are the same.

5.2.5 Compound nouns

Several nouns are formed by compounding two words. The two words are either $noun + noun$ or $noun + adjective$. When the two words are combining, the vowel -*ä*- is used to link the two words. A few compound nouns are presented in Table 5.28. Compound nouns, just like simple nouns, inflect to

Table 5.27: Definiteness in plural nouns

Noun (pl.)	Definite noun	Gloss
bet-oc	bet-oc-u	The houses
wänbär-oc	wänbär-oc-u	The chairs
bäg-oc	bäg-oc-u	The sheep (pl)

accommodate number, gender, definiteness, case, etc in same way as non-compound nouns.

Table 5.28: Compound nouns

Noun	Noun	Compound Noun	Gloss
bet (house)	krstyan (Christian)	betäkrstyan	church
mänfäq (six months)	lelit (night)	mänfäqälelit	midnight
bunna (coffee)	bet (house)	bunnabet	bar
bahr (sea)	zaf (tree)	bahrzaf	eucalyptus tree

A noun could take one or several of the affixes for the different inflection. When there are multiple affixes, the position of an affix relative to the noun or other affixes is constrained. Table 5.29 gives a clue on the relative positions of the affixes.

Where, S1 denotes Plural marker; S2 denotes Affix of possessive pronouns for the genitive; S3 denotes definiteness marker; S4 denotes Accusative suffix S5 denotes emphasis marker. The order among the suffixes S1, S2, S3, S4

Table 5.29: Affix positions in nouns

S1	S2	S3	S4	S5
-oc	-e	-u	-n	-m
-woc	-h/k	-w	-	-
-	-u	-wa	-	-
-	-wa	-itu	-	-
-	-acn	-ytu	-	-
-	-achu	-ytuwa	-	-
-	-acäw	-	-	-

and S5 is,

S1-S2-S3-S4-S5.

Nouns can be derived from verbs as well as from other nouns and from adjectives

5.2.6 Nouns derived from verbs

Nouns can be derived from verbal roots. [Dawkins, 1960] discusses five groups of nouns derived from verbs: infinitive, the agent, the instrument, the manner and the product.

The infinitive: is a verb-noun, since it partakes in the nature of both a verb and a noun. It is a verbal in that it describes an action and it is a substantive as it is the name of the action. Both the infinitive and the -ing form of the

English verb-nouns are translated by the Amharic infinitive.

The agent: denotes the performer of the action of the verb.

Instrument: Agent and instrument forms are quite distinct. The instrument denotes means, or place employed for performing the action of the verb.

The product: denotes what is produced by the action of the verb.

The manner: denotes the manner in which the activity is performed. Table 5.30 has instances showing the different forms.

Table 5.30: The infinitive, agent, instrument, product and manner

root	infinitive	Agent	Instrument	Product	Manner
sbr	mä-sbär	säbar-i	mä-sbär-iya	sbrat	a-säbabär
flg	mä-fäläg	fälag-i	mä-fäläg-iya	flagot	a-fälaläg
hwq	mä-awoq	awaq-i	mä-awoq-iya	Iwqät	-
flh	mä-fla-t	a-fl-i(a-f-yi)	mä-afl-iya	fl	a-fälal
frs	mä-fräs	a-frax	mä-afräx-iya/a	frax	a-färaräs
Tyq	mä-Täyäq	Täyaq-i	mä-Täyäq-iya	tyaqE	a-Täyayäq
qdh	mä-qda-t	qäj-i	mä-qj-iya/a	qji	a-qädad
n*w*r	mä-nor	n*w*ar-i	mä-nor-iya	nuro	a-n*w*an*w*ar

[Amare, 1997] and [Yimam, 1994], present several instances of nouns created from verb roots by intercalating different vowels between the radicals; and others that are created by adding suffixes to the root without vowel intercalation cf. Tables 5.31, 5.32, 5.33.

Table 5.31: Nouns derived from verb roots by intercalation.

Verb root	Derived noun	Intercalation
msl	msl	CCC
srg	serg	CeCC
ngr	neger	CeCeC
skr	skar	CCaC
Tbb	Tbeb	CCeC

Table 5.32: Nouns derived from verb roots by consonant reduction

Verb root	Derived noun	Reduction
Tmh	Tm	h→null
T^wm	Tom	T^w →To

5.2.7 Nouns derived from other nouns

Nouns take certain affixes to form derived nouns. In Table 5.34 instances of the different affixes that nouns take other nouns is presented.

5.2.8 Nouns derived from adjectives

Adjective also take certain affixes to form nouns. Table 5.35 shows typical examples.

Table 5.33: Nouns derived from verb roots by attaching different suffixes

Verb root	Derived noun	Affixation
grd	grd-ox	CCC-ox
Tbq	Tbq-na	CCC-na
xlm	xlm-at	CCC-at
drg	drg-it	CCC-it
wTh	wT-Et	h→null, CC-Et
ftn	fätän-a	CäCäC-a

Table 5.34: Nouns derived from other nouns

Noun	Derived noun	Affixation
lj	lj-nät	NOUN-nät
kbr	kbr-ät	NOUN-ät
wxa	wxa-o	NOUN-o
self	self-äNa	NOUN-äna

5.3 Pronouns

Pronouns can be free or bound. The bound pronouns are discussed with verb morphology. Free personal pronouns take the affixes for nouns when used in the Accusative and Genitive. Table 5.36 presents free personal pronouns.

Table 5.35: Nouns derived from adjectives

Adjectives	Derived noun	Affixation
däg	däg-nät	ADJ-nät
qrb	qrb-ät	ADJ-ät
blh	blh-at	ADJ-at

5.4 Adjectives

Adjectives modify nouns. The adjectives come before the noun they modify. They can be of different functions:

- Those that describe type: *qäy* (red), *Tqur* (black), ...
- Those that describe behaviour: *sänäf* (lazy), *kfu* (mean), ...
- Those that describe size/quantity: *tlq*(gross), *bzu* (many), ...

Adjectives could be simple or derived. The number of simple adjectives are relatively fewer. Some simple adjectives are listed in Table 5.37. Adjectives are inflected for number, case and definiteness.

5.4.1 Number

There are two ways of plural formation: by adding the suffix $-oc$ to the singular adjective or by reduplication of penultimate consonant and insertion of the vowel *a* cf. Table 5.38. The number of the adjective and the number of the noun it modifies have to agree. In the sentences below, the second

Table 5.36: Personal pronouns

Person	Pronoun
Singular	
1	InE
2Masc	antä
2Fem	anci
2Respect	Irswo/ antu
3Masc	Irsu/Isu
3Fem	Irswa/Iswa
3Respect	Irsacäw/Isacäw
Plural	
1	INa
2	Inantä
3	Inärsu/Inäsu

sentence is not correct because the adjective is plural but the noun is singular. Remember the number of the noun has also to agree with the number of the verb.

aCaCr ljoc mäTu = Short (pl) children came (pl)

aCaCr lj mäTu = Short (pl) child came (pl)

Table 5.37: Simple adjectives

Adjective	Gloss
yäwah	meak
mogn	foolish
tlq	gross
tnx	small or little
räjm	tall
fäTan	fast

Table 5.38: Inflection for number

Singular	plural
fäTan	fäTan-oc
blh	blh-oc
räZm	räZaZm
säfi	säfafi

5.4.2 Gender and definiteness

Adjectives exhibit a masculine or feminine gender. The genders are expressed by the definite article suffixes which are also gender markers in the noun (commonly, -it and u). The gender of the adjective should also agree with the gender of the noun. When not using the definite article, the adjective could be either feminine or masculine. This can be contextually understood by considering the gender of the noun modified by the adjective. The following

example demonstrates that. The adjective $wofram$ does not have any gender marker, but it modifies a feminine noun.

wofram lam = fat cow

The other forms of definite article coming with nouns, such as *-yt/ytu* for feminine nouns with vowel ending and -w for masculine noun with vowel ending also work for adjectives. The alternative *-wa* in place of *-u* for the feminine also holds.

5.4.3 Case

The case observed for adjectives is the case of the noun they modify. But the case for the genitive seen in the nouns is not marked in the adjective. Only the accusative marker $-n$ makes a change in the form of an adjective modifying an accusative noun. The sentence below suffices to support the discussion,

qäyu-n Cama adärägäw

qäyu-n (the red) *Cama* (shoes) *adärägäw* (he wore)

Note that, when the accusative affix exists on the adjective, it is no more in the noun.

Adjectives can be derived from verbs, nouns or from verbal morphemes [Amare, 1997].

5.4.4 Adjectives derived from verbs

Adjectives are derived from the roots of the verbs. Different kinds or methods of derivation from verbal roots exist. Table 5.39 shows the possible conjugations. Insertion of *ä* between the radicals,

Table 5.39: Adjectives derived from verb roots

root	CäCäC	CäCC	CCuC	CäCaC
snf	sänäf	-	-	-
qbT	qäbäT	-	-	-
drq	däräq	-	-	-
rzm	-	räZm	-	-
ITr	-	aCr	-	-
gzf	-	-	gzuf	-
kbr	-	-	kbur	-
Tqr	-	-	Tqur	-
fTn	-	-	-	fäTan
qll	-	-	-	qälal
lgs	-	-	-	lägas

5.4.5 Adjectives derived from nouns

Adjectives are derived from nouns by attaching the suffixes $-eNa$, $-ma/ama$ or $-am$. $-ma$ is attached for nouns with final a and $-ama$ otherwise cf. Table 5.40.

Table 5.40: Adjectives derived from nouns

Noun	Noun-äNa	Noun-ma/ama	Noun-am
nägär	nägär-äNa	-	-
mls	mlas-äNa	-	-
kurat	kurat-äNa	-	-
ayn	-	ayna-ma	-
tärara	-	tärar-ama	-
hod	-	-	hod-am

5.4.6 Adjectives derived from verbal morphemes

By attaching $-awi$, $-a$, or $-u$ on certain morphemes; derived adjectives can be formed cf. Table 5.41.

5.4.7 Compound adjectives

Compound adjectives are formed by compounding a noun and an adjective cf. Table 5.42.

5.5 Adverbs

An adverb describes the time, place and conditions in which the action of the verb took place. The adverbs in Amharic are very few. But the function of the adverb can be accomplished by other words and phrases. Noun phrases, prepositional phrases and subordinate clauses do the function of adverbs.

Table 5.41: Adjectives derived from morphemes

Morpheme	Morpheme-eNa	Morpheme-a	Morpheme-u
zämän	zämän-awi	-	-
xäraf	-	xäraf-a	-
qorat	-	qorat-a	-
qoxax	-	qoxax-a	-
kf	-	-	kfu
zng	-	-	zngu

Table 5.42: Compound adjectives

Noun	Linker	Adjective	Compound adj.
xl	ä	muq	xlämuq
Igr	ä	qäCn	IgräqäCn

The most commonly occurring adverbs are:

gäna (yet)

tolo (soon)

$kfuNa$ (badly)

$gmNa$ (badly) (dialect)

$mnNa$ (how?)

$jlNa$ (foolishly) (dialect)

Some phrases that serve as adverbs are presented in the following sentences:

Thomas nägä fätäna yifätänal = Thomas will take exam tomorrow.

Thomas (Nom.) *nägä* (Noun as Adv.) *fätäna* (Acc) *yifätänal* (verb)

wodä gäTär hEdin = We went to the country

wodä gäTär (Prepositional phrase as Adv) *hEdin* (Verb)

brCqow bä-dngät täsäbärä = The glass got broke suddenly.

brCqow (Nom) *bä-dngät* (Prepositional phrase as adv) *täsäbärä* (Verb)

5.6 Prepositions and conjunctions

Conjunctions and prepositions in Amharic have similar behaviors and they are categorized in the same class of words (mestewadid). They exist as affixes to nouns or verbs (*kä*-(from), *slä*- (about), *lä*-(to, for)).

So much about the morphology of Amharic verbs, the development of a morphological analyser to deal with these word formation processes is discussed in the next chapter.

Chapter 6

Morphological Analysis of Amharic Words

6.1 Introduction

Morphological analysers are the very basic and very useful tools for natural language computing. In a system like the one developed in this project the use of morphological analysers is of paramount importance. In aligning translation texts, when grammatical functions of a word are expressed by inflecting it in one language and in the other language grammatical functions are expressed using independent words, it happens that a very much inconsistent counts of words is observed. Hence, the judgements made based on these counts is also distorted. To avoid this a morphological analyser is developed.

6.2 Design

Natural language texts are made of words that are sequences of morphemes combined in some format. When computing natural language texts, however, morphemes are in many cases more useful than words. Hence, the need for a morphological analyser that decomposes words into their component morphemes. It could also be the case that not all morphemic components are important but a reduced canonical form for words that are related. Some applications that need morpheme level information are: information retrieval, machine translation, thesaurus, spell checkers, hyphenating systems, speech recognition, etc.

A morphological analyser takes a string of morphemes as an input and gives an output of underlining morphemes or morphosyntactic interpretations(lexical forms). For instance the word "students" can be analysed as,

students = student + s or

students = student + noun + plural

Either of the two cases are correct, but what should be included in the analysis side is the choice of the linguist depending on what is needed for the application. One thing is, however, important in either cases; one has to know the rules of word formation in the particular language of concern.

Considering Amharic words, words can be analysed to produce a stem or a root as the canonical form. In Semitic languages the root is in many cases used as a base form, for reasons that the stems are really not of different underlining meaning as long us they are generated from the same root even if they have different grapheme .

It is quite clear that several surface forms are generated from the roots. Figure 6.1 gives a simplified diagram of how big number of words are generated from a single root *sbr* (break).

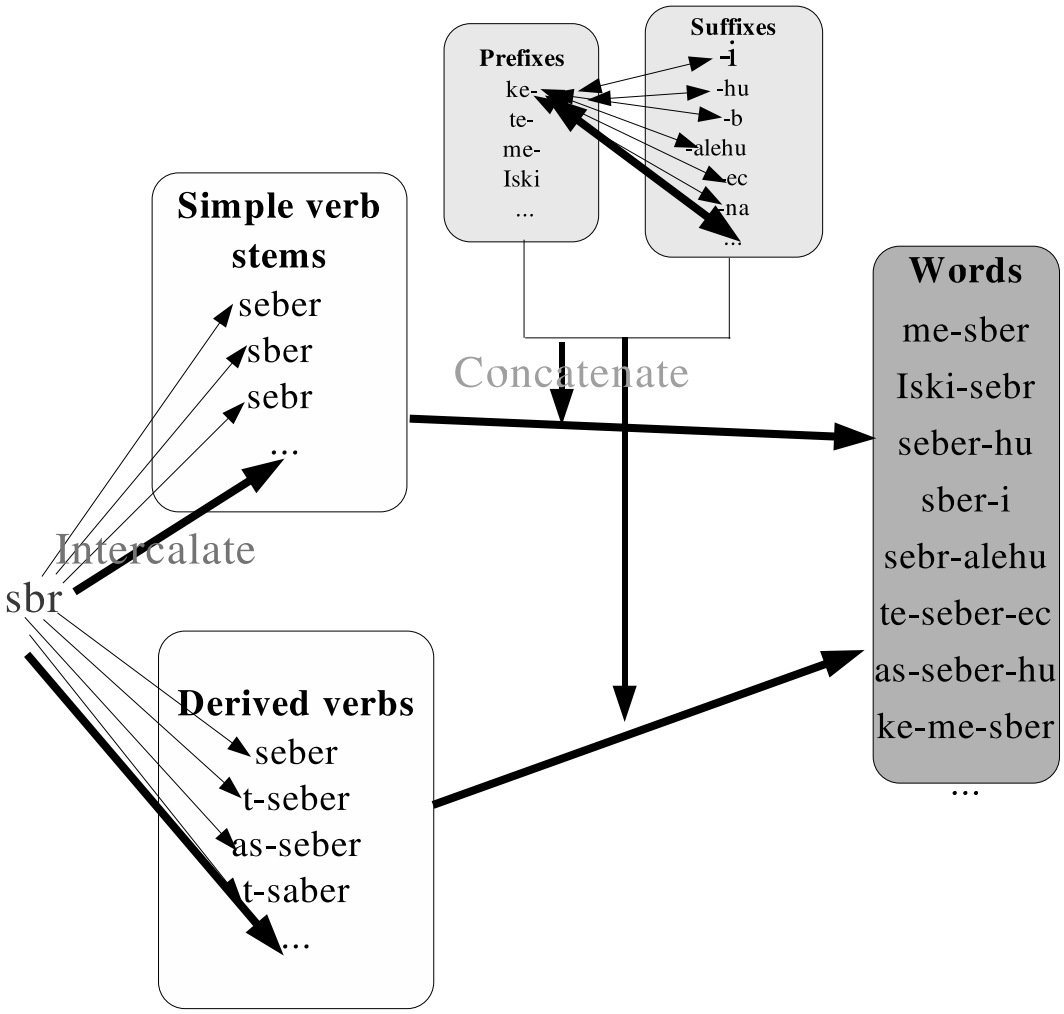


Figure 6.1: Word formation sketch

The function below shows how words can be formed from roots.

$f = \{(r, w) | r \in R \wedge w \in W \wedge W = f(r) = RMP(r)\}$

Where, r denotes the root; w denotes a surface form; R denotes the set of

roots in the language; W denotes the set of words (or verbs in particular) in the language; RMP denotes the rules of morphotactics and phonological alternations.

One can also express the process of word formation in two steps:

$root \rightarrow stems \rightarrow words$

$f = \{(r, s) | r \in R \wedge s \in S \wedge S = f(r) = IC(r)\}$,

generates stems from roots ($root \rightarrow stems$). Where, IC denotes intercalation; s denotes the stem and S denotes the set of stems.

The function ,

$f = \{(s, w) | s \in S \wedge w \in W \wedge W = f(s) = A(s)\}$,

generates words from stems ($stems \rightarrow words$). Where, A denotes affixation.

In the next sections of this paper, the system architecture, the internal behaviours and external interaction of a word generation system is presented. The process of word formation is discussed in depth rather than the process of analysis of words because the analysis is a function of the morphological and phonological rules of word formation. One can visualize a morphological analyser to be a system that is invoked by a user who basically has a word or words and tries to obtain the analysis. The use case diagram in Figure 6.2 shows the interaction of the user with the system,

To analyse a word the system must first recognize the word, which implies that it has a repository of words from which it can search the words. If it does not find the word input by the user in the repository it does not analyse it. Figure 6.3 describes the general work flow during analysis.

The words in the repository are words some how generated from roots.

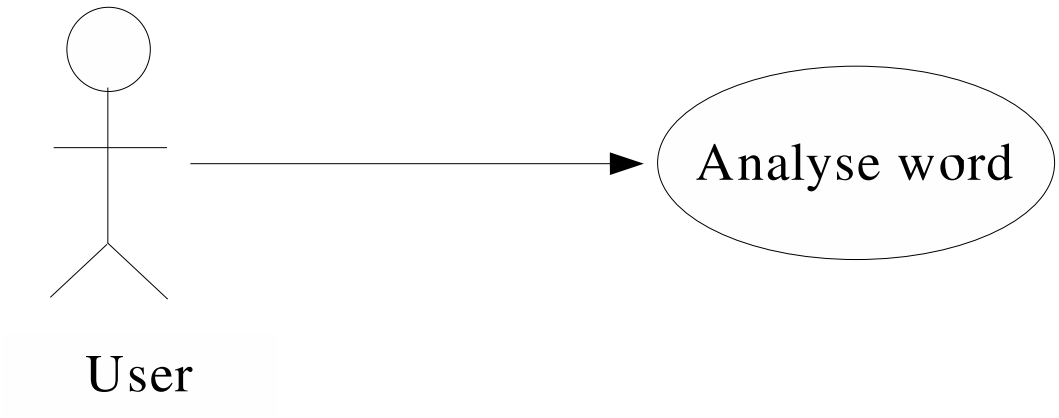


Figure 6.2: Analyse word use case

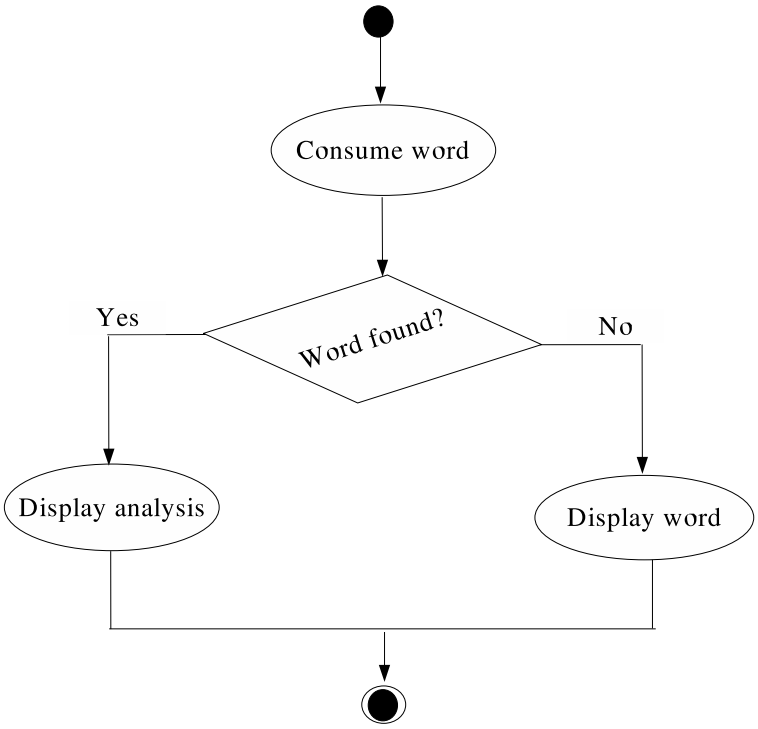


Figure 6.3: System activity to user request

This demands for the system to have a lexicon of roots. But this lexicon of roots needs rules of generation to make up words. In addition affixes that are concatenated to the roots or stems need also to be part of the system. The activity diagram in Figure 6.4 shows the work flow of word generation and analysis.

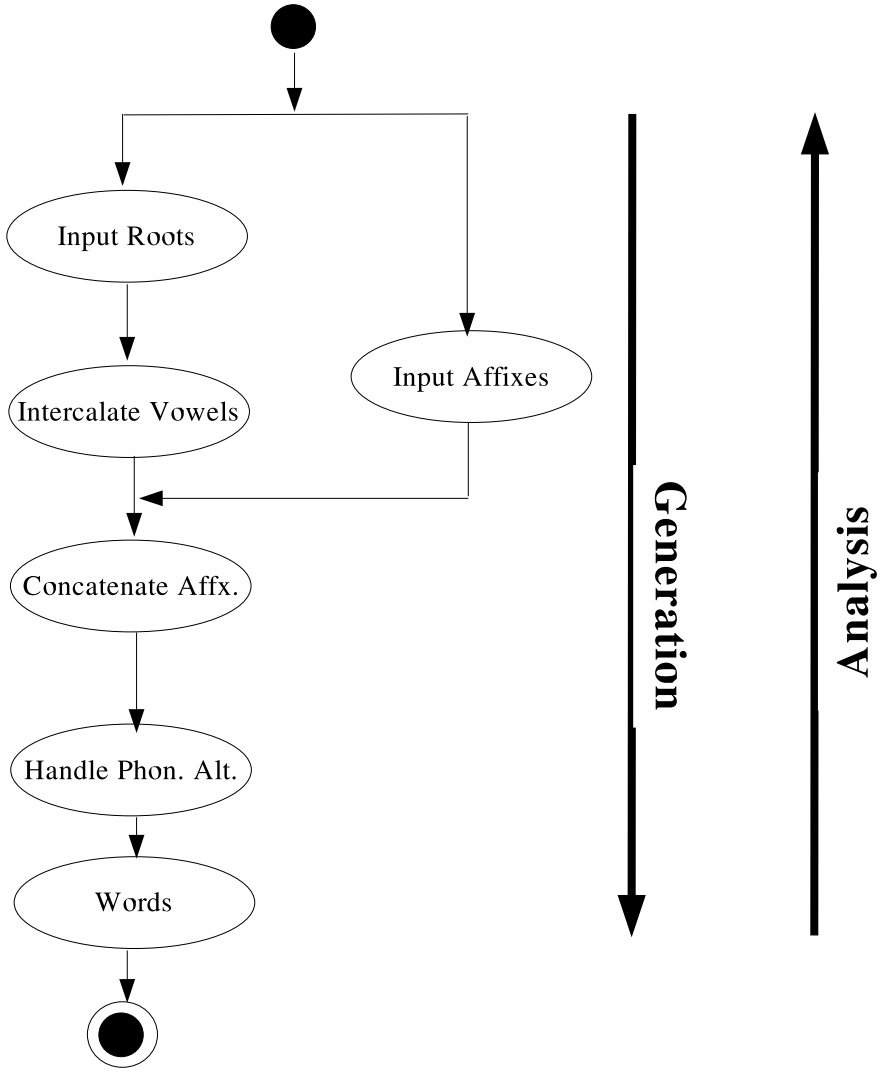


Figure 6.4: Word generation and Analysis

The realization of the activities is made possible by the existence and interaction of different classes in the system. The Class Diagram in Figure 6.5. shows the classes in a morphological analyser and their relationships.

As you may observe in the class diagram; on a root several affixes can be attached and these same affixes can be attached to other roots. Note that affixes are not concatenated to the roots directly but to the stems generated from them. Again from one root several surface forms (words can be generated). The input to the analyser is a list of words obtained from corpora. From a single corpora several input words are naturally obtained, which are then possibly analysed by the analyser.

The major activities in the word formation process are intercalation and affixation. The design of these operations and other activities such as gemination, reduplication, insertion and phonological alternations is presented below.

Intercalation: It is a process by which vowels are inserted in between the consonants of the root. The details for each class of verbs is discussed in the review of Amharic verb morphology in chapter 2. Figure 6.6. shows an interaction of the a triradical root $R_1R_2R_3$ with a vowel e in CV templates CVCVC, CVCC and CCVC. The result is stems $R_1eR_2eR_3$, $R_1eR_2R_3$and $R_1R_2eR_3$.

Gemination: Gemination patterns of Type A, Type B or Type C occur in the stems. A Type A gemination, which is gemination of the penultimate consonant in the perfective is formed as.

$R_1eR_2eR_3 \rightarrow R_1eR_2R_2eR_3$

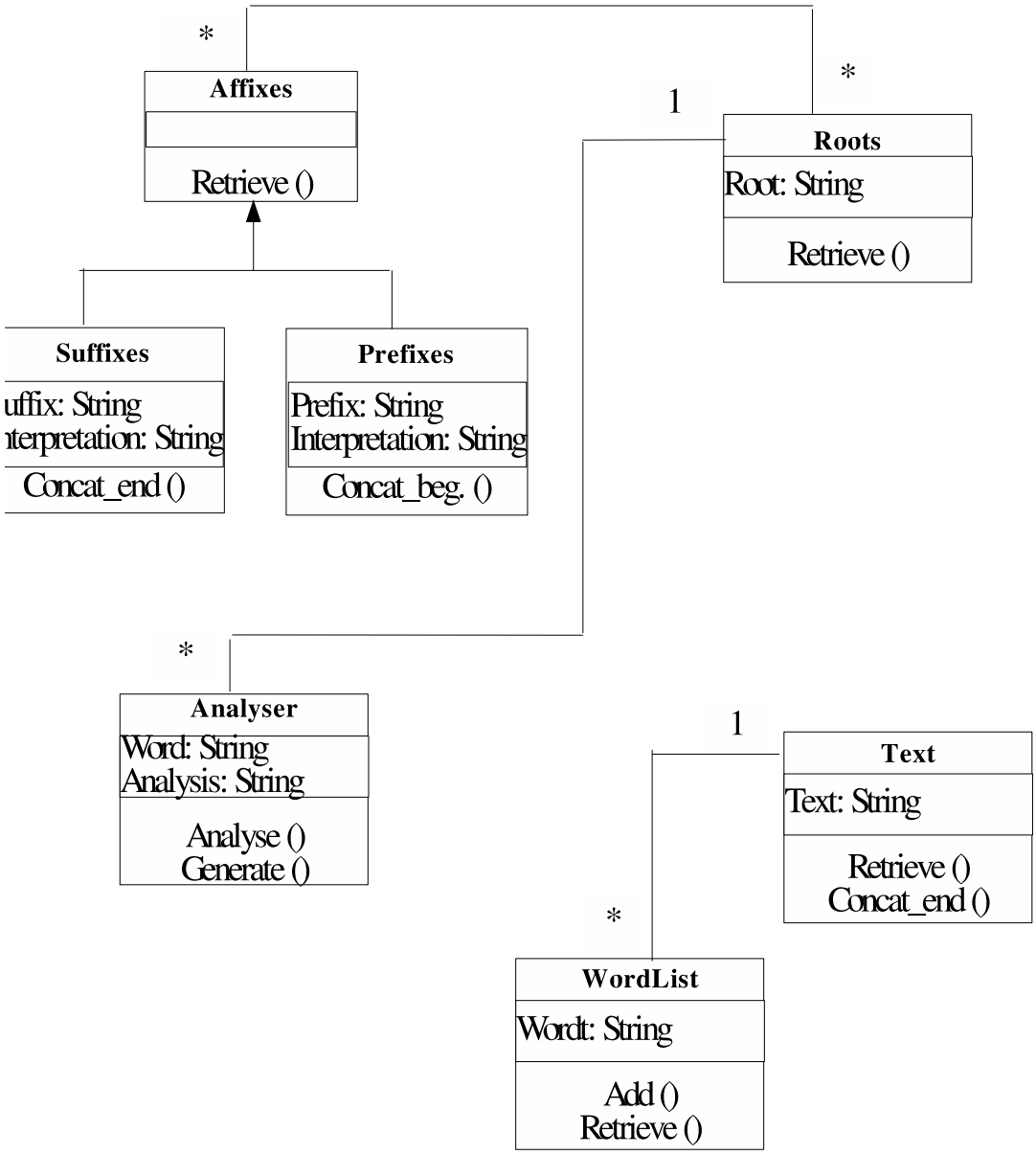


Figure 6.5: Classes and their relation

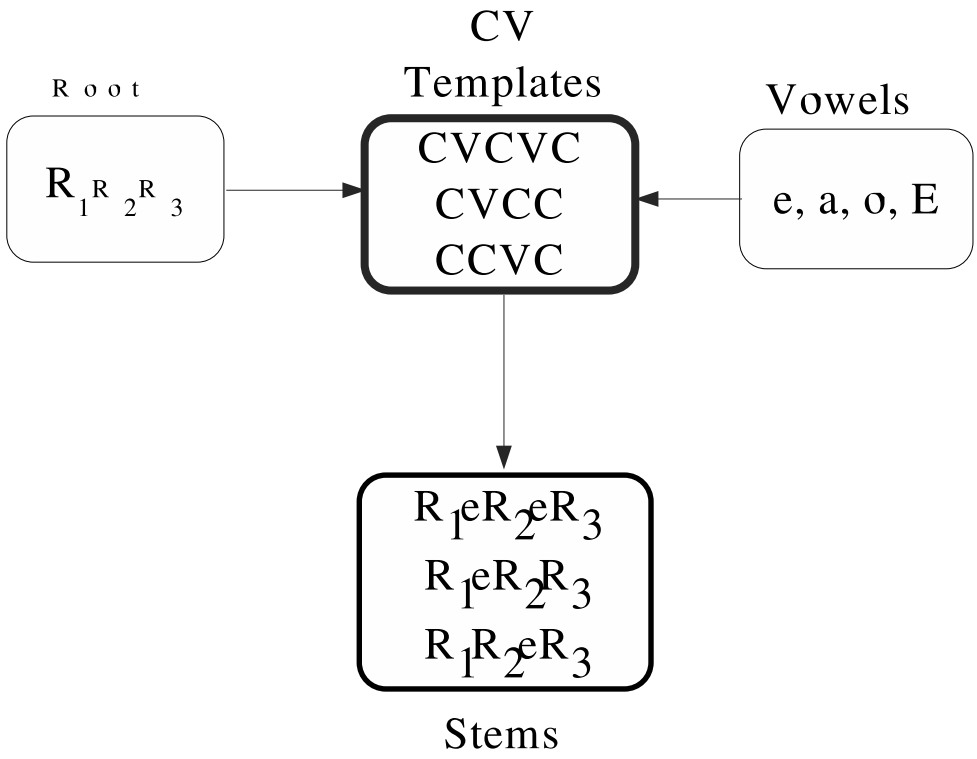


Figure 6.6: Intercalation

Gemination in the Type B verbs is,

$R_1 eR_2 eR_3 \rightarrow R_1 eR_2 R_2 eR_3$

$R_1 eR_2 R_3 \rightarrow R_1 eR_2 R_2 R_3$

$R_1 R_2 eR_3 \rightarrow R_1 R_2 R_2 eR_3$

Gemination in the Type C verbs is,

$R_1 eR_2 eR_3 \rightarrow R_1 eR_2 R_2 eR_3$

$R_1 eR_2 R_3 \rightarrow R_1 eR_2 R_2 R_3$

Reduplication and Vowel insertion: Occur in derived verbs are handled in the manner,

$R_1 eR_2 eR_3 \rightarrow R_1 eR_2 a\ R_2 eR_3$

Where the penultimate consonant is duplicated and a vowel a is inserted in between the penultimate consonant and the duplicate consonant.

Simple vowel change that occurs in verbs where the subject is the causative of the action is manifested in the perfective stem. The vowel next to the first consonant.

$R_1 eR_2 eR_3 \rightarrow R_1 aR_2 R_2 eR_3$

Affix concatenation: Concatenates prefixes and suffixes to the stem. Figure 6.7 describes the process.

In the end a morphological analyser based on finite-state formalisms is developed. The major word formation processes are addressed in depth.

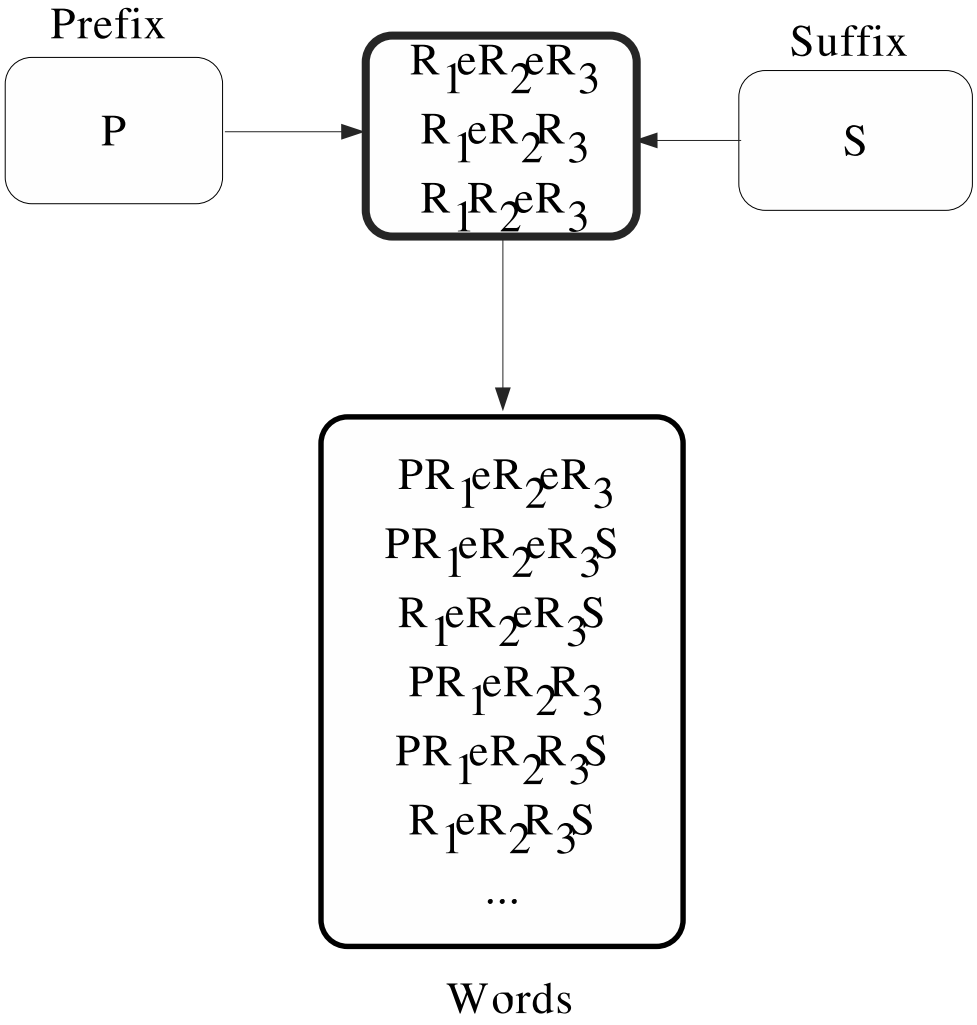


Figure 6.7: Affix concatenation

6.3 Implementation

Many basic procedures in natural language processing standardly employ finite-state techniques for implementation, including tokenisation, phonological and morphological analysis, shallow parsing, spelling correction and others; cf. [Karttunen, 2003]. Morphological constructions can be described particularly efficiently with regular expressions; cf. [Beesley and Karttunen, 2003], [Kay, 1987], [Koskenniemi, 1984], and [Kiraz, 2000].

6.4 Finite-state machines

Finite-state machines (FSMs) are abstract machines. They are consisted of a finite-number of states and transition functions. Given an input, they jump through a series of states according to a transition function. FSMs are compiled from regular expressions. Since most morphological phenomena can be described with regular expressions finite-states machines have become straight forward in modelling morphological processes. FSMs are bidirectional by nature. In other words they can be used for both morphological analysis and generation. FSMs can be represented using a state transition diagram as in Figure 6.8.

6.4.1 Finite-state automata

Finite-state machines with one tape are known as *finite-state automaton* (FSAs). A FSA simply accepts the strings of a single language. It gives

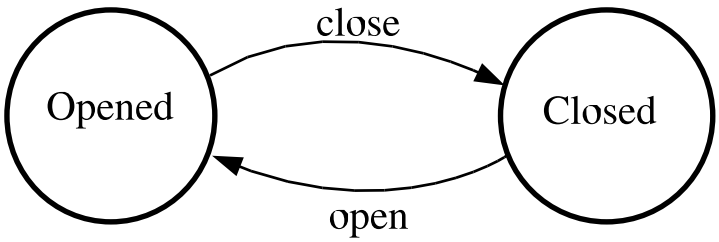


Figure 6.8: Finite-state machine.

a binary output, saying either *yes* or *no* to answer whether the input is accepted by the machine or not. The language defined would contain every word accepted by the machine. If all input is processed the current state is an accepting state, the input is accepted, otherwise not. The acceptor automata in Figure 6.9 has the language containing the strings: *work*, *worked* and *working*.

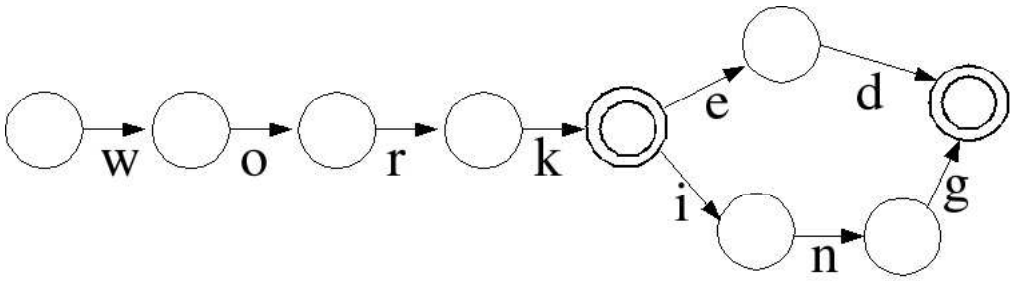


Figure 6.9: An acceptor automata.

Formally, an acceptor finite-state automaton is described in a quintuple $< \Sigma, Q, q_0, \delta, F >$, where:

- Σ is the input alphabet (a finite non empty set of symbols).
- Q is a finite set of states.

- q_0 is an initial state, an element of Q.
- δ is the state transition function: $\delta : Qx\Sigma \rightarrow Q$.
- F is the set of final states, a subset of Q.

6.4.2 Finite-state transducers

Finite-state machines can also have two tapes. Such FSMs are called *finite-state transducers* (FSTs). A FST transduces the contents of its input tape to its output tape cf. Figure 6.10. It encodes a relation between two regular languages. The FST in Figure 6.11 describes the relation between the lower and upper languages:

1. Lower language: *work, worked*, and *working*
2. Upper language: *work, work+past*, and *work+cont*

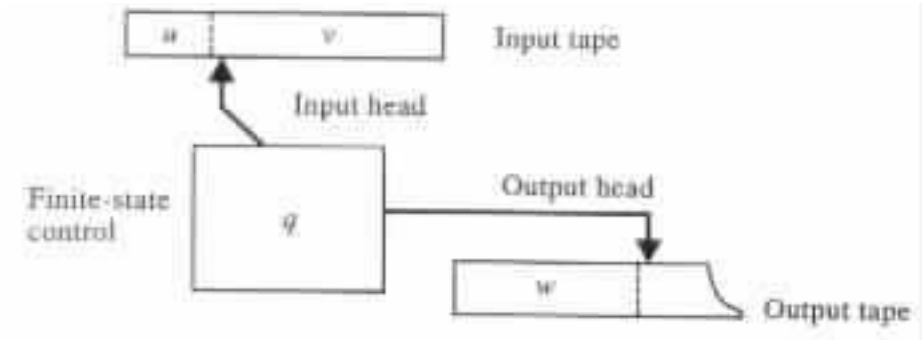


Figure 6.10: FST logic.

A finite-transducer T is formally described in a six tuple $< \Sigma$, Γ,Q,q_0, F, $\delta >$ such that:

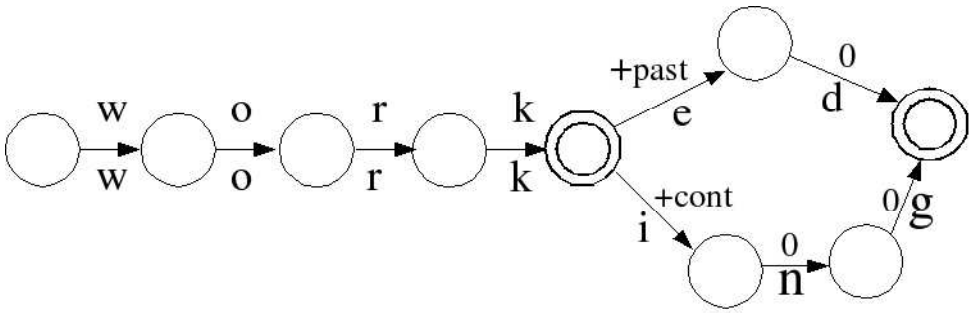


Figure 6.11: Finite-state transducer.

- Σ is a finite set, called the input alphabet;
- Γ is a finite set, called the output alphabet;
- Q is a finite set of states;
- q_0 is the set of initial states (a subset of Q);
- δ is the state transition function: $\delta : Qx\Sigma \rightarrow Qx\Gamma$
- F is the set of final states (a subset of Q).

6.5 Regular expressions

FSTs are generally compiled from regular expressions. Regular expressions are formal languages for representing sets and relations. Regular expression operators help to build complex regular expressions. Most frequently used operators include concatenation, union, subtraction, composition, cross product and replace rules. For regular languages or relations A & B,

- *Concatenation* (A B): extend each string of language A with all the strings of language B
- *Union* (A | B): a language with all the strings of the languages A & B
- *Subtraction* (A − B): strings in language A but not in language B.
- *Composition* (A.o.B): if $A = a : b$ and $B = b : c$ then,

$$A.o.B = a : c$$

- *Crossproduct* (A.x.B): For $A = a$ and $B = b \mid c$ then,

$$A.x.B = a : b \mid a : c$$

Complex Replace Rules allow us to define complicated finite-state tasks. For regular languages A and B,

- *Simple* replacement: $A- > B$
- *Multiple* replacement:$A- > B, C- > D$
- *Context sensitive* replacement: $A- > B \mid\mid L1\_R1$
- *Optional* replacement:$A(- >)B$
- *Cascaded* replacement: $A- > B$ followed by $B- > C$
- *Parallel* replacement:$A- > B, B- > A$

6.6 Finite-state morphological analysis

There are different methods of to analyse words into their basic forms, among which are simply gathering affix lists and stripping them off the words, trying to learn word breaks from corpora and rule based generation and analysis of words using finite-state automata. For reasons of superior functionalities of bidirectionlity, compactness, cleanness and straight forwardness provided by finite-state approaches of morphological treatment, in this project an attempt to use them to describe Amharic mopology in particular and to some extent English morphology has been made.

Morphological analysis using finite-state transducers (FSTs) is based on the assumption that the mapping of words to their analysis constitutes a regular relation, i.e. the underlining forms constitute a regular set, the surface forms constitute a regular set, and there is a (possibly many-to-many) regular relation between these sets. In languages whose morphotactics is morph concatenation only, FSTs are straightforward to apply. Handling non-concatinative (or partially concatinative) languages is, however, a bit complicated and challenging cf. [Kay, 1987, Beesley and Karttunen, 2003, Trost, 2003].

6.7 Formal properties of word forms

The basic morphological modelling convention is that there is a small finite upper bound to root length (e.g. *sbr*) and to intercalated stems forming regular sets. Relations between roots and stems are also regular.

$$root + vocalism + template = stem$$

$$sbr + \ddot{a} + CVCC = s\ddot{a}br$$

Words are constructed from stems by concatenation of prefixes and suffixes, also regular sets. Morphophonological operations have finite contexts. Regular sets are represented in FSAs and regular relations in FSTs. When the input and output of an FST are reversed, the FST models the inverse regular relation. The reversibility property of FSTs is useful. The 'generate' mode is used for generation, the 'accept' mode for analysis (cf. Figure 6.12).

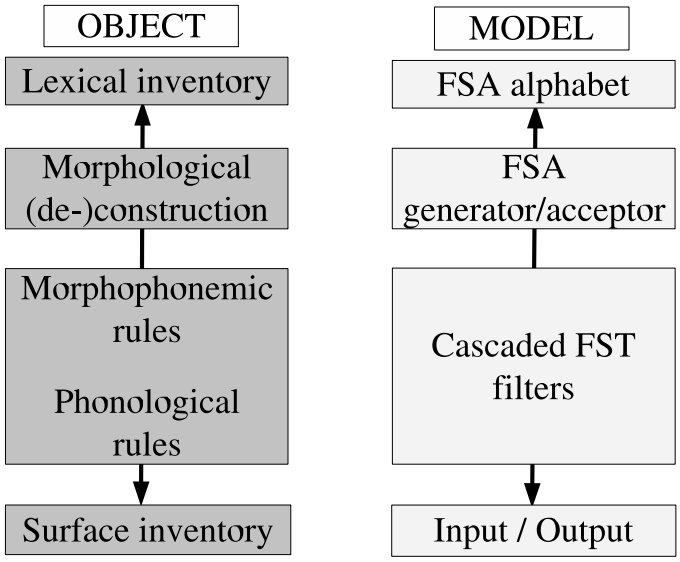


Figure 6.12: Modelling conventions for FSTs.

6.8 Amharic morphology in finite-state formalism

Amharic, like other Semitic languages has challenges to morphological analysis and generation. Among others, complex stem formation rules, reduplication and internal changes and phonological alternations need to be handled carefully. In this project work an attempt to use the Xerox Finite-State Tool (XFST) for the analysis of Amharic words into their corresponding lexical representations has been made. XFST is a general-purpose interactive utility for creating and manipulating finite-state networks [Beesley and Karttunen, 2003]. XFST has an interactive interface providing access to the basic algorithms of finite-state calculus. It provides high flexibility in defining and manipulating finite-state networks. XFST also provides a compiler that includes powerful rule formalisms known as replace rules.

The task of the morphological analyser is to map surface forms into lexcal forms and viceversa cf. Figure 6.13.

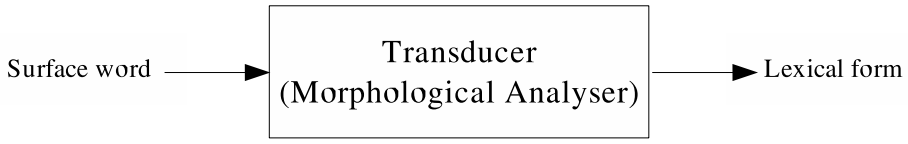


Figure 6.13: Morphological analyser

The surface form is just a set of surface words obtained from corpora(*säbärä* , *säbäräch*, etc). The lexical form consists of a canonical representation of the word and a sequence of tags that show the morphological characteristics of

the form in question and its syntactic category (*[sbr+Perf]+3P+SG+MASC, [sbr+Perf]+3P+SG+FEM*, etc.). The relations between surface forms and lexical forms are represented in a network. Once the network is constructed analysis is just mapping a word in the lower side to the form in the upper side. But to construct such a network two things are needed:

1. A source lexicon that defines the set of valid lexical forms of the language, and

2. A set of finite-state rules that assign the proper surface realization to all lexical forms and morphological categories of the language

The absence of a lexicon of Amharic words in their base form has been a major problem. About 1277 Amharic verb roots were compiled from [Bender and Fulas, 1978]; other irregular verbs were gathered from [Dawkins, 1960]. Deverbal nouns and adjectives were also obtained from these sources. Non–derived adjectives, adverbs, prepositions and conjunctions are few, and were manually collected. Simplex nouns are also hard to find. Lists of names were collected from the Bible, as well as names of places, kinship terms, body parts, local environmental terms and numbers (cardinal and ordinal). The lexicon automata was then compiled with the Xerox lexicon compiler (LEXC).

The generation of word forms is basically divided into two major procedures:

1. stem formation

2. word formation

The n-radical roots, $2 <= n <= 4$ are generated from the phonological part of the lexical entry (referred as a BASE form in this work). Our procedure corresponds to traditional and generative Ethiopian grammatical descriptions [Yimam, 1999, Bender and Fulas, 1978, Dawkins, 1960].

6.8.1 Compiling root lexicon

To compile the root lexicon Xerox lexicon compiler (LEXC) is used. In LEXC the entries are represented in such a list,

!trirad-lex.txt

LEXICON Root

LEXICON Class1

bdn #;

bql #;

bkt #;

blT #;

brd #;

brq #;

bsl #;

bsr #;

Cmq #;

This script generates a network that accepts the strings listed. The network in Figure 6.14 accepts the strings: *mkr*, *mnn*, *mrg*, *mrmr*, *mskr*, *msTr*.

This lexicon is then uploaded on to the stack in XFST using the script,

#!/usr/bin/xfst -f

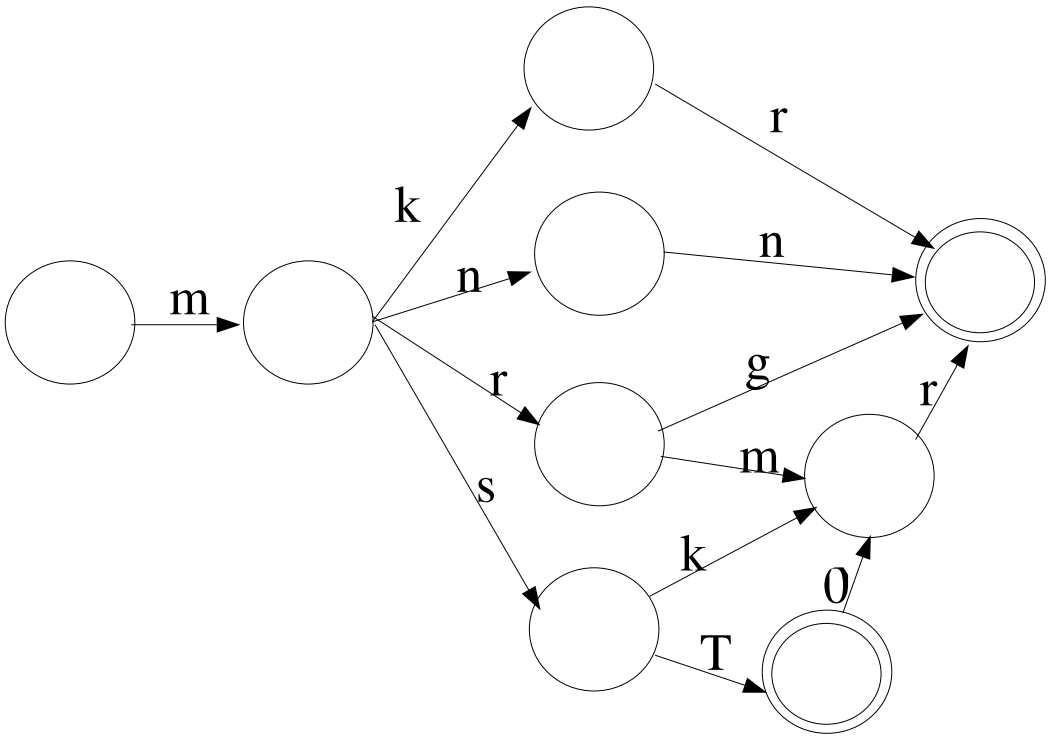


Figure 6.14: A finite-state automata accepting strings.

clear stack

read lexc </home/saba/xfst/verbs/lexicon/class1.txt

define root

6.8.2 Contraction

The contraction of stem where certain consonants are converted into a vowel or a syllable is reduced

Glide 'h': smh ⟶ sma ⟶ säma

h− >a

Flat consonant 'ʷ': qʷlf ⟶ qʷäläf ⟶ qoläf (qʷäläf)

{ʷä}(− >) o

Sharp consonant 'ʸ': Tʸs ⟶ Tʸäs ⟶Tes (Tʸäs)

{ʸä}(− >)e

6.8.3 Intercalation

Stems of simple verbs and deverbal nouns and adjectives are formed by intercalation of vocalisation into consonantal roots. Semitic stem interdigitation has been treated several times; cf. [Kay, 1987, Kataja and Koskenniemi,

1988, Beesley and Karttunen, 2003]. Kay designed a multitape FS technique for the interdigitation of roots, CV–templates and vocalisations in Arabic, and [Kataja and Koskenniemi, 1988] demonstrated interdigitation of Semitic roots (taking Ancient Akkadian as an example) using intersection over regular languages.

In [Beesley and Karttunen, 2003] a 'merge' operator for Arabic stems is described, a pattern filling algorithm which combines two regular languages, a CV template and fillers (root & vocalisation). The output of the merge operator is a regular expression that can be computed by the compile-replace algorithm of XFST. This algorithm works well for Amharic too.

define pattern "+Perf":{CVCVC} |"+Perf":{CaCVC} |"+Imperf":{CVCC} |"+Ger":{CVCC} |"+Jus":{CCVC} |"+Jus":{CCC} |"+Inf":{CCVC};

define vocalisation 0:"[e]";*

read regex "[":"∧[" 0:"{" root 0:"}" 0:".m>." 0:"{" pattern 0:"}" 0:".<m." vocalisation "]":"∧]";

list C b c g h j p q v x t y k l m n f w r z d s B C D F G H J K L M N P Q R S T W Z

list V e

compile-replace lower

Merge is a pattern-filling operator than combines a template and fillers into one network. For example for the root *sbr* the perfective stem is formed by merging the vocalisation and the root into the template CVCVC (See Figure 6.15 6.16 6.17).

After merging and compile-replacement the final network looks like the

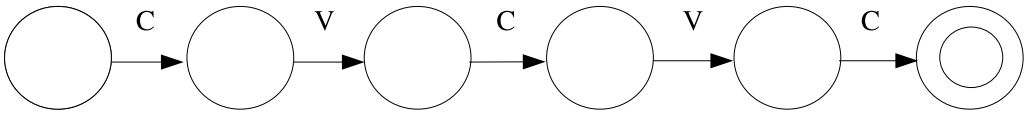


Figure 6.15: Template

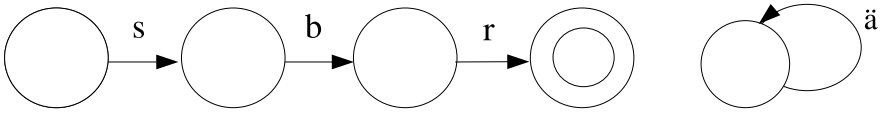


Figure 6.16: Fillers: Root & Vocalisation

network in Figure 6.17

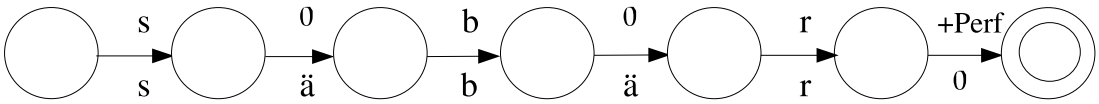


Figure 6.17: Lexical and Surface Forms

with:

Upper side: sbr+Perf

Lower side: säbär

A more straightforward approach, however, would be to simply insert vocalisation between radicals. This requires accessing positions between consonant sequences. A novel bracketing 'diacritic' convention is used to locate vowel positions and right and left contexts to discriminate between different positions (see Figure 6.18).

C is defined: List of consonants: *define C b |c |d |...*

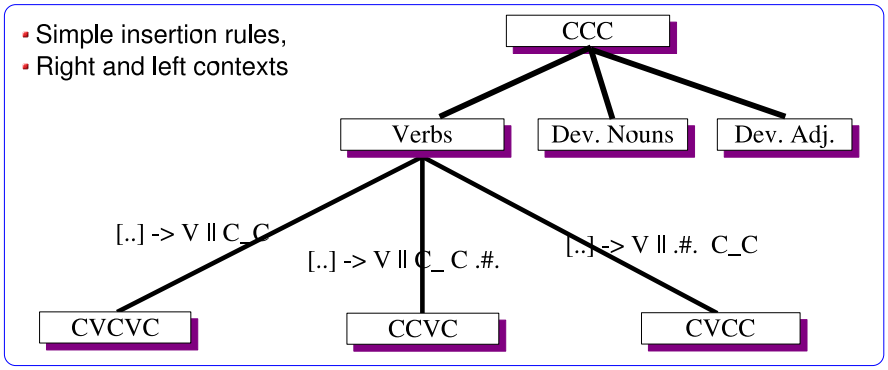


Figure 6.18: Vowel intercalation.

6.8.4 Internal changes

Derived verbs with internal changes involving penultimate consonant reduplication and vowel insertion are handled mostly by single replace rules. For example to generate *säbabär* from *säbär*, the rule used is:

$$\{b\}(->)\{bab\}||\ddot{a}\_\ddot{a}$$

which results in *säbabär*, while retaining the original underived *säbär*. The reduplication in plural of adjectives have also similar formalism:

räjm → räjajm (several tall th.): j ->jaj $||.\#.C\ V\_$

Tqur → Tquaqur (several black th.): qu ->$quaqu$ $||.\#.C\_$

säfi → säfafi (several large th.) f ->faf $||.\#.C\ V\_$

The problem in such cases, it is not possible a single rule to work on all words, so it is better to group them based on how they modify. Other very interesting reduplication processes are the full stem reduplication in collective nouns such as:

qTäl → qTäl-a-qTäl (leaves).

Reduplication of collective nouns is handled by using the self concatenation operation *word^2* which concatenates a word to itself with the compile-replace algorithm of [Beesley and Karttunen, 2003], and using a bracketing rule to find the mid position to insert the vowel.

A second method that also gives the same results is without using the compile-replace algorithm just with the self concatenation operator and a temporary file to deal with single element in the lexicon at a time to avoid over production of unwanted results. This operation demands the use of a shell script outside the Finite State Tool (Xerox Finite State Tool-XFST) (See Figure 6.19).

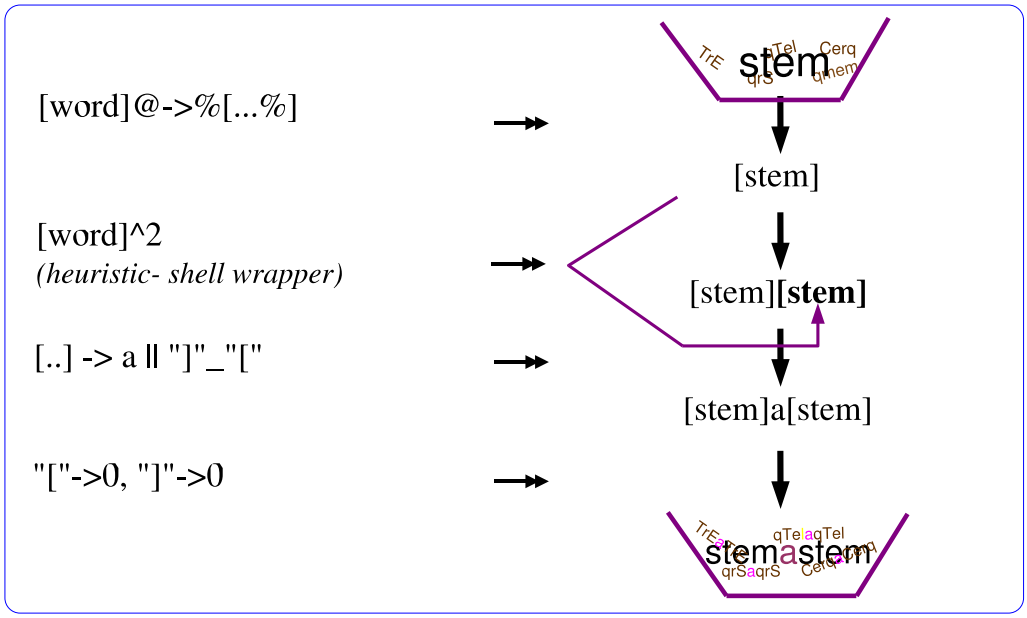


Figure 6.19: Full stem reduplication with a linker vowel.

6.8.5 Affix concatenation

The regular operation *concatenation* is used to concatenate affixes to the stem. When concatenating, illegal sequences of vowels are avoided by using replace rules and also impermissible affix combinations are controlled by introducing constraints:

[P1] [P2] [P3] [P4] [stem1|stem2|...]

[[S1|S2|S3] [S4] [S5][S6|S7]] [S8]

where P1-P4 stand for prefix categories and S1-S8 are suffix categories that a verb stem can take. Prefixes, stems and suffixes have specific positions. In case of prefixes, all categories may occur together, but no more than one from each category. There are constraints on the suffixes: [S1|S2|S3] are alternatives and cannot exist together in one word. The same is true for [S6|S7]. Similar procedures of concatenation are applied for other POS as well.

During concatenation, however, one needs to define constraints, since not all kinds of affixes are taken by every stem, particularly on verbs. Flag diacritics are used to control the constraints. Flag diacritics are extension of Xerox finite-state implementation. They are normal multi-character symbols that have a distinctive spelling. They are treated specially by application routines that are sensitive to them and treat them like epsilons when a network is applied to an input string. They do not appear in output strings. The flag diacritics provide the following functionalities:

1. Keep transducers small

2. Enforce desirable long distance constraints

3. Enable illegal paths in networks to be blocked at runtime

The most common type of flag diacritics are called U-Type (Unification flag diacritics). The syntax for writing unification flag names is :

@U.feature.value@

The feature name or the value name,

- cannot include dots
- are case sensitive
- are assigned by the developer

For example one can define flag diacritics for the nominative case and number plural as,

@U.CASE.NOM@

@U.num.plur@

The stem *fäTär* (created) takes the prefix *tä-* but not *a-*. Hence to the stem the flag diacritics *"@U.a.abs@"* and *"@U.tä.pres@"* are attached before the concatenation of the affixes.

{fäTär} *"@U.a.abs@" "@U.tä.pres@"*

{a} *"@U.a.pres@"* | {tä} *"@U.tä.pres@"*

6.8.6 Full stem reduplication

Amharic also has noun stem reduplication (with epenthetic vowel). The reduplication and vowel epenthesis operations are visualised in Figure 6.20.

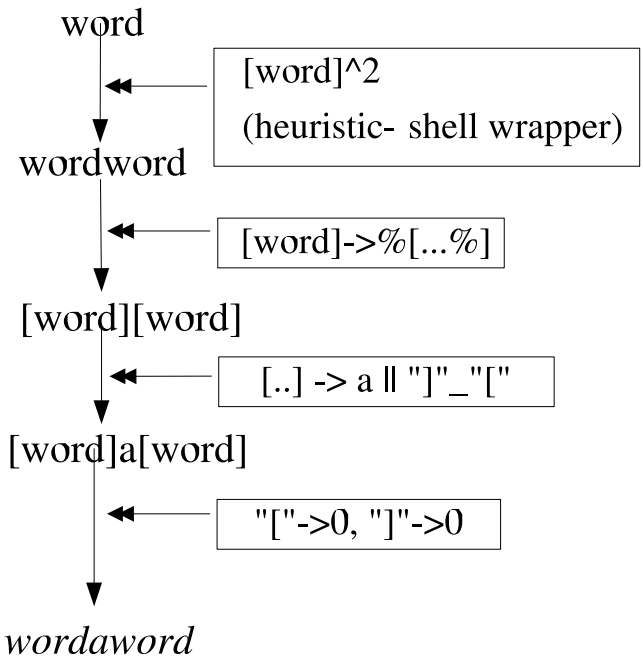


Figure 6.20: Reduplication cascade.

The required analysis procedure is handled initially outside the FS system by a shell wrapper which feeds a stream of input words to the system; the actual reduplication is then performed in the FS context. The implementation uses a novel bracketing 'diacritic' convention (not 'flag diacritic' [Beesley and Karttunen, 2003]). Formally, this is a heuristic which treats the surface lexicon as the union of singleton sets of surface forms, and applies the reduplication FST to the singleton sets individually.

6.8.7 Phonological processes

During affix concatenation, it is possible for vowels to occur in sequence that would result in a change of grapheme. To handle this problem simple replace

rules are used. For example,

$\{aa\} -> \{a\}$, replaces the sequence *aa* by *a*.

$\{ae\} -> \{aye\}$ replaces the sequence *ae* by *aye*.

Finally, palatisation was handled by a replace rule that replaces dentals with palatals:

$\{di\}(->)\{pi\}$, maps *di* to *pi* and retains *di*

$\{di\} -> \{p\}$, maps the retained *di* to *p*

(the order of operation matters)

$\{de\} -> \{pe\}$ maps each *de* to *pe*

The transducers created for each word class are finally merged by the union operation. This single transducer is then used whenever analysis of surfaces forms need to be made.

6.8.8 Romanisation

Since Amharic writing system is syllabic performing all the operations using this system makes the system quite complicated and error prone. To avoid that the compilation is done in Latin transliteration. But then interfaces on the input and output side are designed so that in the end the transducer simply consists of strings in Amharic script. This interface is created by simple replace rules of finite-states which replace an Amharic symbol into its corresponding Latin and viceversa.

6.9 Implications to the alignment system

A preliminary evaluation of the system was made by analysing words. The morphological analyser though not bad for a start had problems of over and under generation that it was possible to apply a part and not the whole of the system. The main problems were attributed to several factors among which were the generality of the theoretical descriptions of the morphology of words, far from complete base lexicon of all word classes except the verbs and the absence of a standard spelling which results in different word forms that did not abide by the rules defined (results reported in detail in [Amsalu and Gibbon, 2005b]. Hence, the task has been limitted to stripping the most salient affixes that included affixes of subject and object markers, inflections for gender number and certain prepositions and conjunctions as well as affixes of adverbial clauses. On the English side also treatment of affixes for number, tense, derived noun affixes, and adverbs has been made. Hence, we rerun the program on stems as follows,

- Store surface forms and their corresponding base forms in a database
- Redo the frequency count and score calculation
- Extract stem tanslation lexicon
- Using the stem lexicon as a lookup for surface forms rerun the chunk generator

The precision of the procedure, however, improved only by 1.6%. At a later experiment made on another data (See comparative study of different languages in an upcoming chapter) nevertheless a 10% improvement using the same shallow stemmer was obtained. However, stemming words written in a syllabic language is very difficult because once change in grapheme is created at the joining point of stems and affixes it is difficult to know if that change occurred after or before affixation. One of the reasons that there hasn't been much improvement could also be attributed to this. For this reason and for use in future activities in any natural language processing we reduced the scope into smaller subset of the language, such as considering a specific class of words and continued developing finite-state morphological analyser. Some of the efforts are reported in [Amsalu and Demeke, 2006a,b].

6.10 Summary

This part of this project work addresses the basic and first step in Natural language processing of a complex language such as Amharic. Morphological analysers are the basic and very useful tools in any natural language processing. Considering translation, understanding the structure of words and their modification to serve different grammatical functions helps to formulate the transfer of the grammar of one language to the other.

This part of the study has given the initial studies for Amharic in this direction. A a finite-state morphological analyser which addressed all parts of speech in Amharic is developed. More or less a good coverage of the

verb roots is used. There were only limited list of nouns and adjectives. Other parts of speech do not have many elements and we tried to gather them all. The transducer for verbs was so complex to develop, particularly a deeper analysis of the constraints was lacking, which then resulted in over generation. The incomplete lexicon of nouns and adjectives has also resulted in not recognising certain words.

For the alignment task we focused on using a shallow stemmer that removes the most salient affixes from both Amharic and English. The use of a shallow morphological analyser could easily be attained for operations that do not require deep analysis.

Part II

Model II

Chapter 7

Maximum Likelihood Local Alignments

In this chapter model II of our word alignment system which is procedure based on maximum likelihood estimate of translations is described. The most likely translation words in translation sentences are determined by the measure of the similarity of their distribution in the entire corpus. To improve the drop in precision which was inevitable at high recall we introduced methods of reusing of know translations list and adding a priori labelling of most likely translations based on the morphology and syntax in the given context. The results show that for the recall level obtained our procedure is quite good.

7.1 The challenges of improving recall

One of the major challenges of extracting bilingual lexical resources from parallel corpora is the difficulty to get a significant number of lexical entries as compared to the size of corpus used. In particular, when using statistical methods, the problem is exaggerated. Often a small set of words that occur with a higher frequency are addressed [Amsalu, 2006, Sahlgren and Karlgren, 2005, Ker and Chang, 1996, Wu and Xia, 1995, Kay and Röscheisen, 1993]. On the other hand, natural language processing operations such as machine translation, cross-language information retrieval, terminology banks and computer assisted language learning systems demand bilingual lexica of high coverage. Hence, attaining an acceptable coverage of lexicon is of paramount importance. Methods of using a large amount of corpora have been established to cope with the problem. But again, unfortunately, for many languages large quantities of bilingual corpora are not available.

In this chapter we propose a method of attaining increased lexical acquisition by statistical similarity measures of maximum likelihood. We use a small size of translation texts to generate many translation words. The algorithm is tested on Amharic-English bilingual texts. In Section 7.2, some features of translation sentences that are relevant for text alignment are discussed. In Section 7.3 algorithmic analysis of the optimal alignment for parallel sentences is made. How the distribution of translation terms in parallel corpora can be indicators of similarity is also presented. Section 7.4 discusses the evaluation of the results. Methods of enhancing the system are presented in Sections 7.5, 7.5.2.

7.2 Characteristics of translation sentences

Parallel sentences are two groups of sequences of words explaining the same message. The symbols in the words are not necessarily identical; neither are the lengths of each word or the number of words in the two sentences. Alignment systems try to align all or some of these words in the sentences. In many statistical approaches of alignment the chances for most of the words to be aligned is very low when the dataset used is small. In fact it is not a rare case that none of the words in a sentence may be aligned. But one thing is true, these words are translations. Ideally, each word in the source language (or the meaning contained by the word) is expressed in some way in the target language. But the question of how to find these relations is not easy to answer.

For a machine the problem is as if a human translator is expected to match a pair of sentences in languages the translator does not know and with symbols not familiar. They are just sequence of symbols, but somehow they are related. For example if we have parallel sentences and each word is represented by a single symbol, say,

1 2 3 4 (sentence I)

a b c d e f g (sentence II)

We know word 1 in sentence I, is a translation to one or more of the symbols in sentence II, but don't know which one. If we have a text with several such sentences, can we use it to guess which words are the most likely correct translations? We set out to prove if the distributional properties of

words in the entire corpus can give us information to come up with the most likely translation equivalents in sentence pairs that were extracted from the translation texts themselves. The assumption we are basing our experiment on is that *if two sentences are translations of each other so are the words in the sentences*. Subsequent Sections will give detail account of the line of argument.

7.3 Quest for the optimal alignment

Our approach tries to align words in parallel sentences given their distribution in a larger corpora. We attempt to align words in one pair of sentences. We do not want to find the translation in some other sentences in the text but within the translation sentences. Because the translation of each of the word in the source sentence is embedded somewhere just in the target sentence. If we manage to do so, the recall for the words in the shorter sentence will simply become a hundred percent.

In an m x n matrix of words in translation sentences where m and n are the number of words in source and target sentences and m is the number of words in the shorter sentence, there are $\frac{n!}{(n-m)!}$ permutations of possible alignments. Among these alignment possibilities one of them is the optimal alignment. Thus, each word in the source sentence must be aligned to its most likely translation in the target sentence. To get this alignment we need to have a measure of similarity of each word in the source and target text. This information is obtained in the the translation texts where the sentences

are extracted from.

In this work the scoring scheme we devised in Chapter 3. is used to calculate the scores. The scores for each term in the source language with each term in the target language are stored in a repository. Note that, the bilingual corpora for calculating scores is the same corpora from which the sentences to be aligned are retrieved. These score measures are used to align words in single translation sentences. Therefore, we construct a matrix of scores for each translation sentence. To get a better understanding of the argument, let us take the first translation sentences of the data used in our experiment.

Table 7.1: Similarity score matrix.

	ልጅ	ክርስቶስ	ትውልድ	የዳዊት	የኢየሱስ	መጽሐፍ	የአብርሃም
a	0.056	0.004	0.016	0.012	0.008	0.004	0.004
abraham	0.022	0.138	0.148	0.100	0.154	0.364	0.545
christ	0.131	0.864	0.095	0.171	0.214	0.077	0.077
david	0.241	0.178	0.140	0.667	0.069	0.074	0.074
genealogy	0.024	0.091	0.100	0.154	0.333	0.500	0.500
jesus	0.092	0.041	0.018	0.023	0.024	0.005	0.005
of	0.154	0.021	0.015	0.026	0.009	0.004	0.007
record	0.024	0.091	0.100	0.154	0.333	0.500	0.500
son	0.776	0.101	0.029	0.171	0.033	0.017	0.017
the	0.061	0.015	0.014	0.007	0.003	0.002	0.002

In Table 7.1, the scores of each word in the source language sentence to the words in the target language sentence is presented. This matrix is plotted in a two dimensional Cartesian plane of source words vs. scores showing the scores of the words in the source sentence with those of the target as shown

in Figure 7.1.

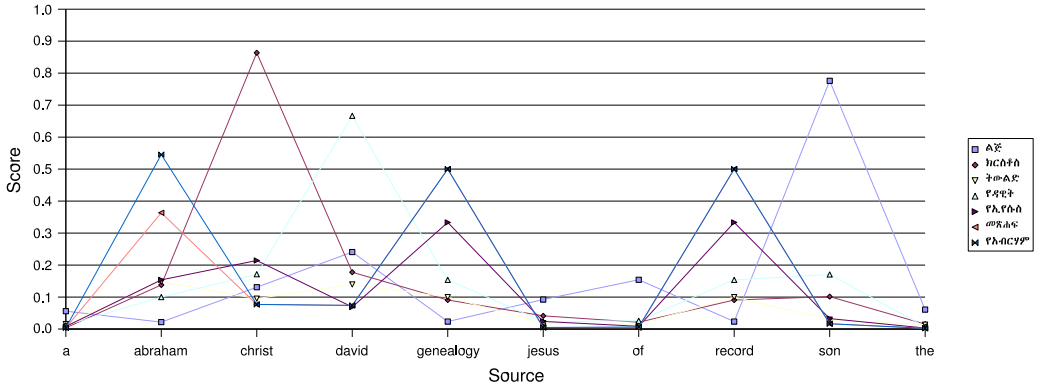


Figure 7.1: Source target mapping.

To find the optimal alignment we want to find the optimal score points for each word. If the word in the target document at the optimal point of the source word is aligned with another word with even a greater score, the optimal point for the first word will be its second high point. But again the target word at this second high point also needs to be examined if it has another point where it is the optimal alignment. This process will go on until no better point is found.

Taking the first word in the chart in Figure 7.1, we see it is aligned with its highest score with the word *son*. We also observe that *son* does not have a score greater than this score with any other word, therefore this point is where its likely translation is found. Using the same procedure we observe that the second word in the X-axis is aligned with the highest score to the word *christ*. The third word Amharic word has a best score with *abraham*, but *abraham* has even a higher score with the seventh Amharic word hence we consider the second highest point which is *david*, again *david* is aligned

with a higher score with another Amharic word, so we consider the third highest score. The search goes on until we obtain a word which does not have a higher score anywhere else. Our algorithms has thus score matrix generator (Figure 7.2), ranking routine, optimal alignment generator.

For each source term, we list out the combinatorics sorted descending according to which then we give ranks. If there are two or more target terms that give the same score they are also given the same rank, the next low score getting a rank of the rank of the previous score plus one. For this operation we sorted our combinatorics, by sentence, by source word and by score respectively cf. Figure 7.3

The ranking for each term in the source and its score list with target words. The candidates are ranked ascending from 1 to n, 1 given to the pairs with the highest score. If the second pair has also the same score as the first both are given a rank of 1 but the third entry is given a rank of 2 and not 3 and so on. The search for the translation of the first source word in our score matrix in Table 7.1 would include the search space in the tree traversal 7.4. We can also see it as a tree traversal where breadth first search is made with time complexity is $O(b^n)$.

Obviously, there will be gaps for those words that do not have high points that excel over others. In most cases that happens because some words translate morphological or syntactic phenomena rather than other words. Hence, these gaps in many cases are likely to be words which are inflectional patterns for the shorter sentence. This is true assuming that we have a perfect translation i.e. there are no deletions or insertions and the translation

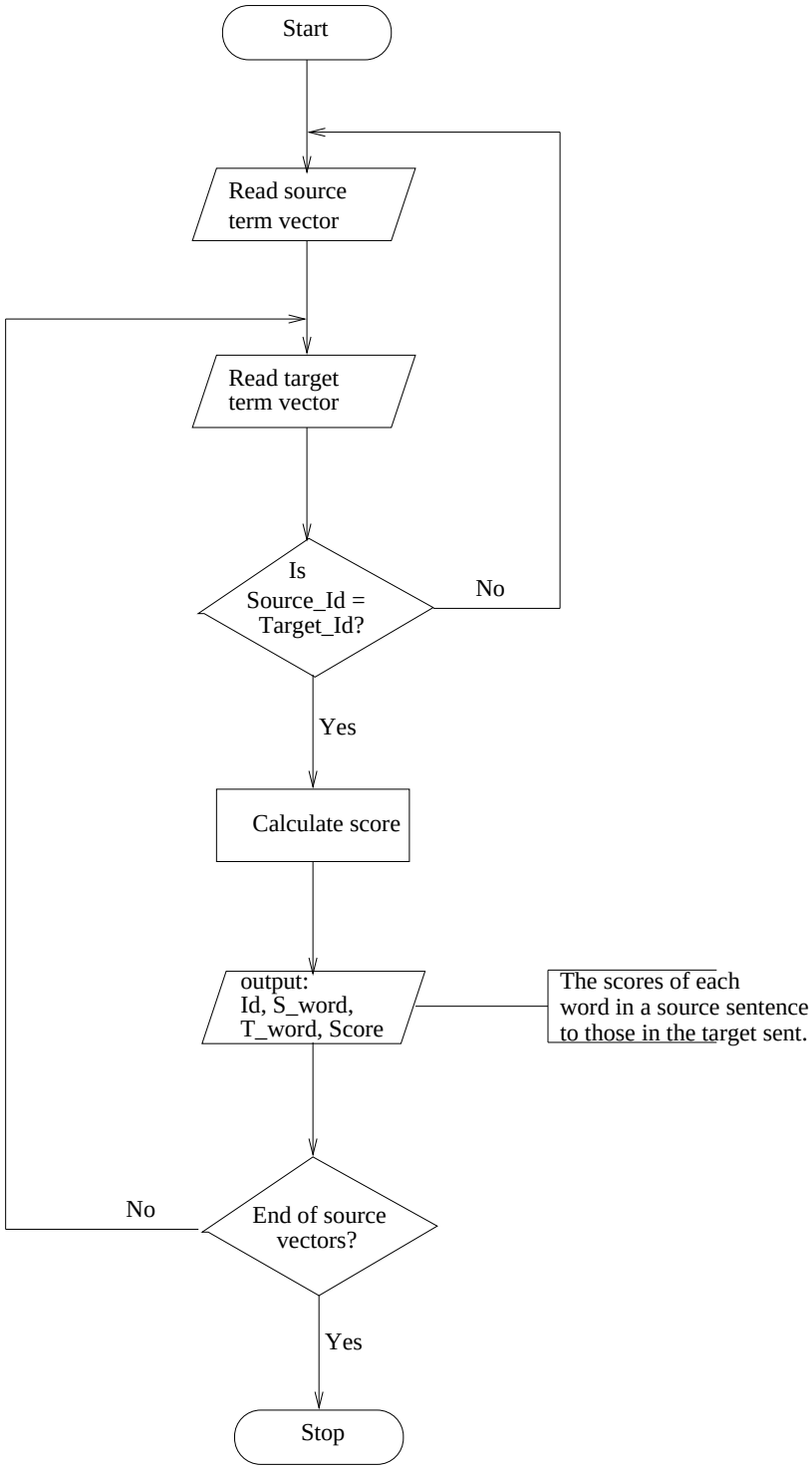


Figure 7.2: Translation sentences matrix generator.

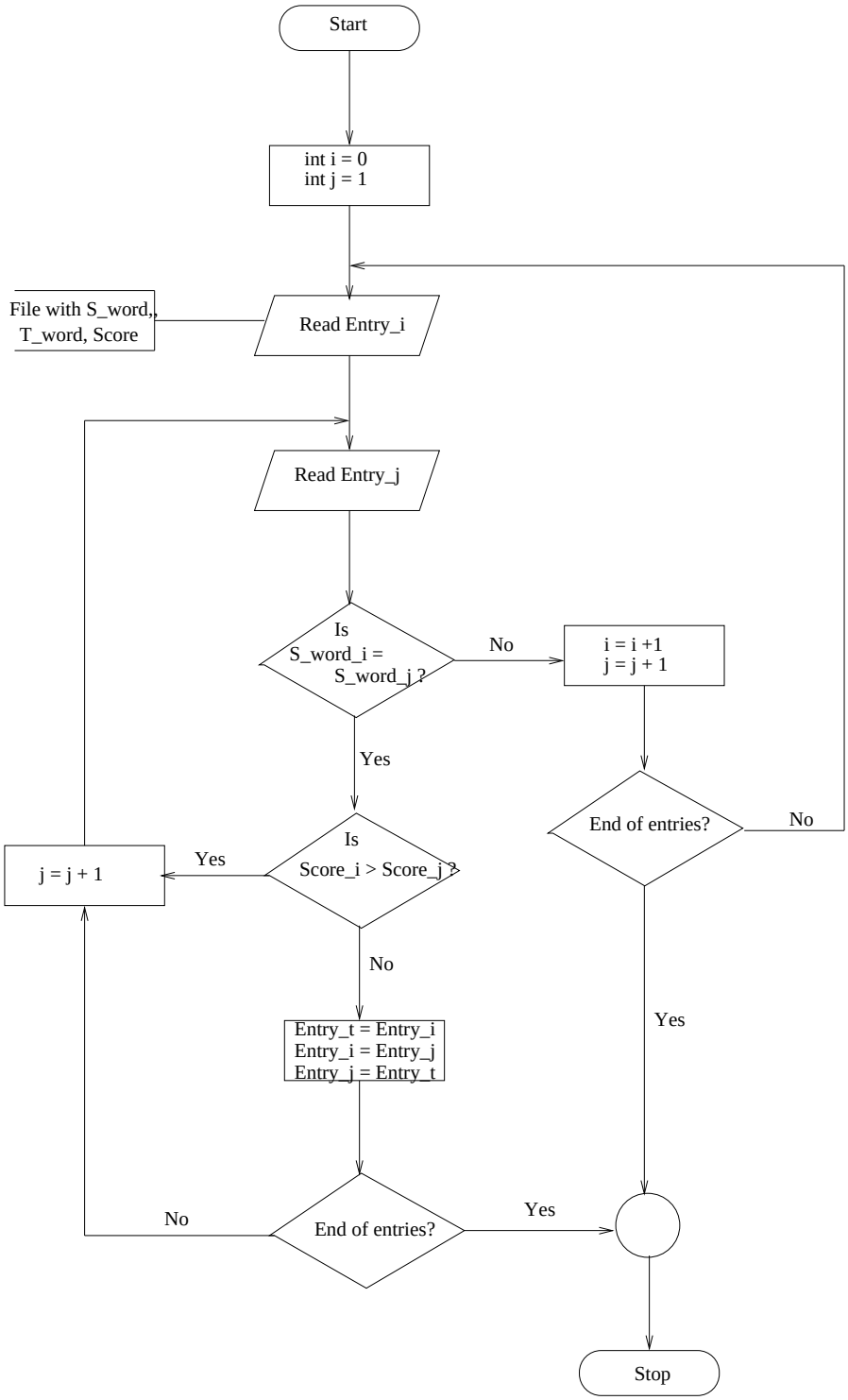
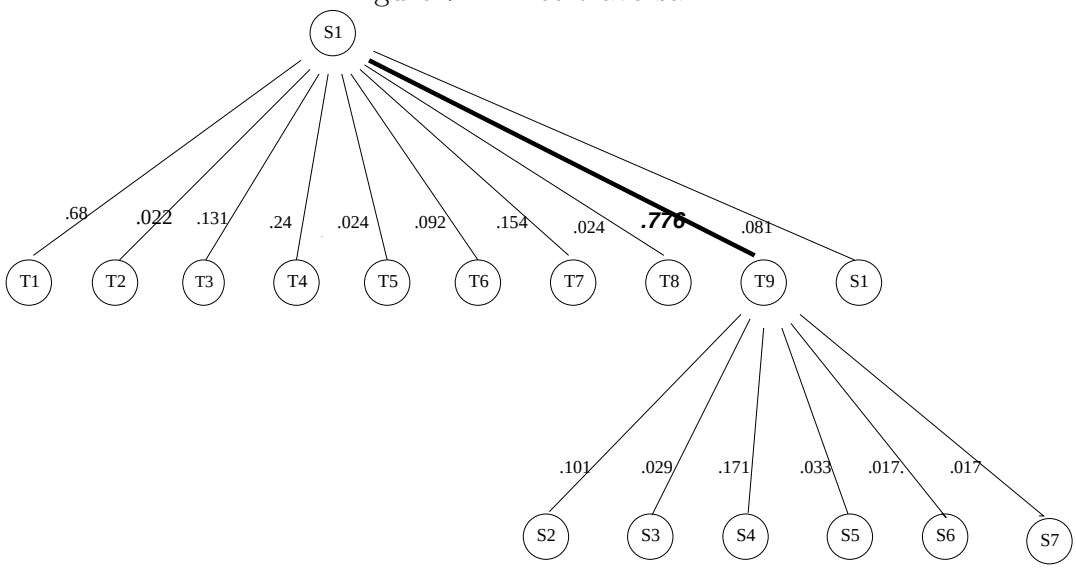


Figure 7.3: Sorting and ranking algorithm.

Figure 7.4: Tree traversal.



is accurate.

7.4 Evaluation of the results

The algorithm has been evaluated on a dataset of 1749 sentences taken from the bible, namely, the book of Mathew and Mark (the same dataset used fro global alignments in Chapter 3). From these sentences a repository containing the score of distribution similarity of each term in the source document to each word in the target document is generated. The total number of entries in the repository is 476,165 excluding those with zero score. The repository is used when searching for the optimal scores. We evaluate the proposed approach by first aligning the words, and then comparing the acquired lexica to manually compiled translations. Basically, two types of evaluation can be made.

1. Determine for how many of the sentences correct alignment has been obtained

2. Determine how many words are correctly aligned

Evaluation method 1 was not considered, because in many cases some of the words in the sentences are correctly aligned and the rest not. So one cannot get information on how far the words in the sentences have been aligned in such binary evaluation. Using method 2 however one can observe how many correct alignments we have achieved. To compute the precision obtained in evaluation method 2, since the time consumption of manual evaluation of all alignments is not affordable, a sample of alignments is randomly picked and statistical measures of confidence interval were used to project for the whole set.

Out of 76 randomly selected sentences which are constituted of 761 words, 64% of overlap with the true translations was achieved. From this we infer that at 95% confidence level the overall result has an accuracy within the confidence interval 62% - 66%. The recall values are of course 100% with respect to the language which is highly inflected.

These results are very good to attain at such a high recall level. The fact that Amharic and English are disparate languages belonging to different language groups signifies that better results might be obtained for language pairs which are closely related. Among others the scores obtained for identically inflected language pairs are more accurate.

Two directions for gaining better results are investigated in subsequently. First, we want to reuse the lexicon generated by Model I as a known transla-

tions list. This will reduce the number of alignment possibilities. Second, we try to maximise the score of certain word alignments based on recognition of morphological and syntactic patterns that allow us to decide they are more likely to be translations than others.

7.5 Enhancing Methods

As discussed in the previous sections trying to align each word in translation sentences ideally gives a broader access to the texts and hence a bigger lexicon size for a given document pair. But this is possible if only we succeed in aligning the words correctly. The experimental results indicate that the alignments have poor precision level. To improve this precision level we devised two methods of recognising possible alignments prior to the local alignment procedure. The first method is the reuse of the lexicon extracted by Model I. The second method is a rather sophisticated pattern recognition approach of maximising the scores for certain translation pairs.

7.5.1 Reuse of initial lexicon

In the global alignment method in Model I, we extracted lexicon with a relatively high precision. Model II functions independently of Model I. At this point, however, we learned that, if the lexicon generated by Model I is used as a known translation set for Model II, then it helps in excluding certain words from our search space. In translation sentences if we know one or more pairs of translations then we can exclude them and try to align the

rest. Obviously the chances of creating incorrect matches decreases at the same rate. If we have 5 words in the source language sentence and 10 words in the target language sentence, the combinatorics of possible alignments would be $\frac{10!}{(10-5)!}$. But if a pair of words are known translations, the combinatorics decreases to $\frac{9!}{(9-4)!}$, which is basically 10 times smaller and if we know a second additional pair the search would decrease to $\frac{8!}{(8-3)!}$ which is 10*9 smaller. If the length of the shorter sentence is m words and that of the longer is n, for every known translation pair k, the decrease in the search space is $\frac{n!}{(n-k)!}$. Of course the issue of doing this in several iterations is a worth trying operation. Meaning, taking the most likely translations and including them in known translations list and then realigning the rest or merging Model I and Model II. The pseudocode for this operation is:

read words in translation sentence
if there is a word in source sent
    if its translation is in target sent
        exclude the words

Eventhough this is theoretically sound, in practice since the size of lexicon generated by Model I is minimal, it does not decrease our search space to a large extent. For this reasons we considered a complementary routine that maximises the scores of certain translation pairs than others.

7.5.2 Pattern recognition approach of maximising scores

In this routine we involved the use of the morphosyntax of both languages as a basis for guessing necessary translations with a higher probability. For

resource-poor languages such as Amharic where large quantities of text are not available, it is crucially important that all possible methods of getting the most out of a limited size of corpora are explored and exploited. Patterns of text that occur with fairly uniform syntactic structure are used as features for classifying translation terms throughout the text. Our method is simple but quite effective in the quality of the translation information it gives. The rules accompanying the method are language specific.

Description of the methods

Our approach depends on words that show regular syntactic appearance in Amharic-English bilingual texts. These words are used as features for identifying translation pairs that are associated with them. Our procedure follows standard pattern recognition algorithm in which three basic steps are involved,

- Gathering observations
- Feature detection
- Classification

Sensors gather observations, which are basically words in sentences. These sentences are known translations of each other. A feature extraction algorithm attempts to extract features from the observation. Our classification scheme does the actual job of classifying or describing observations, relying on the extracted features. The system is a supervised learning system with a

priori labelling of patterns, which are the training sets for training the classification scheme. The classification scheme is a structural pattern recognition system based on the structural interrelationships of features.

Priori labelling of patterns

The feature detector is based on a priori knowledge of patterns that help to classify our observation. Determining these features that can discriminately classify our observations was a challenge we had to solve in the beginning. In other words the priori knowledge was developed empirically.

The features that we believe will give us enough information to classify translations are basically the words which have morphosyntax that can consistently be matched in the two languages. We made experiments to find out which features are discriminative enough than others. A similar work to this one is a program for the automatic alignment of parallel texts, originally developed for the English-Norwegian Parallel Corpus (ENPC) [Hofland and Johansson, 1998, Hofland, 1996]. They focused on language-specific information, in the form of so-called *anchor words* or a simple bilingual lexicon referred to as the *anchor list*. We also use such patterns in both languages which have regular syntax and morphology. The alignment program makes use of some parameters, as explained below. The considerations in selecting the anchor list are:

1. The words had to be reasonably frequent
2. They had to have fairly straight forward equivalents in the two languages.

Some words were included even though the words were not very frequent, provided that a good correspondence could be established. Some of these anchor words which we call also signals are described next. We focused on words in closed classes: prepostions, auxiliary verbs, adverbs and conjunctions.

i. Prepositions

A Preposition is a word used to express some relation of different things or thoughts from each other. If we consider their syntax in sentences, in many cases they have a fixed number of ways of using them in both languages.

One such preposition is the prepostion *with*. In English *with* explains the togetherness of some objects or events. One of the objects or events that is together comes after the prepostion *with* syntactically, and in no other form unless perhaps it has another application other than the one described here. In Amharic such kind of togetherness is represented often by the use of a prefix *kä–* attached to the object or event and another word *gar* that comes just after it. The prefix *kä–* can occur serving other purposes too. But the word *gar* cannot be used for any other application. Hence, if a word with the prefix *kä–* is followed by *gar*, it can only be equivalent to the word that comes after *with* in the English. Let's see examples:

(1) Sara ate dinner *with Thomas*	Sara *käThomas gar* rat bälach (Sara with Thomas dinner ate)
(2) Sara ate dinner *with her friend*	Sara *käguadegnawa gar* rat bälach (Sara with her friend dinner ate)
(3) Sara ate dinner *with her*	Sara *kärsua gar* rat bälach (Sara with her dinner ate)
(4) Sara ate dinner *with Thomas and Elisabet*	Sara *käThomasna käElisabet gar* rat bälach (Sara with Thomas and Elisabet dinner ate)
(5) Sara cuts the bread *with a knife*	Sara dabown bäbilawa qoräTäc (Sara the bread with a knife she cut)

Forms like (1) are easy to extract, we simply take the word next to *with* and align it to the word which has a prefix *kä* and followed by the word *gar*. If we take example (2) it becomes a bit complex, because the whole prepositional phrase should be taken into account in which case the correct alignment is $herfriend$ with *guadegnawa*. To resolve this problem we extended our procedure to chunk noun phrases that come after the preposition. But if we have a structure like in (3) considering the next word after *her* would be a mistake since the correct alignment is *her* with $rsua$. We did not use any external parser to chunk these noun phrases. Eventhough we did not consider it here, punctuation marks such as a comma, a semicolon or end of sentence boundary could help us know how far to go. In example (4) we find two nouns. On the Amharic side the prefix *kä* attaches to both

nouns. We relaxed our routine to search to the left of the word preceeding *gar* until a word without the prefix *kä* appears. In the same way we move to the right of *was* in the English sentence equal number of steps that we moved to the right in Amharic. But this count would not work if we count words such as *and*. Hence, in counting we exclude function words such as *and*. Assuming the order is not reversed, i.e. *with Thomas and Elisabet* is not translated as *with Elisabet and Thomas*, the correct alignment would be in the sequence written, first word to first word. Of course as the sentences get longer and complexer the chances of making failure increase. The use of *with* in constructions such as (5) have a completely different form. But since our procedure searches for matching forms in both sentences, when it doesn't find the equivalent form in one of the sentences it simply jumps to the next sentence.

To find out if this generalisation works we made an experiment by trying to access such occurrences and align the words or chunks associated in the data source we used in this study. Table 7.2 gives as an excerpt of the output. The data source used for evaluation consists of parallel corpora with 20,347 Amharic and 36,537 English words, accommodated in 1749 aligned sentences.

In Table 7.2 (a) shows as an instance of two occurrences in one Amharic sentence while there is only one match in the translation sentence in English. (b) indicates a correct match in which there are multiple nouns. (c) show a wrong match due to (a). In (d) there is an extra *in* which was encountered due to case (3) in our example sentences. In (e) the part *andsinners* was not included in our extract though that was the correct match. This fail-

Table 7.2: The use of *with* as an anchor.

Amharic (ከ_... ጋር)	English (with ...)	Remark
ከእኛ ጋር	with child	X
ከእኛ ጋር	with us	✓
ከእርሱ ጋር	with him	✓
ከእናቱ ከማርያም ጋር	with his mother mary	✓
ከእናቱ ከማርያም ጋር	with gifts of gold	✓
ከአባታቸው ከዘብዴዎስ ጋር	with their father zebedee	✓
ከባላጋራህ ጋር	with your adversary	✓
ከባላጋራህ ጋር	with him	✓
ከእርስዋ ጋር	with her in	
ከእርሱ ጋር	with him	✓
ከአብርሃምና ከይስሐቅ ከያዕቆብም ጋር	with abraham isaac and jacob	✓
ከእኛ ጋር	with us	X
ከኢየሱስና ከደቀ መዛሙርቱ ጋር	with him and his disciples	✓
ከቀራጮችና ከኃጢአተኞች ጋር	with tax collectors and sinners	
ከእነርሱ ጋር	with them	✓

(a) (b) (c) (d) (e)

ure occurred because *taxcollectors* is one word in Amharic and it disturbed number of moves to be made form the anchor.

The over all result gave out of 118 occurrences 74% were correct. This is based on a very strict evaluation where any partial matches are considered as false alignments.

Another very useful word for our purpose is again a prepostion, namely *on*. When *on* is used in a context of placing something on some surface or on something, it is often if not always expressed in one form in Amharic. The surface or the thing will have the prefix *bä–* and the word *lay* follows. Example would be:

(1) They live *on a hill*	*bätä lay* ynoralu (on a hill they live)
(2) Let's discuss *on this issue*	*bägudayu lay* inwäyay (on the issue let's discuss)
(3) The pen is *on the table*	skrbitow *Tä räpezaw lay* näw (the pen on the table is)

Case (1) and (2) are the ones that we tried to capture. Form (3) exists though its frequent use may depend dialectically. In any case the results obtained are such that, there were 61 appearances out of which 87% are correct.

ii. Adverbs

The adverb *after* is used to describe that an event occurred after some other event occurred or in similar contexts. In English the words that describe the event that happened first come after the term *after*. These terms could be single or multiple but are often terminated by a punctuation mark mostly a comma or a period. Its counter part in Amharic is described by a prefix *kä–* attached to the first word of the terms that indicate the event that occurred first and a terminating word *bähuala*. Our lexical target here is to map the term(s) that indicate the event that occurred first. Here, we need to notice that this approach is even vital for some systems such as machine translation systems that often require bilingual lexica of above word level. Some examples would be:

(1) *After it rained*, the earth became wet	*käzänäbä bähuala* märetu räTäbä
(2) *After lunch*, coffee is served	*kämsa bähuala* buna yqärbal
(3) *After Sara and Hellen*, Dawit is born	*käSarana käHellen bähuala* Dawit täwälädä

after is quite regular. It has also extended forms such as in (3). A similar procedure as in the case of *with* is used for such forms. Note that the prefix *kä* repeats on both objects (Sara and Hellen). For the anchor *after*, out of 24 appearances 75% were correct matches.

The word *when* on the other hand is followed by an event which marks a certain time or a cause for a certain output. In Amharic such situations can be expressed in many ways most of which do not have any fixed rules that would be occurring often. But there is a form if it occurs is quite regular, that is *bä– ... gize*. The time event or the case defined occurs in between *bä–* as a prefix of the first word from the left and the word *gize* from the right.

(1) *When Thomas came* *Thomas bämäTa gize*

The matching we seek here is *came* with *mäTa*. But this forms are frequent depending on dialect. In our bible data there were 81 appearaces out of which 57% were correct. Obviously this is not a good anchor.

iii. Conjunctions

The most salient conjunction *and* also occurs in a fixed context in that in English it appears between two words that may be items, ideas, names, etc., that have some common property. Our target translations are the word(s) to the left, and to the right of the conjunction. In Amharic the equivalent

form rarely occurs as a free morpheme. Often it occurs as a suffix (*-na*) to the first word to be connected followed by the other word. The occurrences of the conjunctions at the end or at the beginning of sentences are excluded from the analysis, because they do not manifest within the context of our generalisation of regularity.

(1) *Sara and Ellisabet* went to the USA	*Sarana Ellisabet* wädä USA hedu
(2) Since her exam results were not good and she was not able to improve them, she failed	-

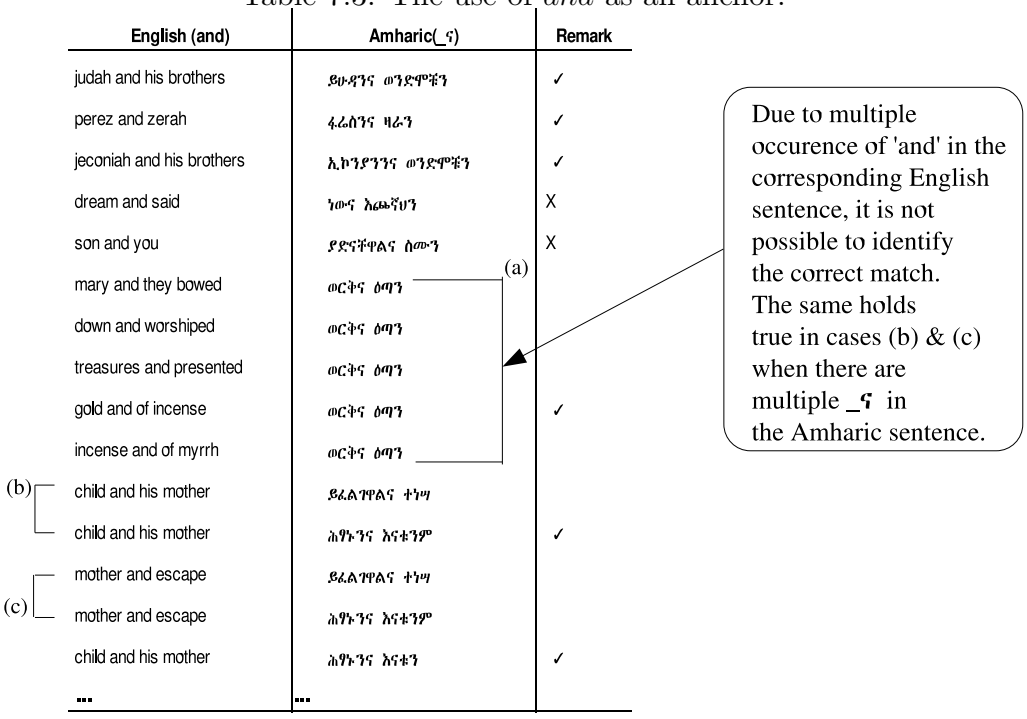
Assuming the order (of Sara with respect to Ellisabet) is not reversed the matches we obtain in (1) will be in the same order, first word to first word. Bust cases such as (2) are too complex to deal with. One can give many examples of different cases for the use of *and*. Table 7.3 shows an excerpt.

One can observe repeated occurrences of *and* in a single sentence specially on the English side. Such cases are difficult to extract even if their equivalents exist in the second language. This is because we do not know which one translates which and it is not advisable to depend on the order they occur, since that would not be the case always. The preliminary results demonstrate that the anchor *and* produced 24% correct and 23% partial matches in a total of 894 appearances.

iv. Auxiliary verbs

Other features we have considered examining are the auxiliary verbs *is* and *was*. Both *is* and *was* come often just preceeding a main verb and some-

Table 7.3: The use of *and* as an anchor.



times are followed by a noun as well in English. In Amharic the equivalent forms *näw* and *näbär* respectively appear following a main verb. Since the order of Amharic is SOV, we expect them at the end of the sentence preceeded by the main verb. For *is* and *was* also some initial results of 49% out of 86 hits and 33% out of 33 hits respectively have been correctly extracted.

7.5.3 Discussion

There are cases where wrong signals are generated and multiple signals are also generated and is difficult for the sensor to analyse all the observations. For example the signal *and* can exist in many contexts than we predict and there could be more than one *and*s in a single sentence or observation. The

other pattern that gives wrong results was also the use of *when*. *When* normally signifies a time when an action took place. The word that could potentially be a good translation indication is a verb, some time the verb comes after when but there are cases where the subject of the actioncomesfirst and the verb follows. At this time unless we have a way of identifying the subject as being not the verb then we fail. In the case of and more than 50 percent of the cases are wrong, about one third was correct.

Most of the failures in the module for *when* are inability to detect correct boundaries (particularly in the English sentences) and the effects of alternation in vowel clusters formed by the attachment of the prefixes on Amharic to words that begin with vowels. The major failures for *and* were that *and* being a very salient word, it occurs quite frequently for various uses and often even with repeated appearances in a single sentence in spite of the fact that the two languages have different word order. That filters of function words are not introduced in this part of the module also had additional adverse effects of having a significant number of partial matches. The consideration of the left side context for the auxiliary verbs could also possibly improve the results to some extent. In general the outputs are very good. Better results could be obtained by improving the rules, specially in finding more precise methods of bounding the right and left ends of the chunks of translations, and also better ways of discriminating among multiple occurrences in individual sentences need to be devised. Better approaches of dealing with function words that create a lot of noise is also recommended.

7.6 Implications to local alignment of words

Given translation sentences being able to find such features that would give some hint on the translations help to reduce the number of non-aligned pairs. Lets look how our example translation sentence would be aligned in this context.

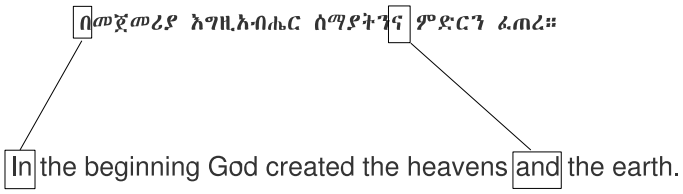


Figure 7.5: Feature extraction.

In Figure 7.5 we have two features, *in* and *and*, which occur only once. On the Amharic side prefix corresponding to *in* is observed in only one word and no other. The suffix corresponding to *and* is also occurring at only one position. Here we could easily conclude that the prepositional phrase *in the beginning* is translated by በመጀመሪያ and *the heavens* and *the earth* translate to ሰማያትን and ምድርን respectively. One could argue that *the heavens* could have been translated to ምድርን and *the earth* to ሰማያትን . Yes, that is possible but the probabilities for such translations would be lower. In our case we chose to increase the weights for aligning such features rather than putting them into a known translations list. That is if the translation probability P(*in the beginning*,በመጀመሪያ) was p then we raise it to p*1.3 to only increase its chance of matching. So in the example transaltion sentences with the help of our chunk parser and the signals generated by the features we have

high chances of matching.

in the beginning	በመጀመሪያ
the heavens	ሰማያትን
the earth	ምድርን

We are then left with choices of two words on each side of the language. The second and last word on Amharic side and the word *God* and the word *created* on the English side. In Amharic the last word in a sentence is a verb. A simple feature such as an *-ed* ending of the word *created* or the initial capital in *God* allows us to match created with the last word of the Amharic sentence. The potential of this approach is so big. A thorough study of these rules needs to be done in the future.

7.7 Aligning verbs

Another attempt that has been made using another approach was to align verbs. The method we used can be summarised,

- Develop a chunk parser for Amharic verbs
- Use external dictionary to spot English verbs

The assumption that led us to the use of this method is that in a source language sentence, if we identify the verb and we also spot the verb in the target language translation sentence, then it is most likely the case that these verbs are translation of the language.

There was no need of developing a sophisticated parser for Amharic. At least we did not think so in the beginning. This is because Amharic verbs are place at the end of the sentence. An example,

(1) *Sara misa bälac* (Sara lunch ate)

In such a case one simply needs to capture the last word. Now, if there existed an auxiliary verb the auxiliary verbs comes at the end preceeded by the main verb.

(2) *Sara misa iyäbälac näbär* (Sara lunch eating was)

Auxiliary verbs in any case are very few so we extended our chunk parser to accommodate this case. That is when there is an auxiliary verb take the word preceeding it. On the English side we simply used an online dictionary of English verbs to spot the verb in the sentence.

The experimental results however were not good. Let's look at sentences from our data (See Table 7.4)to understand what happened. The underlined words are identified by the alignment system as verbs.

There are several points to see:

- Too many complex (very complex) sentences
- sentences do not necessarily have main verb
- Sentences in the data source we used (the bible) do not necessarily abide by the rules of grammar of the language
- The verbs may serve other purposes than a verb in the context

Even if we removed the modal and auxiliary verbs we still have the complexities. It is even difficult to imagine a parse tree resolving this problem.

Table 7.4: Verb spotting.

17	አንግዲህ ትውልድ ሁሉ ከአብርሃም እስከ ዳዊት አሥራ አራት ትውልድ፥ ከዳዊትም እስከ ባቢሎን ምርኮ አሥራ አራት ትውልድ፥ ከባቢሎንም ምርኮ እስከ ክርስቶስ አሥራ አራት ትውልድ ነው።	Thus there were fourteen generations in all from Abraham to David, fourteen from David to the exile to Babylon, and fourteen from the exile to the Christ.
18	የኢየሱስ ክርስቶስም ልደት እንዲህ ነበረ። እናቱ ማርያም ለዮሴፍ በታጨች ጊዜ ሳይገናኙ ከመንፈስ ቅዱስ ፀንሳ ተገኘች።	This is how the birth of Jesus Christ came about: His mother Mary was pledged to be married to Joseph, but before they came together, she was found to be with child through the Holy Spirit.
19	አጩዋዋ ዮሴፍም ጻድቅ ሆኖ ሊገልጣት ስላልወደደ በስውር ሊተዋት አሰበ።	Because Joseph her husband was a righteous man and did not want to expose her to public disgrace, he had in mind to divorce her quietly.
20	እርሱ ግን ይህን ሲያስብ፥ እነሆ የጌታ መልአክ በሕልም ታየው፥ እንዲህም አለ። የዳዊት ልጅ ዮሴፍ ሆይ፥ ከእርስዋ የተፀነሰው ከመንፈስ ቅዱስ ነውና አጩኛህን ማርያምን ለመውሰድ አትፍራ።	But after he had considered this, an angel of the Lord appeared to him in a dream and said, "Joseph son of David, do not be afraid to take Mary home as your wife, because what is conceived in her is from the Holy Spirit.
21	ልጅም ትወልዳለች፤ እርሱ ሕዝቡን ከኃጢአታቸው ያድናቸዋልና ስሙን ኢየሱስ ትለዋለህ።	She will give birth to a son, and you are to give him the name Jesus, because he will save his people from their sins."
22,23	በነቢይ ከጌታ ዘንድ። እነሆ፥ ድንግል ትፀንሳለች ልጅም ትወልዳለች፥ ስሙንም አማኑኤል ይሉታል የተባለው ይፈጸም ዘንድ ይህ ሁሉ ሆኖአል፥ ትርጓሜውም። አግዚአብሔር ከእኛ ጋር የሚል ነው።	All this took place to fulfill what the Lord had said through the prophet:
		"The virgin will be with child and will give birth to a son, and they will call him Immanuel" which means, "God with us."
24	ዮሴፍም ከእንቅልፉ ነቅቶ የጌታ መልአክ እንዳዘዘው አደረገ፤ አጩኛውንም ወሰደ፤	When Joseph woke up, he did what the angel of the Lord had commanded him and took Mary home as his wife.
25	የበኩር ልጅዋንም አስክትወልድ ድረስ አላወቃትም፤ ስሙንም ኢየሱስ አለው።	But he had no union with her until she gave birth to a son. And he gave him the name Jesus.
26,27	ኢየሱስም በይሁዳ ቤተ ልሔም በንጉሡ በሄሮድስ ዘመን በተወለደ ጊዜ፥ እነሆ፥ ሰብአ ሰገል። የተወለደው የአይሁድ ንጉሥ ወዴት ነው? ኮከቡን በምሥራቅ አይተን ልንሰግድለት መጥተናልና አሉ ከምሥራቅ ወደ ኢየሩሳሌም መጡ።	After Jesus was born in Bethlehem in Judea, during the time of King Herod, Magi from the east came to Jerusalem
		and asked, "Where is the one who has been born king of the Jews? We saw his star in the east and have come to worship him."
28	ንጉሡ ሄሮድስም ሰምቶ ደነገጠ፥ ኢየሩሳሌምም ሁሉ ከእርሱ ጋር፤	When King Herod heard this he was disturbed, and all Jerusalem with him.

7.8 Summary

This chapter documents an alignment algorithm that tries to make a local alignment of words in a sentence to the words in its translation. It makes an estimation of the most likely translation for each word in the shorter sentence and aligns them all. The criteria for choosing the most likely translation is based on the similarity scores obtained from the whole corpus. Since at such a high recall, it is not possible get high precision, methods of enhancement that reduce wrong alignments are introduced. These enhancement methods are the reuse of known translations list generated by Model I and the use of morphological and syntactic features that maximise the scores of the most likely translations are used.

This approach of aligning texts is a new approach. Since it is just at its early state, more could possibly be done by for instance increasing data size to get more precise score values, introduction of a morphological analyser and alignment at stem level and increasing the scope of priori labelings.

Chapter 8

Evaluation

Model I and Model II along with the chunker, shallow stemmer and enhancement methods have been evaluated on a data taken from the bible, the books of Matthew and Mark. These books together consist 20,347 Amharic and 36,537 English tokens, which encompass 6867 and 2613 types in Amharic and English respectively. It is to be recalled that the evaluation of each algorithm is reported in previous chapters. This chapter describes other methods of evauating the Models: performance comparison accross different language pairs, application to noun identification and comparison with a well known alignment system.

8.1 Comparative Study

We tried to compare the performance of our alignment Model I in four languages. Our motivation is to compare what the impact of the difference in typologies would have on the performance of a statistical word alignment

system. The languages considered are: Amharic, English, Hebrew and German.

Hebrew is a Semitic language of the Afro-Asiatic language family. Hebrew grammar is partly analytical, expressing such forms as dative, ablative, and accusative using prepositional particles rather than morphological cases. However, inflection plays a decisive role in the formation of the verbs, the declension of prepositions, and the genitive construct of nouns as well as the formation of the plural of nouns and adjectives. English on the other hand is a West Germanic language which is a branch of the Indo-European family of languages. English grammar displays minimal inflection compared with most other Indo-European languages. It lacks grammatical gender and adjectival agreement. Case marking has almost disappeared from the language and mainly survives in pronouns. At the same time as inflection has declined in importance in English, the language has become more analytic, and developed a greater reliance on features such as modal verbs and word order to convey grammatical information. German is also a West Germanic language. It is usually cited as an example of a highly inflected Indo-European language. It is an inflected language. German nouns inflect for gender, number and case. Verbs also inflect for person, number, mood and aspect. Both German and English are written in Latin alphabet.

In our evaluation data which is the book of Genesis in the bible we clearly see the differences in complexity across these languages (Table 8.1).

In the statistics of the words in terms of tokens, there are slight indication that Amharic is the most complex followed by Hebrew. German is again more

Table 8.1: Evaluation data

-	Amharic	Hebrew	English	German
Running words	20,836	20,613	35,331	35,906
Unique words	7,167	5,143	2,829	3,578

complex than English. When we look at the number of types, we observe a significant difference. Amharic is the most complex followed by Hebrew. There is also an indication that German is more complex than English but less complex than Hebrew.

These texts went though the necessary preprocessing steps. The comparison has been made under the same thresholds of frequency (sum of frequency of candidates) greater than 9 in all cases and score threshold greater or equal to 0.55.

Source	Target	Correct	Wrong	Compound	Total	Precision	Recall
Amharic	English	112	37	7	154	71.4%	4%
Amharic	Hebrew	190	26	5	221	86%	3.7
Amharic	German	143	33	10	186	82.3%	4%
Hebrew	German	252	43	4	299	84.3%	7%
Hebrew	English	202	31	4	237	85.2%	7.1
German	English	341	15	2	358	95.3%	12.1%

Table 8.2: Alignment results

Note: With the use of a shallow stemmer the Amharic-English precision

improved by 10%.

8.2 Enriching mono-lingual text from bilingual corpora

We developed a system that maps German nouns to their translations in Amharic. We took advantage of the spelling of German nouns which always begin with a capital letter to identify the nouns in Amharic which do not have special marking. There is no upper or lower case representations in Amharic. All cases are the same either for certain class of words or for letters at the beginning of sentences. Nouns take affixes such as plural markers, clitics for gender or case. However, those clitics are also taken by adjectives. Hence, it is difficult to recognise them unless we have efficient syntactic parsers that take a sentence and parse the words based on the grammar of the language. Therefore, we took a step in which we could use German translations of Amharic text to make a partial parsing of nouns which in many cases would also be noun phrases since the words in Amharic are complex. However, the complexities of Amharic nouns are much lower in comparison to that of verbs which keeps us being encouraged on getting better results.

The obvious question one would be asking is if there are many Amharic-German translation texts? We sought for possibilities of using documents produced by German radio Deutshewelle Amharic program which produces news and other programs every day in Amharic.

Our procedure of alignment is identical to the global alignment in Chapter

3, except that here we only consider nouns.

1. Normalise capitalisations at the beginning of sentences
2. Generate a lexicon of only of the words in the German text which begin with a capital letter
3. Align these words with their most likely translation

8.2.1 Preprocessing German text

The preprocessing step was a bit complex than the usual operations of removing punctuation marks and tokenising. The major task was dealing with capitalisation. In our previous work we lowered every uppercase to lowercase. We certainly do not want to do that here, since our objective is to align nouns which are known for their initial capital in German. But then we have another problem to solve, because sentences begin with capital letter at the beginning of sentences regardless of whether they are nouns or other words. On the other hand the beginning of sentences could also be nouns. In the German bible text we actually even found that words after ; and : also begin with an initial capital. Hence, we developed a routine that handles these problems. The algorithm does the following three operations.

1. generate a list of words with initial capital when not at the beginning of a sentence, a clause or a list (See Figure 8.1)
2. for every other words check if they are in the list in 1, if not disregard them

One could ask the question, why not simply exclude the words with initial capitals when not in the interior of sentences. The reason is simple; because we are dependent on the frequency information we obtain from words in text.

In this routnine we have two output:

1. List of nouns in the interior of sentences

2. list of words capitalised words at the beginning of sentences and after ;s and :s

The next thing we do is identify nouns from the initial capitalised word list. For that we did not use any external source such as noun dictionaries. Rather we checked if these words also exist in the inner part of sentences in the corpus itself. Searching in external noun dictionaries is not necessary in this case because any way if they do not have multiple appearance in the document, they are low frequency words and do no not add much if at all they do to the results of our computation. See Figure 8.2 for the routine that spots nouns at the beginning of sentences and clauses.

8.2.2 Aligning nouns

Given the list of nouns in the German side of the translation, we try to search the best translation for each one of them in the Amharic document. Model I of our alignment system was used to run the rest of sorting, frequency count and similarity comptation and alignment. We also considered only the first best match. The results showed that for frequency of words greater than 3 the precision rate is greater than 80%.

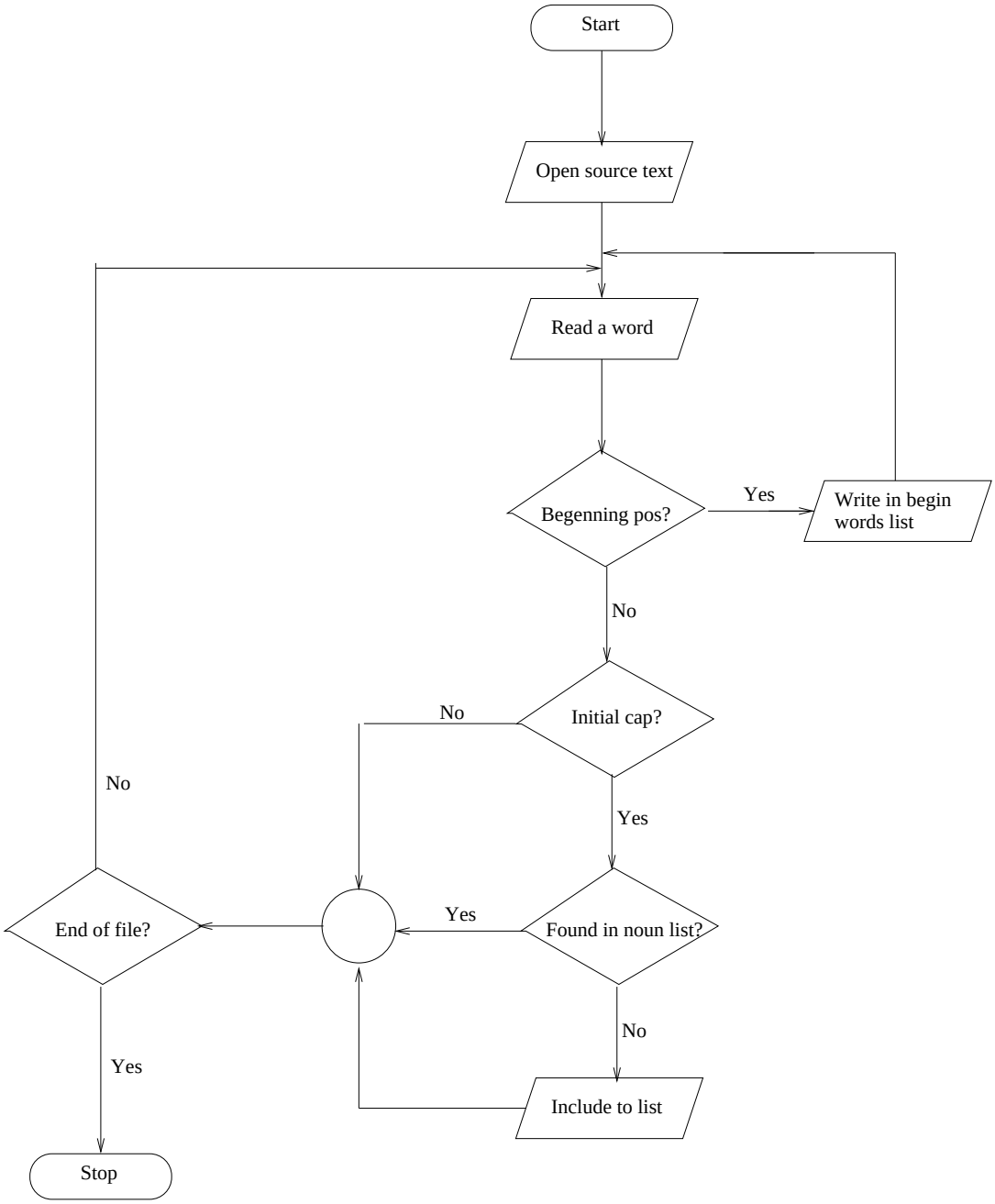


Figure 8.1: German nouns in interior of sentences

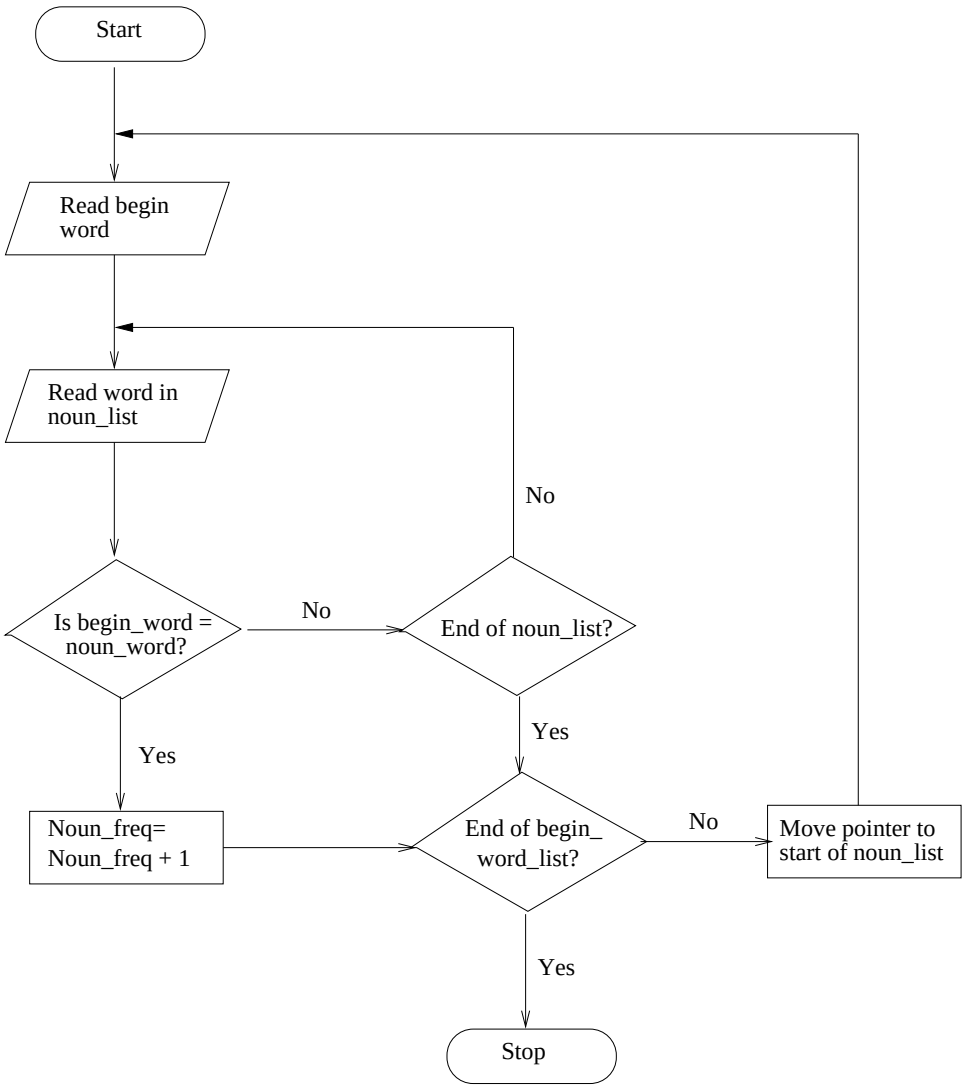


Figure 8.2: Spotting nouns

8.3 Comparison of Model II with GIZA++

Our system had to be compared to the state of the art systems, to find out how well it performs. Todate the alignment system which has got high recognition as the most successful is GIZA, an implementation of the IBM statistical machine translation algorithm [Brown et al., 1993]. GIZA++ an extension of GIZA. GIZA was first developed by SMT group at the Center for Language and Speech Processing, John-Hopkins University in 1999. Later an extended version that includes an implementation of the IBM-3, IBM-4, IBM-5 alignment models and also HMM alignment model was developed by Franz Josef Och [Och and Ney, 2003]. The alignment model takes as an input at least three files: The first two consisting of words with their frequency and a unique identification for the source and target documents and a third file consisting of the source and target translation sentences, words replaced with identification numbers and the frqency of those translations.

We got an output of sentence pairs and their alignment which was n:m. To evaluate the system we used the same data source on which alignment Model II in this project was evaluated and we also used the same procedure: select certain number of sentences (in our case 76 out of a total of 1749) randomly. We used a random number generator (near random) to generate those numbers and then project the evaluation results statistically. Before we evaluated through all the 76 sentences we observed a uniformly distributed alignment result on the 14 sentences in our 76 list and the alignments were so poor for every sentence pair that we found it not necessary to go through all of them. In the sentences we considered, there we obtained an accuracy of

21.8%. This results are very low compared to our alignment Model II where we obtained more than 60%.

8.4 Summary

Apart from the evaluations made for each module we evaluated our system on four languages which includes German, English, Hebrew and Amharic. Possibilities of using parallel text for transfer of annotations in one language to the other by way of aligning text units has also been tested. This approach is theoretically sound and if applied to a large amount of text better results than we have could be obtained. The use of simple linguistic tools such as morphological analysers would also be an advantage. The method we used is generic and can be used by other language pairs too. In addition, alignment Model II has been compared with the implementation of the IBM translation model GIZA++. Model II in this work perfomed better.

Chapter 9

Conclusion

This project involved the development of several algorithms and techniques of bilingual word alignment. The alignment methods developed are aimed at generating a bilingual lexicon particularly for Amharic and English.

The goal is to achieve a number of benefits in applications of such a lexicon generated from translation texts. The wealth of information in translation texts, even if not easy to attain, is undeniably enormous. Translations provide words as used in different contexts, their modifications for different grammatical functions and their functions in sentences. Domain specific terms can be accessed faster in texts in that domain than in any other source. A translation text is basically an annotation of the other text. Considering Amharic and English, they are used in parallel in Ethiopia, for Though Amharic is the national language of Ethiopia, English is the instructional language in high schools and universities. This has given English a status whereby it is used as a second language at workplaces as well. However,

the transition between the two languages is not a smooth one because of the absence of an automated way of switching between them. As a result those who understand English well depend on English as an access to information and those who do not understand it or who are not comfortable with it stick to Amharic and remain detached from access to information in English. This work is an initial step towards closing this gap.

An efficient word alignment algorithms from sentence-aligned parallel corpora is presented. For the most part, procedures for statistical translation modelling that make use of measures of term distribution as the basis for finding correlations between terms are used. Linguistic rules of morphology and syntax have also been used to complement the statistical methods.

This work is a contribution both to natural language processsing in Amharic and general bilingual-language processing in Amharic and English. On the other hand, it is a contribution to existing studies in translation text alignment in showing the problems and proposing solutions in language pairs that are different from major languages. It is also a study on language pairs that are quite disparate, having differences in morphology, syntax and not sharing a significant proportion of vocabulary. It is also a study on a language impoverished with respect to linguistic tools and data. This work is also an important contribution to other languages that are in a similar situation, technologically speaking, to Amharic.

The system is developed based on existing methods which is then enriched and modified iteratively, as well as introducing new approaches to alignment. Aligning parallel texts involves alignment at different levels of text ranging

from the course grained alignment of paragraphs and sentences or simply chunks of text to the fine grained alignment of phrases and words. The shallower the granularity of the text units to be aligned the more difficult the task. Yet, smaller units remain of prime importance for many applications where translation data can be used. This work focuses on the last task which is aligning words and chunks. Hence, the data used is an already sentence aligned translation text.

Two models are developed: Model I and Model II. They both run independently, but Model II takes the output of Model I as an input at some point. The different parts of each Model are described in the order they are developed. Model I consists of three components:

i. A statistical 1:1 word alignment system: Global aligner;

ii. A chunk parser that generates English chunks equivalent to complex Amharic words; and

iii. A finite state morphological analyser and shallow stemmer for Amharic and English.

The first step was developing procedure *i.* by using basic methods for aligning words statistically. The initial system produces matches of 1:1 alignment of Amharic lexemes to weekly inflected English words.

For a word alignment system, texts need to be aligned at sentence level. The distribution of words across these sentences is then compiled. Three parameters are used to describe the distribution of each term in the segments and in the texts as a whole: *Global-frequency*: Frequency of occurrence in the

corpus, *Local-frequency*: Frequency of occurrence in a segment, and *Placement*: Position of occurrence in the corpus (i.e. segment ID).

Each of the translation texts is mapped into a two dimensional matrix with sentences as columns and words as rows. The entries in the matrix stand for local frequencies of the words in the corresponding sentences. Therefore, each word is a weighted vector of its distribution; where the weight is its local frequency in the respective segment. The similarity in distribution of source and target words is computed from the entries in these vectors. For each source word the target word that gives the highest score is taken as the possible translation.

Using this method a list of aligned lexicon which has an accuracy of over 80% for words with frequency greater than 4.5 on average has been obtained. With filtering, words of frequency 3 also give good results. The score threshold level for which a good percentage was found before filtering was 0.55. This means the distributions of translation candidates need only overlap in almost half of the cases.

The algorithm in *ii.*, is a chunk parser that generates English chunks equivalent to Amharic lexemes. Often, when aligning bilingual corpora, one may not get a 1:1 alignment of words. This is more so in disparate language pairs. To resolve this problem, the widely used solution is to break down more complex words to their underlying components. In this work, an approache that works the other way round is reported, where several words in the simpler language are brought together to form phrases equivalent to the complex words in the other language. After securing 1:1 alignment by the

previous procedure, the English terms are relaxed to construct the actual chunk which translates the Amharic version. This chunk is generated from the parallel corpora itself wherever the translations co-occur. Basically, morphosyntactic rules of both languages are used to determine the boundaries of chunks. It was possible to generate about 72% of the chunks correctly. This allows us to be more flexible and generate non 1:1 alignments as well.

Even though the chunk parser parses sentences for correct chunks it does not solve the shortcomings of statistical methods in dealing with variants of the same word. Therefore, a finite-state morphological analyser for Amharic words has also been developed. A description of the morphology of Amharic and particularly the major morphological processes that are difficult to handle such as infixation and complex rules of partial and total reduplication have been described in finite-state formalisms. In the final implementation, there has been problems of over generation of the morphological analyser that sticking to using only a shallow stemmer was the alternative left. For equal treatment of both languages English words have also been stemmed of more salient affixes. The improvement obtained due to these shallow stemmers for both languages were different on different data (1.6% and 10%). In any case it seams more advantageous to use them than not to use them.

Model I of the alignment system has these main components. Now the major problem in this model is its low recall, that is, given a text of certain size, the lexicon it generates is small. To improve this situation, Model II which also has three components has been developed. This model attempts to take two sentences which are translations of one another and match every

word in the shorter sentence to that in the longer sentence. This would of course increase recall by far. The tradeoff on the precision is reduced by use of enhancing methods. Hence, the three main components are:

iv. Maximum likelihood local alignment system;

v. Enhancement by reusing known translations;

vi. Pattern recognition method for maximising scores .

The maximum likelihood aligner estimates the most likely translation in the target sentence for every word in a source sentence. It basically finds the most likely translation based on the scores of similarity obtained from the translation text as a whole. But since the target word with the highest score for one source word may have an even higher score for another source word, the translations should be mutual best translations. This method resulted in a precision of around 60% which obviously is lower than Model I. But the recall is 100% since every word in the shorter sentence is aligned. The routines in *v* and *vi* were developed To further improve the precision.

The first enhancement method involves the use of the lexicon generated in Model I as a list of known translations. That is those words in the translation lexicon have been excluded from the alignment space. This gives us an advantage in that for every known translation the number of possible alignments decreases, hence, also reducing the chances of failure to align correctly.

The second method is actually a pattern recognition procedure where morphosyntactic patterns, features which highlight the more likely matches,

are gathered and a decision is made on highly likely translations. Since there could be ambiguous options, specially as the sentences get longer, translations obtained using this procedure are not excluded. Rather the score is maximised by multiplying it by a factor of 1.3. For some features the results obtained are encouraging.

These systems have been evaluated in different ways. The bible has been used as a data source for testing the algorithms:

- Aligning Amharic-English translation data: Given the aligned lexicon generated for each Model, the number of correct transaltions is calculated.

- The chunk parser has been evaluated on the lexicon generated. That is for each translation, an attempt to generate the correct chunks in the text is made. A very strict evaluation was made in that a slight deviation from the exact match was given a wrong mark.

- The morphological analyser developed for Amharic is also evaluated by running words in text into the morphological analyser.

- The performance of alignment Model I across four languages (Amharic, English, German, Hebrew) has also been evaluated. interesting contrasts have been observed in this evaluation.

- The performance of alignment Model II has also been compared to GIZA++, an implementation of the IBM statistical alignment system. The results showed that GIZA++ really produced very poor results.

The methods developed are just at their preliminary stage. New approaches and techniques and more linguistic and statistical information needs to be integrated into them in order to obtain better results.

References

Steven Abney. *Principle-Based Parsing*, chapter Parsing by Chunks. Kluwer Academic Publishers, 1991.

L. Ahrenberg, M. Andersson, and M. Merkel. A simple hybrid aligner for generating lexical correspondences in parallel texts. In *Proceedings of the 36$^{\text{th}}$ Annual Meeting of the Ass. for Computational Linguistics (ACL)*, pages 29–35, Montréal, Canada, 1998.

Atelach Alemu, Lars Asker, and Gunnar Eriksson. Building an amharic lexicon from parallel texts. In *Proceedings of: First Steps for Language Documentation of Minority Languages: Computational Linguistic Tools for Morphology, Lexicon and Corpus Compilation, a workshop at LREC*, Lisbon, 2004.

Getahun Amare. *Zämänawi yamarNa Säwasäw bäqälal aqäraräb*. Commercial Printing Press, Addis Ababa, 1997.

Saba Amsalu. Data–driven amharic–english bilingual lexicon acquisition. In *Proceedings of the International Conference on Language Resources and Evaluation*, Genoa, Italy, 2006.

Saba Amsalu and Girma A. Demeke. Finite state morphotactics of amharic simple verbs. *to appear*, 2006a.

Saba Amsalu and Girma A. Demeke. Induction of amharic verb stem lexicon for finite-state morphological analysis. In *The 5$^{\text{th}}$ World Congress of African Linguistics*, Addis Ababa, Ethiopia, 2006b.

Saba Amsalu and Dafydd Gibbon. A complete fs model for amharic morphographemics. In *Proceedings of FSMNLP*, Helsinki, 2005a.

Saba Amsalu and Dafydd Gibbon. Finite state morphology of amharic. In *International Conference on Recent Advances on Natural language processing 2005*, pages 47–51, Borovets, Bulgaria, 2005b.

I. Arad. *A quasi-statistical approach to automatic generation of linguistic knowledge*. PhD thesis, UMIST, Manchester, 1991.

L. Ballestros and W.B. Croft. Phrasal translation and query expansion techniques for cross-language information retrieval. Technical report, University of Massachusetts, 1997.

Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of the 43$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics*, pages 597–604, 2005.

Regina Barzilay and Lillian Lee. Learning to paraphrase: An unsupervised approach using multiplesequence alignment. In *Proceedings of HLT/NAACL*, 2003.

Regina Barzilay and Kathleen McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of the 39$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics*, 2001.

Abiyot Bayou. Design and development of word parser for amharic language. Master's thesis, School of Graduate Studies of Addis Ababa University, Addis Ababa, 2000.

Kenneth R. Beesley and Lauri Karttunen. *Finite–State Morphology*. CSLI Publications, 2003.

Lionel M. Bender and Hailu Fulas. *Amharic Verb Morphology*. African Studies Center, Michigan State University, 1978.

Girmaye Berhane. Word formation in amharic. *Journal of Ethiopian Languages and Literature*, pages 50–74, 1992.

D. Bourigault. Sirface grammatical analysis for the extraction of terminological noun phrases. In *Proceedings of the 14*$^{\text{th}}$ *International Conference on Computational Linguistics (COLING'92)*, pages 977–981, Nantes, France, 1992.

S. Boutsis and S. Piperidis. Aligning clauses in parallel texts. In *Proceedings of the 3*$^{\text{rd}}$ *Conference on Empirical Methods in Natural Language Processing*, pages 17–26, Granada, Spain, 1998a.

S. Boutsis and S. Piperidis. Ok with alignment of sentences. what about clauses? In *Proceedings of the Panhellenic Conference on New Information Technology (NIT'98)*, pages 288–297, Athens, Greece, 1998b.

P. F. Brown, J Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, R. L. Lafferty, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 1990.

Peter Brown, Stephen Della Pietra, Vincent Della Pietra, and Robert Mercer. Word-sense disambiguation using statistical methods. In *Proceedings of ACL91*, Berkeley CA, 1991a.

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. Aligning sentences in parallel corpora. In *ACL–91*, pages 169–76, Berkeley, 1991b.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. Analysis, statistical transfer, and synthesis in machine translation. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 83–100, Montreal, 1992.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistcal machine translation: Parameter estimation. *Computational Linguistics*, 1993.

Ralf D. Brown, Jaime G. Carbonell, and Yiming Yang. *Parallel text Processing: Alignment and use of Translation Corpora*, chapter Automatic dictionary extraction for cross-language information retrieval, pages 275–298. Kluwer Academic Publishers, 2000.

Ralf D. Brown, Jae Dong Kim, Peter J. Jansen, and Jaime G. Carbonell. Symmetric probabilistic alignment. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts, pages 87 90,*, pages 87–90, Ann Arbor, 2005.

Jason S. Chang and Yu-Chia Chang. Computer assisted language learning based on corpora and natural language processing: The experience of project candle. In *Proceedings of IWLeL 2004: An Interactive Workshop on Language e-Learning*, pages 15–23, 2004.

Stanley F. Chen. Aligning sentences in bilingual corpora using lexical information. In *Proceedings of ACL–93*, Columbus OH, 1993.

Yaacov Choueka, Ehud S. Conley, and Ido Dagan. A comprehensive bilingual word alignment system. application to disparate languages: Hebrew and english. In *Parallel text Processing: Alignment and use of Translation Corpora*. Kluwer Academic Publishers, 2000.

K. W. Church and P. Hanks. Word association norms, mutual information and lexicography. *Computation Linguistics*, 16:22–29, 1990.

Kenneth W. Church. Char_align: A program for aligning parallel texts at the character level. In *Proceedings of the 31st Annual meeting of the association for Computational Linguistics*, pages 1–8, Columbus, Ohio, 1993.

Kenneth W. Church and William A. Gale. Concordances for parallel textssyntax-based alignment: Supervised or unsupervised? In *Proceedings of the 7$^{\text{th}}$ Annual Conference for the New OED and Text Research*, Oxford, 1991.

Kenneth W. Church and J. Helfman. Dotplot: A program for exploring self-similarity in millions of lines of text and code. *The Journal of Computational and Graphical Statistics*, 2(2):153–174, 1993.

Ido Dagan and Kenneth W. Church. Termight: Identifying and translating technical terminology. In *Proceedings of the 4$^{\text{th}}$ Conference on Applied Natural Language Processing*, Stuttgart, Germany, 1994.

Ido Dagan, Kenneth W. Church, and William A. Gale. Robust bilingual word alignment for machine aided translation. In *Proceedings of the Workshop on Very large Corpora: Academic and Industrial Perspectives*, pages 1–8, Columbus, Ohio, 1993.

B. Daille. *Approche mixte pour l'extraction automatique de terminologie: statistiques lexicales et filtres linguistiques*. PhD thesis, Universitè de Paris VII, 1994.

B. Daille, E. Gaussier, and J.M. Lange. Towards automatic extraction of monolingual and bilingual terminology. In *Proceedings of the 15$^{\text{th}}$ International Conference on Computational Linguistics (COLING'94)*, pages 712–716, Kyoto, Japan, 1994.

C. H. Dawkins. *The Fundamentals of Amharic*. Sudan Interior Mission, Addis Ababa, 1960.

Adrià de Gispert, Deepa Gupta, Maja Popović, Patrik Lambert, José B. Mariño1, Marcello Federico, Hermann Ney, and Rafael Banchs. Improving statistical word alignments with morpho-syntactic transformations. In *Proceedings of the 5$^{\text{th}}$ International Conference on Natural Language Processing (FinTAL)*, pages 368–379, Turku, Finland, 2006.

F. Debili, C. Fluhr, and P. Radasoa. About reformulation in full-text irs. In *Proceedings of the Conference on User-Oriented Content-Based Text and Image Handling (RIAO'88)*, 1988.

F. Debili, C. Fluhr, and P. Radasoa. About reformulation in full-text irs. *Information Processing and Management*, 25:647–657, 1989.

Mona Diab and Philip Resnik. An unsupervised method for word sense tagging using parallel corpora. In *Proceedings of the 40*$^{\text{th}}$ *Annual Meeting of the Association for Computational Linguistics*, pages 255–262, 2002.

T.E. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computation Linguistics*, 19:61–74, 1993.

Sisay Fissaha and Johann Haller. Amharic verb lexicon in the context of machine translation. In *Traitement Automatique des Langues Naturelles, TALN2003*, pages 183–192, 2003.

Pascale Fung and Kenneth W. Church. K-vec: A new approach for aligning parallel texts. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 1096–1102, Kyoto, Japan, 1994.

Pascale Fung and Kathleen McKeown. Aligning noisy parallel corpora across language groups: Word pair feature matching by dynamic time warping. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas(AMTA-94)*, pages 81–88, Columbia, Maryland, USA, 1994.

William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual of*

the Association for Computational Linguistics (ACL), pages 177–184, Berkeley, 1991.

William A. Gale and Kenneth W. Church. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102, 1994.

Daniel Gildea. Loosely tree-based alignment for machine translation. In *Proceedings of the 41*$^{\text{th}}$ *Annual Conference of the Association for Computational Linguistics (ACL-03)*, pages 80–87, Sapporo, Japan, 2003.

B. Harris. Are you bitextual? *Language Technology*, 7:41–41, 1988a.

B. Harris. Bitexts: A new concept in translation theory. are you bitextual? *language Monthly*, 54:8–10, 1988b.

Knut Hofland. *Research in Humanities Computing*, chapter A program for aligning English and Norwegian sentences, pages 165–178. Oxford University Press, Oxford, 1996.

Knut Hofland and Stig Johansson. *Corpora and cross-linguistic research: Theory, method, and case studies*, chapter The Translation Corpus Aligner: A program for automatic alignment of parallel texts, pages 87–100. Rodopi, Amsterdam and Atlanta, 1998.

D.A. Hull and G. Grefenstate. Querying across languages: a dictionary-based approach to multilingual information retrieval. In *Proceedings of*

the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96), pages 49–57, 1996.

Ali Ibrahim, Boris Katz, and Jimmy Lin. Extracting structural paraphrases from aligned monolingual corpora. In *In Proceedings of the Second International Workshop on Paraphrasing (ACL 2003)*, 2003.

Pierre Isabelle. Bi-textual aids for translators. In *Proceedings of the 8th Annual Conference of the UW Center for the New OED and Text Research*, volume 1, pages 1–15, Waterloo, Canada, 1992.

Pierre Isabelle. Machine-aided human translation and the paradigm shift. In *Proceedings of the Fourth Machine Translation Summit*, Kobe, Japan, 1993.

C. Jacquemin. *Transformation des noms composès*. PhD thesis, Universitè de Paris VII, 1991.

Lauri Karttunen. *The Oxford Handbook of Computational linguistics*, chapter Finite-State Technology, pages 339–357. Oxford University Press, 2003.

Laura Kataja and Kimmo Koskenniemi. Finite–state description of semitic morphology: A case study of ancient akkadian. In *The 12th International Conference on Computational Linguistics, COLING–88*, pages 313–315, Budapest, 1988.

Martin Kay. The proper place of men and machines in translation. Technical report, Technical report, CSL-80-11, Xerox Palo Alto Research Center, 1980.

Martin Kay. Nonconcatinative finite–state morpholog. In *The European Chapter of the ACL, EACL*, pages 2–10, Denmark, 1987.

Martin Kay. Text-translation alignment. In *ACH/ALLC '91: 'Making Connections' Conference Handbook*, Tempe, Arizona, 1991.

Martin Kay. *Parallel text Processing: Alignment and use of Translation Corpora*, chapter Preface. Kluwer Academic Publishers, 2000.

Martin Kay and Martin Röscheisen. Text–translation alignment. Technical report, Xerox Palo Alto Research Center, 1988.

Martin Kay and Martin Röscheisen. Text–translation alignment. *Computation Linguistics*, 19:121–142, 1993.

Sur-Jin Ker and Jason J. S. Chang. Aligning more words with high precision for small bilingual corpora. In *Proceedings of the 16th conference on Computational linguistics*, volume 1, pages 210 – 215, Copenhagen, Denmark, 1996.

George Anton Kiraz. Multitiered nonlinear morphology using multitape finite automation: A case study on syriac and arabic. *Computational Linguistics*, 26(1):77–105, 2000.

J. Klavans and E. Tzoukermann. The bicord system. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING-90)*, pages 174–179, 1990.

Judith L. Klavans and Evelyne Tzoukermann. Combining corpus and machine-readable dictionary data for building bilingual lexicons. *Machine Translation*, 1996.

Philipp Koehn. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA'04*, 2004a.

Philipp Koehn. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Proceedings of COLING-96, the 16th International Conference on Computational Linguistics*, volume 1, pages 115–124, Washington, DC, USA, 2004b.

Philipp Koehn. Europarl: A multilingual corpus for evaluation of machine translation. Draft, Unpublished, nd.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of HLT/NAACL'03*, 2003.

Kimmo Koskenniemi. A general computational model for word–form recognition and production. In *Proceedings of the 22nd conference on Association for Computational Linguistics*, pages 178–181, California, 1984.

Julian M. Kupiec. An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, Columbus, Ohio, 1993.

Habte Mariam Markos. Towards the identification of the morphemic components of the conjugational forms of amharic. In *Proceedings of the Eleventh International Conference of Ethiopian Studie*, Addis Ababa, 1991.

Gábor Pohl Márton Miháltz. Exploiting parallel corpora for supervised word-sense disambiguation in english-hungarian machine translation. In *Proceedings of the 5$^{\text{th}}$ International Conference on Language Resources & Evaluation (LREC'06)*, Genoa, Italy, 2006.

A.M. McEnery, J.M. Lange, M.P. Oakes, and J. Veronis. *The exploitation of multilingual annotated corpora for term extraction*, chapter Corpus Annotation: Linguistic Information from Computer Text Corpora, pages 220–230. Addison Wesley Longman, London, 1997.

A.M. McEnery and M.P. Oakes. *Using Corpora for Language Research*, chapter Sentence and word alignment in the CRATER project, pages 211–231. Longman, 1996.

I. Dan Melamed. Automatic evaluation and uniform filter cascades for inducing n–best translation lexicons. In *Proceedings of the Third Workshop on Very Large Corpora*, Boston, 1995.

I. Dan Melamed. Automatic detection of omissions in translations. Technical report, IRCS Technical Report, 1996a.

I. Dan Melamed. Automatic detection of omissions in translations. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, Copenhagen, Denmark, 1996b.

I. Dan Melamed. A geometric approach to mapping bitext correspondence. In *Proceedings of Empirical Methods in Natural Language Processing*, Philadelphia, USA, 1996c.

I. Dan Melamed. Automatic discovery of non-compositional compounds in parallel data. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Providence, RI, 1997.

I. Dan Melamed. Pattern recognition for mapping bitext correspondence. In Jean Vèronis, editor, *Parallel Text Processing: Alignment and Use of Translation Corpora*, chapter 2, pages 25–48. Kluwer Academic Publishers, 2000.

I. Dan Melamed. *Empirical Methods for Exploiting Parallel Texts*. MIT Press, 2001.

A.K. Melby. A bilingual concordance system and its use in translation studies. In *Proceedings of the Eighth LACUS Forum*, pages 541–549, Columbia, SC, 1981.

Makoto Nagao. *Artificial and Human Intelligence*, chapter A Framework of a Mechanical Translation Between Japanese and English by Analogy Principle. Elsevier Science publishers B.V., Nato, 1984.

John Nerbonne. *Parallel text Processing: Alignment and use of Translation Corpora*, chapter Parallel texts in computer-assisted language learning, pages 299–311. Kluwer Academic Publishers, 2000.

John Nerbonne. *Computer-Assisted Language Learning and Natural Language Processing*, chapter Handbook of Computational Linguistics, pages 670–698. Oxford University Press, Oxford, 2002.

Hwee Tou Ng, Bin Wang, and Yee Seng Chan. Exploiting parallel texts for word sense disambiguation: An empirical study. In *Proceedings of the 41$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics*, pages 455 – 462, 2003.

Sonja Niessen and Hermann Ney. Improving smt quality with morpho-syntactic analysis. In *The 18$^{\text{th}}$ International Conference on Computational Linguistics COLING 00*, pages 1081–1085, 2000.

Sonja Niessen and Hermann Ney. Statistical machine translation with scarce resources using morpho-syntactic information. *Machine Translation*, 30(2):181–204, 2004.

F. Och and Hermann Ney. Improved statistical alignment models. In *Proceedings of the 38$^{\text{th}}$ Annual Meeting of the Association for Computational Linguistics*, pages 440–447, 2000.

Franz Joseph Och and Hermann Ney. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40$^{\text{th}}$ Annual Meeting of the Ass. for Computational Linguistics (ACL)*, pages 295–302, Philadelphia, PA, 2002.

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

Bo Pang, Kevin Knight, and Daniel Marcu. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of HLT/NAACL*, 2003.

Philip Resnik and David Yarowsky. A perspective on word sense disambiguation methods and their evaluation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, pages 79–86, 1997.

C. J. Van Rijsbergen. *Information Retrieval*. London: Butterworths, 1979.

V. Sadler. The bilingual knowledge bank: a new conceptual basis for mt. Technical report, BSO/Research, Utrecht, 1989a.

V. Sadler. Translating with a simulated bilingual knowledge bank: a new conceptual basis for mt. Technical report, BSO/Research, Utrecht, 1989b.

Magnus Sahlgren and Jussi Karlgren. Automatic bilingual lexicon acquisition using random indexing of parallel corpora. *Natural Language Engineering*, 11(3), 2005.

S. Sato and M. Nagao. Toward memory-based translation. In *Proceedings of the I5th International Conference on Computational Linguistics (COLING-90)*, volume 1, Helsinki, Finland, 1990.

Michel Simard, George F. Foster, and Pierre Isabelle. Using cognates to align sentences in bilingual corpora. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 92)*, pages 67–81, Montreal, 1992.

F.A. Smadja. Retrieving collocations from text: Xtract. *Computational Linguistics*, 19(1):143–177, 1993.

F.A. Smadja and K.R. McKeown. Automaticall extracting and representing collocations for language generation. In *Proceedings of the 28th Annual Meeting of the Association of Computational Linguistics*, pages 252–259, Pittsburgh, Pennsylvania, 1990.

F.A. Smadja, K.R. McKeown, and V. Hatzivassiloglou. Translation collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38, 1996a.

Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. Translation collocations for bilingual lexicons: A statistical approach. In *Proceedings of the Association for Computational Linguistics*, 1996b.

David Smith and Jason Eisner. Quasi-synchronous grammars: Alignment by soft projection of syntactic dependencies. In *Proceedings of Statistical machine Translation Workshop*, New York, USA, 2006.

Harold Somers. Bilingual parallel corpora and language engineering. In *Anglo-Indian workshop on Language Engineering for South-Asian languages (LESAL)*, Mumbai, 2001.

E. Sumita, H. Iida, and H. Kohyama. Translating with examples:a new approach to machine translation. In *Proceedings of the Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages (TMI'90)*, Austin, Texas, 1990.

E. Sumita and Y. Tsutsumi. A translation aid system using flexible text-retrieval based on syntax matching. Technical report, Tokyo Research Laboratory, IBM, Tokyo, 1988.

Jörg Tiedemann. The use of parallel corpora in monolingual lexicography - how word alignment can identify morphological and semantic relations. In *Proceedings of the 6$^{\text{th}}$ Conference on Computational Lexicography and Corpus Research (COMPLEX)*, pages 143–151, Birmingham, UK, 2001.

Harald Trost. *The Oxford hand book of Computational linguistics*, chapter Morphology. Oxford University Press, 2003.

Dan Tufis, Radu, Ion, and Nancy Ide. Fine-grained word sense disambiguation based on parallel corpora, word alignment, word clustering

and aligned wordnets. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING2004*, pages 1312–1318, Geneva, 2004.

Jean Vèronis. *Parallel text Processing: Alignment and use of Translation Corpora*, pages 1–24. Kluwer Academic Publishers, 2000.

Jean Vèronis and Philippe Langlais. *Parallel text Processing: Alignment and use of Translation Corpora*, chapter Evaluation of parallel text alignment systems: The ARCADE project. Kluwer Academic Publishers, 2000.

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

Dekai Wu and Xuanyin Xia. Large-scale automatic extraction of an english-chinese translation lexicon. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas*, pages 206–213, Columbia,Maryland, 1994.

Dekai Wu and Xuanyin Xia. Large-scale automatic extraction of an english-chinese translation lexicon. *Machine Translation*, 9(3-4):285–313, 1995.

Kenji Yamada and Kevin Knight. A syntax-based statistical translation model. In *Proceedings of the 39th Annual Conference of the Association for Computational Linguistics (ACL-03)*, Toulouse, France, 2001.

Y. Yang, J.G. Carbonell, R.D. Brown, and R.E. Frederking. Translingual information retrieval: Learning from bilingual corpora. *Artificial Intelligence Journal (Special issue: Best of IJCAI-97)*, 103:323–345, 1998.

Baye Yimam. *YamarNa Säwasäw (Amharic Grammar)*. E.M.P.D.A, Addis Ababa, 1994.

Baye Yimam. Root reductions and extensions in amharic. *Ethiopian Journal of Languages and Literature*, 9:56–88, 1999.

Hao Zhang and Daniel Gildea. Syntax-based alignment: Supervised or unsupervised? In *Proceedings of the 20th International Conference on Computational Linguistics (COLING-04)*, Geneva, 2004.