University of Southampton
Faculty of Engineering, Sciences and Mathematics
School of Electronics and Computer Science

Arabic Text to Arabic Sign Language Example-Based Translation System

by

Abdulaziz Almohimeed

A dissertation submitted in fulfilment of the requirements for the award of Doctor of Philosophy

November 13, 2012

Dr. Mike Wald
Supervisor

Professor Robert I. Damper
Supervisor

Dr. Adam Prügel-Bennett
Internal Examiner

Professor Stephen Cox
External Examiner

Declaration of Authorship

I, Abdulaziz Almohimeed, declare that the dissertation entitled *"Arabic Text to Arabic Sign Language Example-Based Translation System"* and the work presented in the dissertation are both my own, and have been generated by me as the result of my own original research. I confirm that:

• this work was done wholly or mainly while in candidature for a research degree at this University;

• where any part of this dissertation has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;

• where I have consulted the published work of others, this is always clearly attributed;

• where I have quoted from the work of others, the source is always given. With the exception of such quotations, this dissertation is entirely my own work;

• I have acknowledged all main sources of help;

• where the dissertation is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signature	:	
Date	:	November 13, 2012

Acknowledgements

During my research, I was privileged to work with two great supervisors, Professor Robert Damper and Dr. Mike Wald, who helped me from the first day and were always there to patiently provide guidance and support. I express my sincere gratitude to Professor Damper for giving his time and attention to shape my draft and final thesis. His extensive knowledge and writing experience improved my thesis. I could not have published my work on well-known conferences, such as the EMNLP conference, without his help and guidance. I also sincerely thank Dr. Wald for his guidance and encouragement. His knowledge of deafness and assistive technologies was invaluable. I am grateful for his comments and suggestions during the research and writing process.

I could not have completed my research without the help of these ArSL native signers and experts: Ahmed Alzaharani, Kalwfah Alshehri, Abdulhadi Alharbi, and Ali Alholafi. In addition, I thank Mohamad Luhidan for analysing the morphological details in Corpus 1.

I also thank my parents, Ibrahim and Ruqaia, for supporting and encouraging me during my study. I am sincerely grateful to my father for funding the building of the very first two ArSL corpora. I thank my loving family-my wife Noha for travelling with me far from home, and taking care of our daughter Ruqaia and our son Saad.

Finally, I acknowledge and thank the government of the Saudi Arabia for funding my studies and my living expenses. When I return home, I plan to repay the nation by applying the knowledge that I have gained overseas.

Abstract

This dissertation presents the first corpus-based system for translation from Arabic text into Arabic Sign Language (ArSL) for the deaf and hearing impaired, for whom it can facilitate access to conventional media and allow communication with hearing people. In addition to the familiar technical problems of text-to-text machine translation, building a system for sign language translation requires overcoming some additional challenges. First, the lack of a standard writing system requires the building of a parallel text-to-sign language corpus from scratch, as well as computational tools to prepare this parallel corpus. Further, the corpus must facilitate output in visual form, which is clearly far more difficult than producing textual output. The time and effort involved in building such a parallel corpus of text and visual signs from scratch mean that we will inevitably be working with quite small corpora. We have constructed two parallel Arabic text-to-ArSL corpora for our system. The first was built from school-level language instruction material and contains 203 signed sentences and 710 signs. The second was constructed from a children's story and contains 813 signed sentences and 2,478 signs. Working with corpora of limited size means that coverage is a huge issue. A new technique was derived to exploit Arabic morphological information to increase coverage and hence, translation accuracy. Further, we employ two different example-based translation methods and combine them to produce more accurate translation output. We have chosen to use concatenated sign video clips as output rather than a signing avatar, both for simplicity and because this allows us to distinguish more easily between translation errors and sign synthesis errors. Using leave-one-out cross-validation on our first corpus, the system produced translated sign sentence outputs with an average word error rate of 36.2% and an average position-independent error rate of 26.9%. The corresponding figures for our second corpus were an average word error rate of 44.0% and 28.1%. The most frequent source of errors is

missing signs in the corpus; this could be addressed in the future by collecting more corpus material. Finally, it is not possible to compare the performance of our system with any other competing Arabic text-to-ArSL machine translation system since no other such systems exist at present.

Contents

List of Figures

List of Tables

Abbreviations

- **2D** Two-Dimensional
- **3D** Three-Dimensional
- **AAATE** Association for the Advancement of Assistive Technology in Europe
- **ABSL** Bedouin Sign Language
- **AEWLIS** ATLAS Written Italian Sign Language
- **ALECSO** the Arab League Educational
- **APU** Arabic Processing Unit
- **ASL** American Sign Language
- **ASLLRP** American Sign Language Linguistic Research Project
- **ATIS** Air Travel Information System
- **ATLAS** Automatic Translation into Sign LanguageS
- **ArSL** Arabic Sign Language
- **BLEU** Bilingual Evaluation Understudy
- **BSL** British Sign Language
- **CMU** Carnegie Mellon University
- **CSL** Czech Sign Language
- **DGS** German Sign Language

- **DRS** Discourse Representation Structure
- **EBMT** Example-Based Machine Translation
- **ECHO** European Cultural Heritage Online
- **ELAN** EUDICO Linguistic Annotator
- **EMNLP** Empirical Methods in Natural Language Processing
- **ESRC** the Economic and Social Research Council
- **EU** European Union
- **EUDICO** European Distributed Corpora
- **HAPI** HTK Application Programming Interface
- **HPSG** Head Phrase Structure Grammar
- **HTK** Hidden Markov Model Toolkit
- **HamNoSys** Hamburg Notation System
- **ID** Sign Video Identification
- **ISL** Irish Sign Language
- **JASigning** Java Avatar Signing
- **KuSL** Kuwaiti Sign Language
- **LDC** Linguistic Data Consortium
- **LH** Left Hand
- **LIBRAS** Brazilian Sign Language
- **LIU** Jordanian Sign Language
- **LK** Linguistic Knowledge
- **LOOCV** Leave-One-Out Cross-Validation
- **LREC** Language Resources and Evaluation

- **LSL** Libyan Sign Language
- **MADA** Morphological Analysis and Disambiguation for Arabic
- **MF** Manual Feature
- **MSA** Modern Standard Arabic
- **MT** Machine Translation
- **MaTrEx** Machine Translation using Examples
- **NEMLAR** Network for Euro-Mediterranean LAnguage Resources
- **NGT** Netherlands Sign Language
- **NISL** Northern Irish Sign Language
- **NMF** Non-Manual Feature
- **OVS** Object-Verb-Subject
- **PER** Position-independent Word Error Rate
- **POS** Part-Of-Speech
- **PSL** Palestinian Sign Language
- **PhD** Doctor of Philosophy
- **RBMT** Rule-Based Machine Translation
- **RH** Right Hand
- **SASL** South African Sign Language
- **SL** Signed Language
- **SLMT** Sign Language Machine Translation
- **SMT** Statistical Machine Translation
- **SOV** Subject-Object-Verb
- **STAG** Synchronous Tree Adjoining Grammar

- **SVO** Subject-Verb-Object
- **SWF** Shock Wave Flash
- **SWML** SignWriting Mark-up Language
- **SiER** Sign Error Rate
- **SiGML** Signing Gesture Mark-up Language
- **SiRR** Sign Recognition Rate
- **SignWriting** Sutton Sign Writing
- **TCL** Tool Command Language
- **TEAM** Translation from English to ASL by Machine
- **TESSA** TExt and Sign Support Assistant
- **UEA** University of East Anglia
- **UWB** University of West Bohemia
- **VANESSA** Voice-Activated Network-Enabled Speech-to-Sign Assistant
- **VOS** Verb-Object-Subject
- **VSO** Verb-Subject-Object
- **WER** Word Error Rate
- **XML** EXtensible Markup Language

Chapter 1

Introduction

This dissertation presents the first corpus-based system for translation from Arabic text into Arabic Sign Language (ArSL) for the deaf and hearing impaired. Building a system for sign language translation has additional challenges on top of the existing challenges of translating text-to-text. This chapter will begin by showing our motivation for pursuing this research. It also introduces the additional challenges of translating from text to sign language. Then, it provides an overview of the different components that are involved in building Arabic text-to-ArSL translation systems. In addition, it summarises the important contributions of this study as well as all publications that have been produced. Finally, it gives a brief summary of each presented chapter.

1.1 Motivation

Signed language (also known as sign language) is the natural visual language of deaf communication. The definition of signed language is:

> "a form of communication that uses movements of the hands and other parts of the body together with facial expressions instead of sound. There are many different forms of sign language throughout the world. British sign language (BSL) is the form most commonly used in Great Britain. In Northern Ireland, Northern Irish sign language (NISL), and Irish sign language (ISL is the most common form in Ireland) are all used." (*Oxford Concise Medical Dictionary*, 2007)

Arabic sign language (ArSL) is a form of signed language (SL), according to the definition above. It is the native language of the deaf in Arab countries and is used as a linguistic medium of communication between Arabic people in deaf communities, allowing them to express their thoughts, knowledge, and needs, as well as to share literature, history, and stories. ArSL is an independent language, rather than an interpretation of any spoken language, such as Arabic; it has its own structure, idioms, and grammar. Although there are no data regarding the total number of deaf individuals in Arab countries (Brelje, 1999, pp. 332), according to Allen (2008, pp. 14), the number of deaf people (as officially recorded by seven Arab countries out of 22 total: Algeria, Iraq, Lebanon, Morocco, Saudi Arabia, Tunisia, and Yemen) is approximately 731,240. The population of deaf individuals in each country, in the same order as above, is 240,000, 200,000, 12,000, 155,000, 100,000, 21,240 and 3,000, respectively. In addition, Brelje (1999, pp. 72-73) states that there are 2 million deaf individuals in Egypt, out of a total population of 67 million (approximately 3%). Aside from its use among the deaf, SL is used by those who are hard of hearing, and as a second language for hearing people who are in contact with native signers, such as family members and schoolteachers.

Unfortunately, most members of the Arabic deaf community do not have the opportunity to access higher education. Allen (2008, pp. 20) states that only the following specified Arab countries allow the deaf to continue their university studies: Algeria, Lebanon, Saudi Arabia, Tunisia, and the United Arab Emirates. His claim is an overstatement since, in 2008, there were no universities in Saudi Arabia that allowed the deaf to continue their higher education. In fact, based on my personal communication with Ahmed Alzahrani, the coordinator of the first Saudi higher education deaf program, the first Saudi university started offering a higher education program for the deaf in 2009. This was the Riyadh branch of the Arab Open University, a private educational institution with seven branches in seven Arab countries; in 2009 it began offering a bachelor's degree program for the deaf.

The student capacity of the Arab Open University branch in Saudi Arabia is small compared to the 24 public universities. For example, one of the latter institutions, King Saud University, has a student capacity of 80,000 (Mirza, 2009), and the remaining 23 universities have similar capacities. Therefore, since the

Arab Open University is private, the deaf must pay the required tuition fees to be accepted and to continue their bachelor's degree studies, unlike hearing Saudis, who may choose one of the 24 public universities and continue their education without any financial burden.

The university offers only one program for the deaf, a bachelor's degree in education. The program is designed to produce primary school teachers for deaf schools and was established in 2009. It is worth noting that the first university in Saudi Arabia was founded in 1957 (Rugh, 2002), which means that the Saudi deaf had to wait approximately 50 years to gain access to higher education. In addition, the coordinator of the deaf program at the Arab Open University in 2009 was Ahmed Alzahrani, an expert ArSL interpreter with a master's degree and a member of our signing team. He assisted us in building our first ArSL corpora and helped us to examine both the translation system and our new evaluation technique. The first graduation year for this program was 2011; in that year, 26 deaf students, 17 male and 9 female, earned their bachelor's degrees.

In the same year, King Saud University launched its deaf bachelor's degree programs, making it the first public university in the country to open its doors to the deaf. The university offers three programs for the deaf: bachelor's degrees in special education, art education, and physical education (Al-Raiyis, 2011). The lack of access to quality education has impacted deaf contributions to society compared to other groups, such as the blind and visually impaired. For example, it is impossible to find deaf doctors, lawyers, or judges, while there are many blind lawyers and judges and many blind people with PhDs and in high government positions.

Additionally, the lack of access to higher education for the Arab deaf has a negative impact on the deaf themselves, affecting their basic rights and needs, since high government jobs that are directly related to deaf people have to be filled by qualified persons. For example, the director of the deaf department in the Saudi Arabian Ministry of Education holds a PhD in Special Education, but is not a member of the deaf community. In addition to their inability to occupy government positions related to the deaf and hard-of-hearing, the deaf, with their current educational backgrounds, are unable to contribute their voice directly to any related global events.

In addition to the right to higher education, the deaf as a minority community are harmed by the fact that the majority misunderstands their essential needs. For example, many people believe that the deaf have the ability to read and write Arabic text with ease, and can use it to exchange information. In reality, the deaf struggle or are unable to read and write Arabic text, as has been clearly shown by a survey of Yemeni deaf (Allen, 2008, pp. 57).

Additionally, according to the same survey, most deaf individuals in Arab countries are not satisfied with this situation and demand an improved level of access to governmental and health-related services. The main obstacle facing the Arab deaf in their society is the small number or even complete absence of SL interpreters in public service interpreting. In addition, these services are expensive.

Presently, machine translation (MT) technology allows people to access knowledge published in languages other than their own. Users can utilise this technology to translate to or from most written languages, including, for example, English, Arabic, and Chinese. However, the Arab deaf or hearing-impaired do not have the luxury of using this technology to help them communicate with hearing people or to access published information.

1.2 Research Challenges and Goals

The presented work focuses on Arabic text-to-ArSL translation challenges and the necessary components for this translation, such as corpora, dealing with word ambiguity, increasing corpora coverage, etc. Meeting these challenges constitute the main goals of the research. To understand the challenges of translating from Arabic text to ArSL, and before introducing the necessary translation components, we should start by briefly introducing MT and its current paradigms.

MT for translating text–to–text languages has developed rapidly since 1947, when Warren Weaver first suggested the use of computers to translate between natural languages (Augarten, 1984). The first generation of MT systems was based on predefined rule sets. In these sets, humans manually acquired rules based on the grammatical and structural regularity of the target language. Also, all exceptions to the rules should be captured and addressed to produce an accurate translation. This approach is called rule-based machine translation (RBMT). It has many problems, including the difficulty of developing large rule sets (whereby newly added rules often interact unpredictably with existing rules) and the large numbers of exceptions encountered. As Su and Chang (1992) write: “Although the acquired rules can describe some general behaviour in natural language to some degree, they are unable to cover the wide range of fine-grained knowledge in real applications” (p. 249). Subsequently, the pitfalls of rule acquisition have been avoided by data-driven or corpus-based approaches, such that this is now the dominant technology. There are two distinct traditions in corpus-based translations. The first is statistical (SMT) (Brown et al., 1990), and the second is example-based (EBMT) (Nagao, 1984). The accuracy of translation is directly correlated to the size and coverage of the corpus used. These methods are easily extended by simply adding extra examples. Even with the rapid improvement in the accuracy of the translation approaches and techniques used, humans cannot totally rely on the accuracy of the translation produced by the machine; but these systems are useful as tools for helping humans to determine the correct translation. However, until today, they could not be used on their own because machines still cannot handle the ambiguity and meaning richness of languages.

On the other hand, achieving high-performance MT for ArSL and for most SL forms in general is more challenging, expensive and difficult because ArSL and SL in general have all the problems inherent in text-to-text translation as well as the following four main challenges. The first challenge involved is the social misconceptions about SL, including the belief held by many individuals that SL is a system, not a language, and that it is based upon a spoken language, similar to Braille, which is a written system of spoken language for the blind. In fact, ArSL is an independent language that has its own grammar, structure, and idioms, just like any other natural language.

Another example of these social misconceptions is the common belief that sign alphabets (also known as finger-spelling) are a part of SL; they are not. Sign alphabets are used by the deaf to connect to spoken language alphabets, and since the deaf cannot use their voices to pronounce spoken letters and words, these alphabets are used by deaf people who know a spoken language as a second language to illustrate the spoken alphabet. These alphabets are used for names and places that do not exist in ArSL or for other entities for which no sign exists (i.e., neologisms).

An additional misconception is that many people consider SL a universal language; however, in reality, there are many SL languages, such as Arabic Sign Language, British Sign Language and American Sign Language. A British signer, for example, needs an interpreter to communicate with an American signer.

These social misconceptions have delayed research in this field. In addition, as we will see in the ArSL-related translation systems in section 3.2, they mislead some researchers into assuming that ArSL is based on the Arabic language. Therefore, some researchers have built systems that do not take into account the unique structure and grammar of ArSL. In addition, other researchers have built systems that translate into sign alphabets.

The second challenge that has caused delays in research in the field is the lack of linguistic studies of ArSL grammar and structure, as well as the complete lack of SL documents since there is no standard written system for ArSL or SL. Many recent, efficient MT applications use corpus-based translation approaches. This approach relies totally on the corpus, and the translation accuracy is correlated with its size. Also, since there are no existing ArSL documents that could be used to build a translation corpus, which must be essentially visual (albeit with annotation), the ArSL corpus must be built from scratch, which limits its size and its ability to produce an accurate translation of signed sentences.

The third and fourth challenges involve how to present and evaluate the translated sign output. Since there is no standard written system for SL and because it is necessary to examine the translation output to measure accuracy using independent native signers, translation output can be achieved only by showing output as actual signs. Therefore, most research in sign language machine translation (SLMT) has been forced to deal with computer animation, which is necessary to represent the translated signs. This animation requires knowledge of computer graphics. As a result, the lack of computer animation knowledge on the part of MT experts has affected the amount of research conducted in this field. In addition, using the existing spoken automatic evaluation metrics is challenging since SL uses a multi-channel representation rather than a linear one. In the presented Arabic text-to-ArSL translation system, many techniques have been used and introduced to overcome all of the previous challenges.

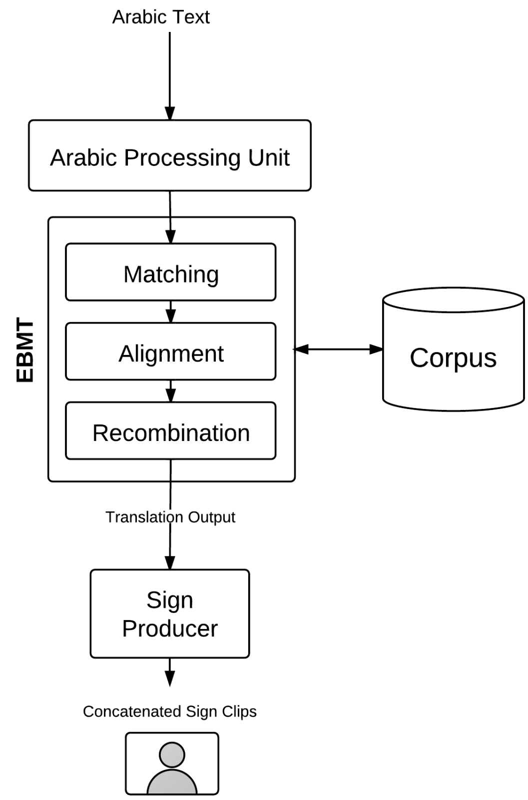


Figure 1.1: An overview of all components of the Arabic text–to–ArSL translation system.

We adopt an EBMT approach for the translation system for numerous reasons. First, the accuracy of this approach is easily extended by simply adding extra sign examples to the corpus. In addition, this approach does not require linguistic rules, relying purely on example-guided suggestions. Moreover, unlike an SMT approach, EBMT can translate using a limited corpus, such as our current ArSL corpus, and its accuracy depends only on the quality of the examples and their degree of similarity to the input text. It is important to note that there are many ways to implement EBMT systems (Carl and Way, 2003). Therefore, readers should expect different versions of Arabic text-to-ArSL EBMT which have been built to yield a final, accurate translation implemented method.

Figure 1.1 gives an overview of all the different components that are needed for translation. EBMT, in general, has three stages: matching, alignment and recombination. In the matching stage, the system finds examples in the parallel corpus that have the greatest similarity with the input text. Then, in the alignment stage, the system identifies which parts of the corresponding translation examples are to be reused for the input text. Finally, in the recombination stage, all parts of the corresponding translation examples are recombined to produce the final translation output.

In addition, we used an Arabic processing unit for reduced word ambiguity as well as to increase corpus coverage. We also constructed two ArSL corpora to be used by the system. The sign-producing unit, as shown in the figure, is responsible for preparing the signing output for the deaf. In our case, we used concatenated sign video to simplify the task of production; the alternative of using avatar is more flexible but more complex to implement. In addition, the use of concatenated sign clips allows us to distinguish more easily between translation errors and sign synthesis errors.

1.3 Outline of the Dissertation

This section provides a brief summary of each chapter in this dissertation.

- *Chapter 2* provides linguistics background information about the Arabic language, including orthography, morphology and syntax. Also, it provides some background regarding ArSL, explaining its structure, vocabulary and grammar. It presents the main misconceptions about ArSL. In addition, this chapter offers an overview of MT, including its architecture and current paradigms. Finally, it discusses the challenges facing Arabic MT systems.

- *Chapter 3* surveys all works related to Arabic morphological analysers, notation systems, sign representation technologies, corpora, translation evaluation techniques, and related translation systems.

- *Chapter 4* presents our constructed corpora. It describes all the decisions and steps involved in building and preparing the corpora for translation systems.

- *Chapter 5* discusses the design considerations and implementation of the ArSL translation system, and all related phases and factors. It introduces all the EBMT versions that have been built to yield a final accurate translation technique. Also, it presents a new technique for reducing Arabic word ambiguity.

- *Chapter 6* starts with a discussion about using humans to evaluate the translation output. Also, it presents the chosen technique for presenting the SL translated output. In addition, it presents new techniques for evaluating the accuracy of SL translation output and discusses the use of a cross-validation technique for automatic evaluation. Finally, it presents the results of an evaluation of the methods implemented in chapter 5.

- *Chapter 7* summarises the presented work and discusses future work.

1.4 Contributions and Publications List

Many factors and challenges related to ArSL translation will be discussed in this dissertation, which will lead to a number of processing methods that support the translation of Arabic text into ArSL. The following list summarises the important contributions of this study, starting with the most important.

- The lack of a standard writing system requires the building of a corpus from scratch, as well as construction of the tools for preparing this corpus. For this study, we constructed the first two parallel ArSL corpora for translation systems. The first ArSL corpus was constructed from a school-level language instruction material. It contains 203 signed sentences and 710 signs. The second corpus was constructed from a children's story and contains 813 signed sentences and 2,478 signs. Both were heavily influenced by Saudi Sign Language due to the background of the signing team.

- We did some investigation of the existing translation approaches reported by other researchers' previous studies, and we came to the conclusion to choose an example-based translation approach. This decision was made with many considerations in mind, including the lack of linguistic studies of ArSL, the high cost of building a corpus, and the cost of extending the coverage of the translation approach. Therefore, we employed two different example-based translation methods and combined them to produce more accurate translation output. We believe that we are the first to build a complete end-to-end ArSL corpus-based translation system, the first that solved the lexical and structural transfer from Arabic text to ArSL and the first to use standard evaluation methods including both automatic evaluation metrics as well as human evaluation to examine the output of the system. In chapter 5, the reader should expect to see different versions of EBMT systems. These versions are part of an investigation of EBMT to find the most suitable technique for Arabic text-to-ArSL translation. An accuracy comparison has been conducted between them.

- While working with corpora of limited size, we found coverage to be a pressing issue; in response, we employed Arabic morphological information to increase coverage and, thus, translation accuracy because all Arabic words that share the same root are related in meaning. In addition to a morphological analyser, we introduced a new technique to reduce ambiguity among a number of words so that the analyser will select the correct analysed word among those that are produced and suggested by the morphological analyser. This new technique and the morphological analyser have been combined in the Arabic processing unit; see Figure 1.1.

- We have chosen to use concatenated sign video clips as output rather than a signing avatar for two reasons. This format is simple, and it allows us to distinguish more easily between translation errors and sign synthesis errors. We present a straightforward technique based on concatenated sign video clips that help the evaluator to examine the translation output. At the outset, we were concerned that the transitions would be visually disruptive and could impair comprehension. To explore this possibility, the concatenated video output was tested by native signers in a team of three native signers plus one interpreter and found to be acceptable. Figure 1.1 shows the production of the concatenated sign output done in the sign producing unit.

- The most popular techniques for evaluating MT systems have been designed to deal with representations in a linear sequence. These techniques fail when researchers attempt to measure the multi-linear sequences of SL; regardless, most research in the field of SLMT has evaluated these techniques by discarding all parts of the body except hand movements, called non-manual features (NMFs) and combining hand movements, called manual features (MFs) as one linear output, or by considering the entire sign as one block and combining MFs and NMFs in a linear representation. Some of the studies that discarded NMFs produced a completely unrealistic evaluation of score results. For example, the sign for "*theft*" would be seen as the sign for "*lemon*", because "*lemon*" shares all of the MFs of "*theft*". In addition, when the sign is treated as one block, the metric score is usually unrealistic, specifically in cases where NMFs are deleted, or in cases where nonexistent NMFs are inserted into signs. In these instances, measurements are equivalent to the score of signs with an extra MF, or to a sign that is completely different from the original sign and shares no NMFs or MFs with it. Therefore, an evaluation metric for SL that agrees with human judgments while automatically generating scores is urgently needed. Therefore, we present a new technique that is an extension of the Word Error Rate (WER) technique, one of the most widespread evaluation techniques in other MT applications. For our presented work, the new technique will give the same results as WER since it deals with the sign as the basic unit in the translation. However, it is very useful in sign recognition systems since it considers the feature as the basic unit.

- As a technical contribution, in addition to the compiling tool, we extend the Al-Khalil morphological analyser features to make it pluggable into any NLP system. The original source, which is written in Java, expected user input only from a text box on its own graphical interface and doesn't have a feature enabling its integration with other NLP systems. Therefore, we rewrote it in Embarcadero Delphi XE, which uses Object Pascal language and extends its features to make it possible to use it with all other systems and other programming languages merely by importing it as an ActiveX component or simply connecting to it using TCP socket ports.

In addition, we present a comparison between Arabic language and ArSL in terms of their linguistic characteristics, including phonology, morphology, and structure. In fact, we found that ArSL is an independent language that is unrelated to the Arabic language except that they share the same cultural background; therefore, all of the existing proposed ArSL translation systems have been developed based on a misconception. It is also worth noting that the work has been conducted under the close guidance of a team that included three deaf native signers and one ArSL interpreter, which we hope ensures relevance to the problems faced by deaf people. As a result of this study, the following publications have been produced:

1. Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. 2009. A new evaluation approach for sign language machine translation. In *Assistive Technology from Adapted Equipment to Inclusive Environments, AAATE 2009, Volume 25*, pp. 498–502, Florence, Italy.

2. Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. 2010. An Arabic Sign Language corpus for instructional language in school. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation, LREC*, pp. 81–91, Valetta, Malta.

3. Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. 2011. Arabic Text to Arabic Sign Language Translation System for the Deaf and Hearing–Impaired Community. In *Proceedings of the EMNLP 2011 Joint Second Workshop on Speech and Language Processing for Assistive Technologies SLPAT*, pp. 101–109, Edinburgh, UK.

4. Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. 2012. Building an Arabic Text to Arabic Sign Language Parallel Corpus using a special-purpose compiling tool. *Language Resources and Evaluation* (Submitted).

5. Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. 2012. An Arabic Text to Arabic Sign Language translation system for the deaf and hearing impaired. *Computer Speech and Language* (Under Preparation).

In addition, I received an invitation from *The Arabian Journal for Science and Engineering* to review a submitted paper in the field of ArSL translation. This journal is published by Springer (http://www.springer.com/engineering/journal/13369) and had an impact factor of 0.224 in 2010.

Chapter 2

Linguistic Background and MT

This chapter provides background information about the Arabic language, including orthography, phonology, morphology, and structure. It also provides some background for ArSL, explaining its orthography, phonology, structure, vocabulary, and grammar. Further, it conveys common misconceptions about ArSL and gives a summary comparison between the linguistic characteristics of Arabic and ArSL. Finally, this chapter presents an overview of MT, including its architecture and current paradigms. In addition, it discusses the challenges facing Arabic MT systems.

2.1 Arabic Language

There are more than 300 million people in 22 Arab countries. These countries are Algeria, Bahrain, Comoros, Djibouti, Egypt, Iraq, Jordan, Kuwait, Lebanon, Libya, Mauritania, Morocco, Oman, Palestine, Qatar, Saudi Arabia, Somalia, Sudan, Syria, Tunisia, the United Arab Emirates, and Yemen. Modern Standard Arabic (MSA) is the official language of all of these countries, and it is the language of Islam's holy book, the Qur'an. Arab individuals use MSA for reading and writing as well as formal oral communication, such as education and media (newspapers, television, and radio). It is important to note that MSA in Arab countries is not a "mother" language, but instead is learned only in school; for everyday communication, Arabic dialects are used. Many Arabic dialects are used by local people, and they are spreading. MSA and Arabic dialects share similar phonological, syntactic, and lexical systems (Boudelaa and Marslen-Wilson, 2010;

Versteegh, 1997). The Arabic language is a Semitic language; it is a member of the same family as Hebrew, Maltese, Amharic, etc. This language group is well known for a non-concatenative morphology. Section 2.1.2 will explain that in detail (Abdelali et al., 2004; Habash, 2010).

2.1.1 Orthography

Arabic script is also used in writing other languages, including Urdu, Persian, and Pashto, and it was previously used with the Turkish language. Arabic script is read from right to left, and Arabic letters are connected in both print and handwriting. Short vowels usually have diacritic marks added to distinguish similar words ("كُتب" means *books*, while "كتب" means *write*). Arabic word input for machine translation should be written with vowel diacritics because without them it will be analysed in different ways, and in some cases, it is very difficult to determine the correct meaning. Arabic words can be written with "etalla", which means stretching some letters to highlight the word or just for text justification. An example is the word "معرفة" which can be written with etalla as "معـــــــــــــــــــرفة" (Buckwalter, 2004).

2.1.2 Phonology, Morphology, and Structure

Phonemes can be defined as "what we have been calling the basic form of a sound and are sensed in your mind rather than spoken or heard" (Fromkin et al., 2009, pp. 272). In other words, "Phonemes are not physical sounds. They are abstract mental representations of the phonological units of a language, the units used to represent words in our mental lexicon" (Fromkin et al., 2009, pp. 273). It is important to note that there is a debate as to whether phonemes can be considered a mental representation since there is no way to validate or prove the existence of phonemes as a mental representation. Some researchers raised concerns about accepting this definition of phonemes, including Studdert-Kennedy (1976). Others defend this role, such as Nearey (1990). Many researchers have accepted the description, consider the issue resolved, and accept phonemes as the basic form of mental representation. In addition to all these opinions, still others argue that phonemes can be considered as one of several representations of speech signal and that all representations are parallel, presenting with each other. As a result, researchers often proceed with their research as if this issue had been resolved and consider phonemes as the basic unit of mental representation (Lotto and Holt, 2000).

In spirit of this controversy, MSA is generally considered to have twenty-eight consonant phonemes and three vowel phonemes. In terms of morphology, Arabic words are derived from a root and pattern and combined with affixes. The Arabic root consists of three to four letters and is defined as a single morpheme that provides the basic meaning of a word. A *morpheme* is defined as the smallest meaningful unit of a language. In Arabic, the root is also the original form of the word, prior to any transformation process (Alkhuli, 1982). In addition, a *pattern* in Arabic can be defined as a discontinuous morpheme consisting of one or more vowels and slots for root phonemes; either alone or in combination with one to three derivational affixes, a morpheme generally has grammatical meaning (Alkhuli, 1982). An *affix* is a morpheme that can be added before (prefix) or after (suffix) a root or a stem. Using one root, several patterns, and numerous affixes, the language can generate tens or hundreds of words (Al-Sughaiyer and Al-Kharashi, 2004). A *stem* is a single morpheme or set of concatenated morphemes ready to accept affixes (Alkhuli, 1982). When a prefix or a suffix is attached to a word, some features of that word change. These features might be a number (singular, dual, plural indicating men or women, etc.), tense (past, present, future), gender (male, female, no gender), or other features. Figure 2.1 illustrates the Arabic derivational system. The three words in the top layer ("كتب" means *write*, "خبز" means *bread*, while "ذهب" means *go*) are roots that provide the basic meaning of a word. Roman letters such as *ktb* are used to illustrate the pronunciation of Arabic words. After that, in the second layer, "xAxx" (where the small letter x is a variable and the capital letter A is a constant) is added to the roots, generating new words ("كاتب" means *writer*, "خابز" means *baker*, "ذاهب" means *going*) called stems. Then, the affix "ALxxxx" is added to stems to generate words ("الكاتب" means *the writer*, "الخابز" means *the baker*, "الذاهب" means *outgoing*).

In terms of structure of the sentence, Arabic sentences usually follow the verb-subject-object (VSO) order, but sometimes they follow the subject-verb-object (SVO) order (Buckwalter, 2004).

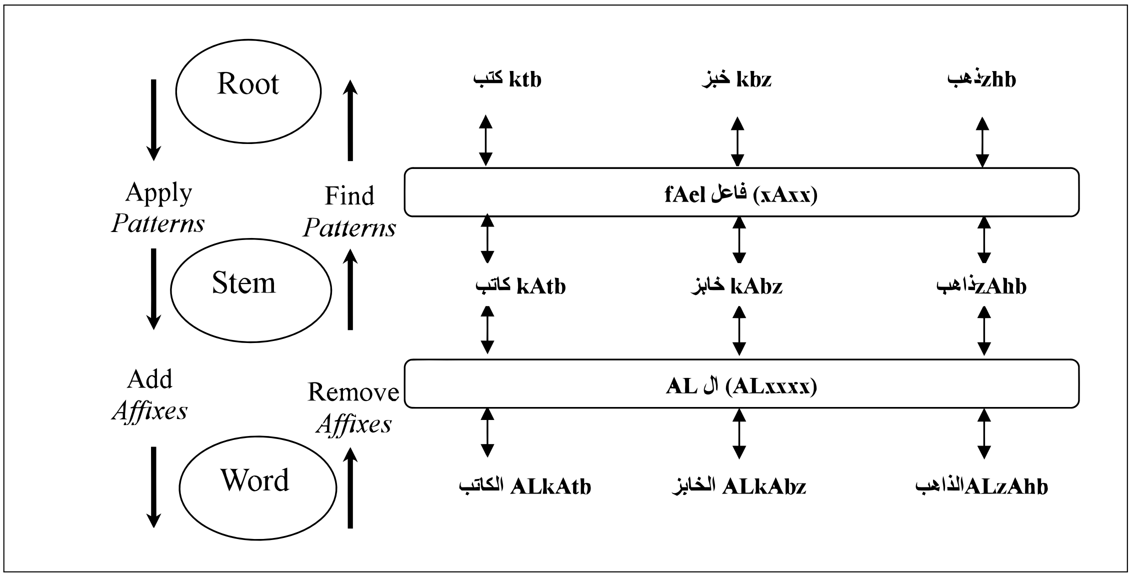


Figure 2.1: An example of the Arabic derivational system. The first stage shows some examples of roots. An Arabic root generally contains between 2 and 4 letters. The second stage shows the generated stems from roots after adding the pattern to the roots. The last stage shows the generated words after the prefixes are added to the stems.

2.2 Arabic Sign Language

Unified Arabic Sign Language or ArSL, as it is called was established as a result of immense efforts. In 1999, the first ArSL dictionary was introduced for deaf Arab usage. It was sponsored by the League of Arab States and the Arab League Educational, Cultural and Scientific Organization (ALECSO) (Al-Binali and Samareen, 2009). Afterwards, *the Geographical ArSL Dictionary* was published in 2004 by the Qatar Society for Rehabilitation of Special Needs (Al-Binali and Samareen, 2009). Since then, many ArSL local workshops have been held in many countries. These workshops aim to help the deaf and deaf interpreters to migrate from their local sign language to the Unified ArSL. Also, they aim to encourage teachers to use ArSL in deaf schools (Al-Binali and Samareen, 2009). As a result, according to Allen (2008), who surveyed the use of sign language in public television, it was found that public television in two countries (Iraq and Qatar, out of 10 respondents) had totally moved to using ArSL for deaf interpretation, instead of using the local SL. Among these channels was the Al Jazeera News channel. Also, the survey showed that 4 countries, out of 10 that responded, have used ArSL as well as the local SL. These countries are Kuwait, Palestine, Tunisia and United Arab Emirates. Also, many deaf school teachers have started using ArSL. Hanafe (2005) conducted a survey of deaf teachers in Riyadh, Saudi Arabia. His survey showed that 55% of teachers only use ArSL, and 23% teachers use it as well as the Saudi Sign Language; meanwhile, only 22% of teachers still use the Saudi Sign Language exclusively.

Before ArSL had been established, many local SLs existed, including Saudi Sign language, Jordanian Sign language (LIU), Kuwaiti Sign language (KuSL), Yemeni Sign language, Egyptian Sign language, Libyan Sign Language (LSL), Palestinian Sign Language (PSL), Bedouin Sign Language (ABSL), and others. There was a debate on whether these local SLs should be considered dialects or not. Crowley and Bowern (1977, pp. 148) stated that if 80% or more of the vocabularies of two spoken languages is similar or identical, the two can be considered as dialects of a common ancestor. Padden (2010, pp. 32), who is a professor at University of California, San Diego, did a study of the similarities between the LIU with PSL, KuSL, LSL, ABSL, and American Sign Language (ASL). Her study was based on 167 signs in PSL, 183 signs in KuSL, 267 signs in LSL, 165 signs in ABSL, and 410 signs in ASL. She concluded that LIU and other SLs in the comparison could not be considered as dialects of one another. Figure 2.2, shows that LIU and PSL have only 58% similar and identical signs in common, whereas KuSL only shows 40% commonality, LSL only 34%, and ABSL only 24%.

By contrast, Mohammed Al-Binali and Samir Samareen experts on ArSL and authors of the first reference book describing ArSL grammar, including structure, morphology, and syntax start their book with a comprehensive history and background of ArSL. Their book has been approved by five Arab academics who are experts in sign language. It claims that the local SLs in Gulf States (i.e., Saudi Arabia, Kuwait, Bahrain, United Arab Emirates, Qatar, and Oman), in Al-Sham States (which include Syria, Lebanon, Jordan, and Palestine) and in some of the North African Arab countries are very similar to ArSL. The authors also state that the similarities among these languages are higher than 70% (Al-Binali and Samareen, 2009, pp. 44). Therefore, if this claim is true, we can certainly consider all SLs in Gulf and Al-Sham states, as well as in some Arab North African countries, as dialects.

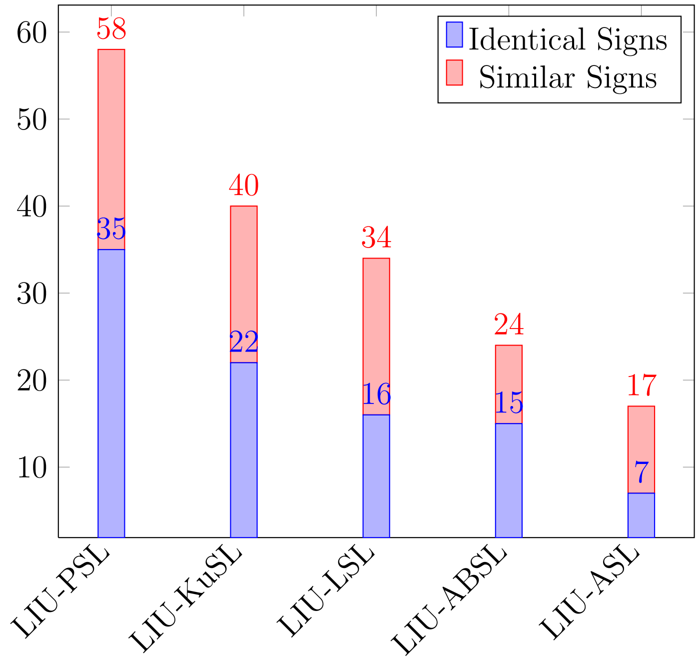


Figure 2.2: Cognates between LIU and other SLs (Padden, 2010, pp. 32).The blue shades mean identical signs and the red shades mean similar. For example, PSL has 35% identical signs, is 23% similar to LIU, and has 58% total similarity.

2.2.1 Orthography and Sources

A standard written system for SLs including ArSL comparable to that used for natural language has not been developed. Therefore, no SL written documents exist. However, in the past three years, with the booming in technology and the internet, there have been many ArSL video materials available on the web such as ArSL interpretation of news, educational videos, etc.

Like any independent language, ArSL and all the Arab local SLs change constantly over time, and many signs have been invented, adapted, and replaced. The general sources of the signs used are (Abdel-Fateh, 2005):

1. inherited from past deaf generations via the deaf community. This source is true for the local SLs but false for ArSL since it is newly introduced;
2. borrowed from other SLs;
3. created from scratch;

4. a representation of actions, shapes, and things in nature;
5. an expansion, as in compounding with and blending of other signs. All of these signs are spreading through the deaf community.

In addition, sign alphabets are used by the deaf to connect to spoken language alphabets, and since the deaf cannot use their voices to pronounce spoken letters and words, these alphabets are used by deaf people who know a spoken language as a second language to illustrate the spoken alphabet. It is used for names and places that do not exist in ArSL or for other entities for which no signs exist (e.g., neologisms). It is also worth mentioning that Arabic sign alphabets were introduced in 1986 (Al-Binali and Samareen, 2009, pp. 47) before the unified ArSL. Figure 2.3, shows Arabic sign alphabets associated with the Arabic alphabets.

2.2.2 Phonology

SL is composed of basic elements of gesture and location, previously called 'cheremes' but modern usage has changed the term to the even more problematic 'optical phoneme' (Ojala, 2011). The term 'cheremes' was coined by Stokoe (1960) during his work on American Sign Language. He used this term rather than 'phonemes' to emphasise the differences between SLs and spoken languages. However, currently, the term 'cheremes' has been replaced with the term 'phonemes' because many researchers argue that phonemes and cheremes are the same: a basic mental representation where acoustics or optics is only the carrier medium to the brain. This claim cannot be validated, but most researchers in the field consider it resolved and use the term 'phonemes' or 'optical phoneme' to emphasise the different carrier medium. These optical phonemes or cheremes involve four elements: hand shape, orientation of the hand, position of the hand in relation to the signer's body, and the movement of direction of the hand (Ojala, 2011). These four elements are called MFs. In addition, SL can involve NMFs that encompass other parts of the body - including facial expression, shoulder movements, and head tilts - in concurrence with MFs. Unlike written language, where a text expresses ideas in a linear sequence, SL employs the space around the signer for communication, and

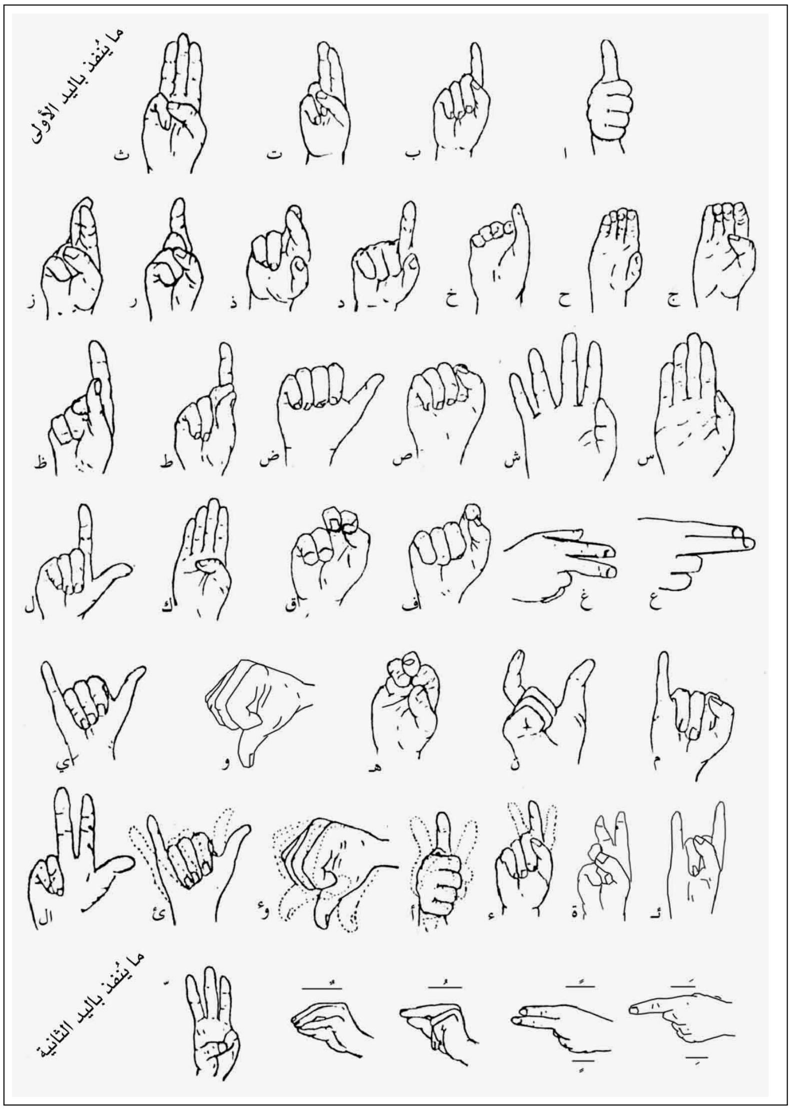


Figure 2.3: Arabic sign alphabets (Abdel-Fateh, 2005).

the signer can use a combination of MFs and NMFs. These are called 'multi-channel signs'. The relationship between multi-channel signs could be parallel or they could overlap during SL performance. MFs are basic components of any sign, whereas NMFs play an important role in composing signs in conjunction with MFs. NMFs can be classified into three types of roles. The first is essential: If an NMF is absent, the sign can have a completely different meaning (Johnston and Schembri, 2007).

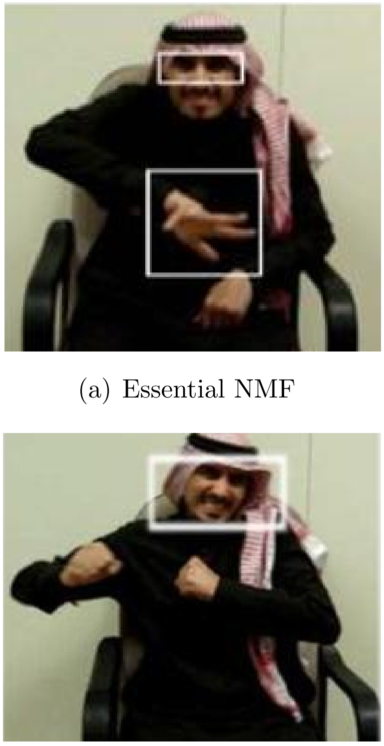
An example of an essential NMF in ArSL is the sign sentence: "Theft is forbidden", where as shown in Figure 2.4(a), closed eyes in the sign for "theft" are essential. If the signer does not close his or her eyes, the "theft" sign will mean "lemon". The second type of NMF is a qualifier or emotion. In spoken language, inflections, or changes in pitch, can express emotions, such as happiness and sadness; likewise, in SL, NMFs are used to express emotion; see Figure 2.4(b). The third type of NMF actually plays no role in the sign. In some cases, NMFs remain from a previous sign and are meaningless. Native signers naturally discard any meaningless NMFs based on their knowledge of SL.

2.2.3 Grammar and Structure

In section 2.2.2, we described the *optical phoneme*, which is the basic unit of sign, as it contains four elements: hand shape, orientation and position of the hand, and the movement of direction of the hand, as well as the NMFs. According to Al-Binali and Samareen (2009, pp. 100), the meaning of a sign changes to a different meaning when one of the four elements changes. An example of that is the "*family*" sign; when the hand shape of this sign changes, it delivers a different meaning based on this movement, to "*committee*", "*agency*", or "*organisation*", as shown in Figure 2.5.

In addition, reversing the hand shape to a different position gives an opposite meaning such as the signs "*legal*" and "*illegal*" as shown in Figure 2.6. Also, a new sign can be delivered from compounding two different signs, such as the sign "*devotion*" which contains two signs, "*fear*" and "*God*", as shown in Figure 2.7.

In terms of ArSL grammar and structure, as we mentioned before, there has been a lack of linguistic studies in both ArSL and local Arab SLs. However, based on Abdel-Fateh (2005) and Al-Binali and Samareen (2009) and from what we have seen in our conducted corpora, ArSL has grammatical structures that



(a) Essential NMF

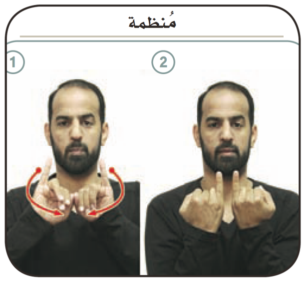
(b) Emotion NMF

Figure 2.4: (a) The sign for "theft", in which the signer uses the right hand while closing his eyes. (b) His facial expressions show the emotion of the sign.

are different from those of the Arabic language. Some general ArSL grammar rules are as follows:

- Tense in ArSL is indicated at the beginning of conversation. Later, the tense may be changed at the start of a new sentence describing a different time.
- Plural is represented by a singular sign combined with a quantity sign such as that for "*much*" or "*few*".
- Emphasis is done by facial expression, repetition, and longer signing time.
- Adverbs are expressed by the position of one hand relative to the other.
- Other features such as conditional expressions and sentence boundaries are expressed by NMFs.

(a)



(b) (c) (d)

Figure 2.5: (a) The sign for "*Family*" where (b) is "*committee*" sign and (c) is "*agency*" sign and (d) is "*organisation*" sign. The only difference between all of these signs is the hand shape(s). All of these signs share the same core meaning: "*group*" (Al-Binali and Samareen, 2009).

ArSL sentence structure is similar in flexibility to standard Arabic language; the sentences can be expressed in different orders such as SVO, VSO, OVS, and VOS.

2.2.4 Vocabulary

ArSL sign types are classified into: (1) simple forms of nouns/adjectives; (2) simple forms of verbs without tense. Prepositions can be expressed by hand position and direction of one sign in relation to the other; intensifiers can be expressed by repetition of signs.

The ArSL sign can belong to the following groups:

- Anto-Signs: two signs with the same MFs except that the movements are different (see Figure 2.8).

(a)

(b)

Figure 2.6: (a) The sign for "*legal*" where (b) is the sign for "*illegal*". They have the same sign elements, except (b) is the reverse of (a) (Al-Binali and Samareen, 2009).

Figure 2.7: The sign "*devotion*" generated from the sign "*fear*" (left side) and the sign "*God*" (right side) (Al-Binali and Samareen, 2009).

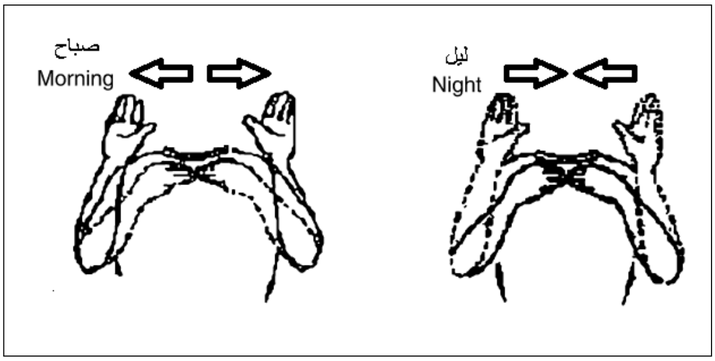


Figure 2.8: An example of anto-signs: morning and night have opposite meanings, and they share MFs but not the movement. Adapted from Abdel-Fateh (2005)
.

- Compounds: a combination of two signs to deliver a new meaning.
- Hamo-Signs: two similar signs used to express different meanings, which are known from the context.
- Syno-Signs: two signs with the same meaning. This type exists in case of shifting from one sign to another, when the first sign is not discarded, so the two signs coexist until one dominates and the other disappears.

2.2.5 Common Misconceptions

In Arab countries, the deaf community is a minority and because the official language of these countries is Arabic, deaf people must learn Arabic reading and writing to access public services such as hospital, educational, and police services. In recent years, public awareness about disabilities has increased. However, progress in deaf accessibility has been limited; one reason for this limitation is public misconceptions about both SLs and ArSL. This Section will discuss these misconceptions.

2.2.5.1 Sign Language is Not a Universal Language

Many people all over the world believe that sign language is a universal language (Huenerfauth, 2008). This belief leads to the assumption that any development in deaf accessibility technology worldwide, especially in industrial countries, will allow the Arab deaf community and all the other deaf communities to benefit from this technology without any adaptation. In fact, as we mentioned before, there are many SLs spreading all over the world. Many of these SLs are not connected to each other and a deaf person needs to learn SLs other than his/her own to be able to communicate with other deaf communities. A deaf person from Saudi Arabia, in order to communicate with a British deaf person, would need an ArSL to/from BSL interpreter. I would say that it would be hard to find an interpreter who knows ArSL and BSL, so they would probably need an interpreter to translate ArSL to/from English, then another interpreter to translate from English to/from BSL.

2.2.5.2 ArSL is Not Based on the Arabic Language

Some people believe that ArSL is based on the Arabic language, which is connected in the same way that the Arabic Braille system and the Arabic language are connected. However, ArSL is an independent communication language, not an interpretation of the Arabic language, and ArSL has its own phonology, structure, and grammar. In addition, it is similar to other natural languages, so it changes constantly, with some signs created or adopted from other deaf communities and other signs forgotten or replaced. Based on our collected corpora, we found that the relations among them are not always on a one-to-one basis. We found many words in some sentences that did not have equivalents. In addition, some signs have two or more word equivalents and vice versa. Most words are translated in ArSL sentences, but words are often skipped. Both Arabic and ArSL share the same cultural background but neither is based on the other.

2.3 Overview of MT

This Section will summarise the MT architecture and the paradigms of the MT research system. It will then discuss the translation approaches that are most used by researchers.

2.3.1 MT Architectures

MT architectures can be categorised into three types: direct, transfer and interlingua translating (see Figure 2.9). Direct translation (Arnold et al., 1994),

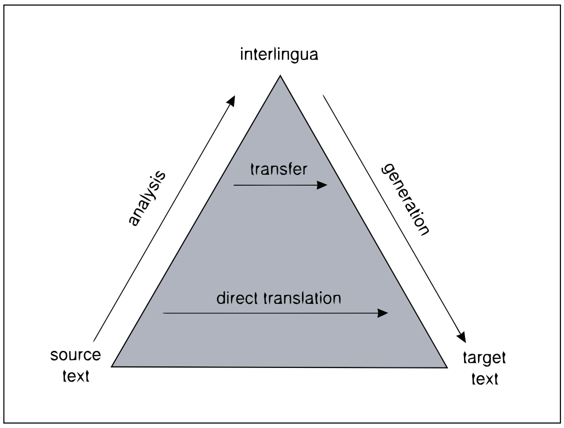


Figure 2.9: Levels of translation.

also called the transformer method, needs a one-stage process to perform lexical analysis. This analysis replaces the word in the source language with the word in the target language. Some systems use direct translation with a small set of rules to deal with simple grammatical differences between the source and the target language. This approach is unidirectional, which means that if the target text is to be translated back to the source language, a different transformer should be used. Direct translation has some problems, such as the target sentence often has an incorrect word order because it does not use a complex grammatical analysis. In addition, if lexical ambiguity exists, the translation will be incorrect. In addition, it will consume resources if it is used for a large number of rules because it is unidirectional and is bound to a specific language pair. This type is useful for specialised corpus with a limited number of lexical entries. However, for complex texts, direct translation is inaccurate.

Transfer translation is also known as indirect translation or linguistic knowledge translation. It uses the lexical, syntactic level, and sometimes semantic level to translate. This translation was introduced to improve the quality of MT, which requires linguistic knowledge of both the source and the target language as well as knowledge of the differences between them so that the researcher starts to work above the lexical level. It has three stages:

1. analysis performs syntactic parsing (some levels of ambiguity, such as lexical ambiguity, are resolved here);

2. transfer applies to syntactic-transformation rules;

3. generation performs a lexical and syntactic transfer.

There is also an intermediate knowledge-representation level called interlingua. This approach needs an analyser for each source language. It generates a language-independent representation such that the target language can be generated from this representation. It has two processes:

1. analysis builds a semantic structure using the meaning of the input sentence. This requires n analysis components, where n is the number of languages;

2. generation uses the semantic structure to generate the target-language sentence. This requires the same number as the analysis components.

2.3.2 Current Paradigms

The main paradigms of MT approaches can be categorised as linguistic-based, such as rule-based translation (RBMT), or corpus-based, such as example-based translation (EBMT) and statistical-based translation (SMT).

RBMT approach is based on predefined rules sets. In these sets, the developer or system users manually acquired rules based on grammatical and structural regularity of the target language. The rules should also capture and address all exceptions to regularity to produce an accurate translation. Also, the source language should have encoding rules that analyse it into its meaningful structural and grammatical units. This encoding is required to successfully translate the source to the target language. This approach has many problems, including difficulties in developing large rules sets, since newly added rules often interact unpredictably with existing rules, and large numbers of exceptions are encountered. Also, capturing all the language details is a problem in this approach. As Su and Chang (1992, pp. 249) wrote, “Although the acquired rules can describe some general behavior in natural language to some degree, they are unable to cover the wide range of fine-grained knowledge in real applications”. Sumita and Iida (1991, pp. 186) have stated, “When one of the following conditions holds true for a linguistic phenomenon, rule-based MT is less suitable than EBMT.

a. Translation rule formation is difficult.

b. The general rule cannot accurately describe the phenomenon because it represents a special case.

c. Translation cannot be made in a compositional way from target words".

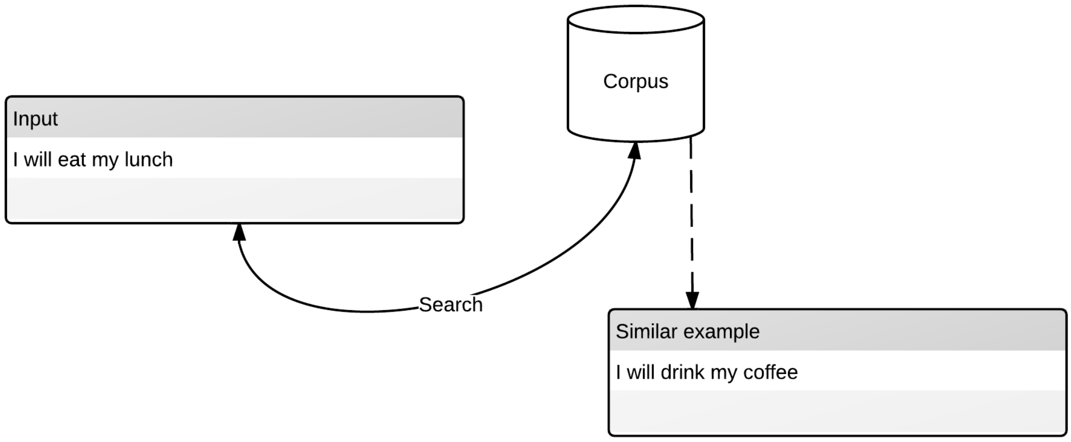
Furthermore, in our particular problem, there is no formal body of linguistic knowledge of ArSL that could guide rule acquisition. For these reasons, we do not consider RBMT acceptable for our experiments in any way.

In recent years, the pitfalls of rule acquisition have been avoided by corpus-based, or data-driven, approaches to the extent that these are now the dominant technology. There are two distinct traditions in corpus-based translations: SMT (Brown et al., 1990) and EBMT (Nagao, 1984). The accuracy of the translation directly correlates with the size and coverage of the corpus used. These methods are easily extended by adding extra examples. We favour EBMT over SMT for a number of reasons. In essence, statistical methods are predicated on the view that language is a set of regularities; it is then the task of some machine learning algorithm to extract these regularities from training data, that is, from the corpus. Some people think this is an incorrect way of looking at language. As Jones (1996, pp. 1) writes 'Language is not something that can be described in a neat and tidy way'. Exceptions are ubiquitous in language (Daelemans et al., 1999; Baayen, 2001). In contrast to SMT, EBMT makes no attempt to extract problematic regularities but instead relies purely on example-guided suggestions. Its accuracy depends only on the quality of the examples and their degree of similarity to the input text. An important reason for choosing EBMT is that it generally requires a far smaller corpus size than SMT does (Gough and Way, 2004), and SL corpora are generally small because resources have to be expended to build them. Hence, we chose EBMT for our ArSL translation experiments.

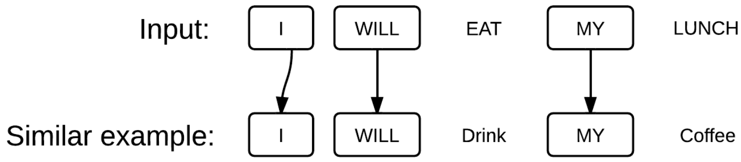
2.3.3 EBMT Methods

Generally, EBMT has three main components (Figure 2.10).

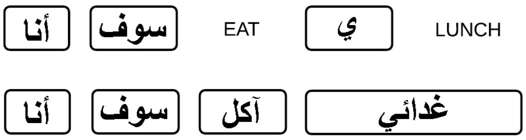
a. Matching: The system will search the corpus to find the example closest to the input sentence. The classic EBMT system proposed by Nagao (1984) suggests using word-based matching using word edit distance (Levenshtein, 1966);



(a) Matching



(b) Alignment



(c) Recombination

Figure 2.10: The main components of an EBMT system.

b. Alignment: In this component, the system identifies corresponding fragments; and

c. Recombination: In this component, the target text is produced from the translated fragments as well as from the dictionary.

The classical methods use original examples paired in the corpus without further pre-processing. However, many proposed EBMT methods use the different linguistic characteristics of the input sentence as well as the examples in the corpus to improve translation accuracy.

2.3.4 Problems of Arabic Machine Translation

When researchers started to study Arabic machine translation, the translations were poor. The reason was that Arabic language has different characteristics from other languages in the Indo-European family. Later research has combined two approaches to increase accuracy of the translations. One approach uses Indo-European translating approaches to emphasise syntactic aspects of the language (Izwaini, 2006) with or without adaptations, and the second type emphasises the linguistic idiosyncrasies of Arabic, such as morphological and grammatical concepts. The results of this new generation of research are promising, and translation accuracy has been increased. Today, several applications have been developed to translate Arabic to or from English, such as Sakhir's Motarjim and Google Language Tools. According to Izwaini (2006), Arabic encounters the same challenges that other languages do, such as:

- Multiple meanings: The meanings of these types of words are known from the contexts in which they appear. An example of a word with multiple meanings is "قوة", which has many meanings, including *power*, *force*, and *strength*.
- Word order: Arabic sentences can be constructed in SVO order or VSO order. When an Arabic sentence is in VSO order, a translation problem of syntax can arise.

Arabic shares with other members of the Semitic family the need for a sufficient morphological analyser which can deal with some language aspects, such as adding diacritical marks. In Arabic, adding diacritical marks is unnecessary because an Arabic speaker can determine a word's meaning from the context; the major problem in translating from Arabic arises from the non diacritical Arabic words, which leads to the incorrect selections of target language words. An example of non diacritical words is the word "كُل", which means *you must eat*. If the word is translated without diacritic, as in "كل", it means *all*. This leads to an incorrect translation of the original statement.

Chapter 3

Sign Language Processing

In order to translate Arabic text into ArSL, a translation system must apply different functions to the Arabic input text. Figure 3.1 displays a block diagram of how the system will function. As in Figure 3.1, The Arabic morphological analyser is an important tool for predicting the stem and root of unknown words that are not in the dictionary. Then, the system can use the word that shares either the same stem or root with the unknown word. Therefore, this chapter begins with a survey of the different types of Arabic morphological analysers and some of the well-known analysers covered in section 3.1. As shown in Figure 3.1, most of the related SL translation systems, including all prototype systems for ArSL, are presented in section 3.2. Since SL does not have a standard written form but, rather, several methods of notation that each serve a different purpose, section 3.3 introduces the two most common SL notation systems, and section 3.4 presents many related SL corpora and the tools used to construct them. In addition, section 3.5 discusses past and current work in both concatenated videos and avatar technology for sign synthesis. Finally, the current method for automatic evaluation is introduced in section 3.6.

3.1 Arabic Morphological Analysers

Many Arabic language processing systems have been proposed and developed for different purposes, such as MT, information retrieval, text-to-speech conversion, and speech recognition. Given that Arabic has a rich and complex morphology, extracting morphological information has become an essential part of any Arabic

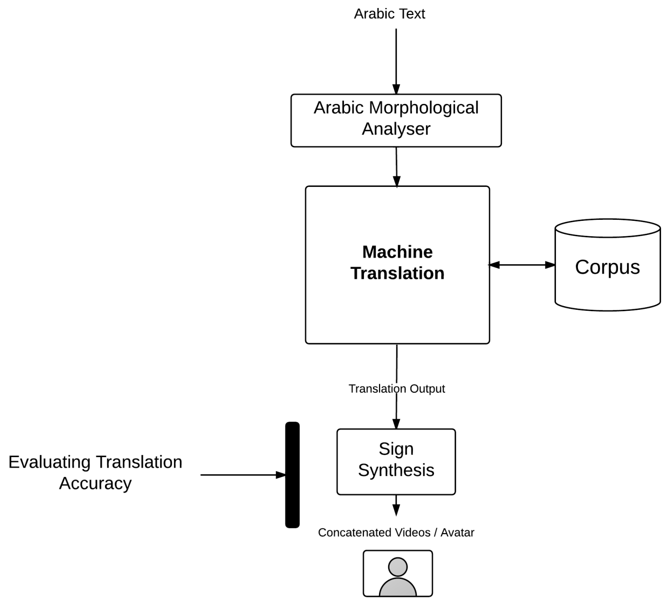


Figure 3.1: Block diagram of how the Arabic text to ArSL translation system will function.

processing system. This is particularly true for Arabic words. The primary functions of the morphological analyser are the segmentation of a word into a sequence of morphemes and the identification of the morpho-syntactic relations between the morphemes (Semmar et al., 2005). Therefore, considerable effort has been devoted to developing systems to extract this information. Al-Sughaiyer and Al-Kharashi (2002) classified Arabic morphological analysers into three categories:

- Table Lookup: This approach employs a large lookup table to store Arabic words with their corresponding morphological elements, including the root, stem, and affixation. A binary search can reduce the access time needed to retrieve the required information. In addition, a compression technique can decrease storage requirements. The problem with this technique is that developing the table requires considerable effort (Al-Sughaiyer and Al-Kharashi, 2004). Further, the analyser cannot do anything with words that are not in the table.

- Linguistic: This approach involves analysing words by deeply reviewing their morphological components, as shown in Figure 3.2. First, it removes

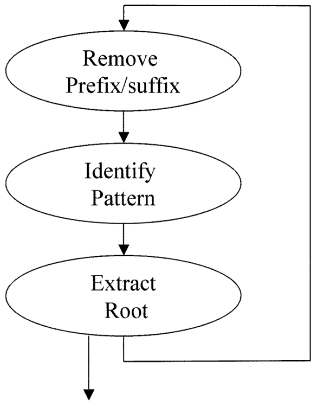


Figure 3.2: All linguistic steps in revealing stems and roots (Al-Sughaiyer and Al-Kharashi, 2004).

the prefix/suffix using a list of affixes so the stem can be extracted. Then, using a list of patterns, it searches for a match pattern. Once it locates the stem pattern, the root can be extracted.

- Combinatorial: In this approach, a given word is used to generate all possible combinations of the given word letters. Each combination is compared against a list of roots. Once a match is found, the root, stem, and pattern can be extracted. Otherwise, system will continue to compare all possible combinations until it finds a match. Figure 3.3 shows an example of how this approach extracts a root. The advantage of this approach is its simplicity, whereas its drawback is that it requires a long processing time and large lists.

As previously noted, there are many Arabic analysers. Some are freely available, one is open source and others are commercial systems. This Section will survey some of the well-known analysers used widely in the field of Arabic NLP.

3.1.1 Buckwalter's Arabic Morphological Analyser

This analyser was developed by Tim Buckwalter and released in 2002 (Buckwalter, 2002). It is distributed by the Linguistic Data Consortium (LDC) and written in Perl script language. The analyser is freely available on the LDC

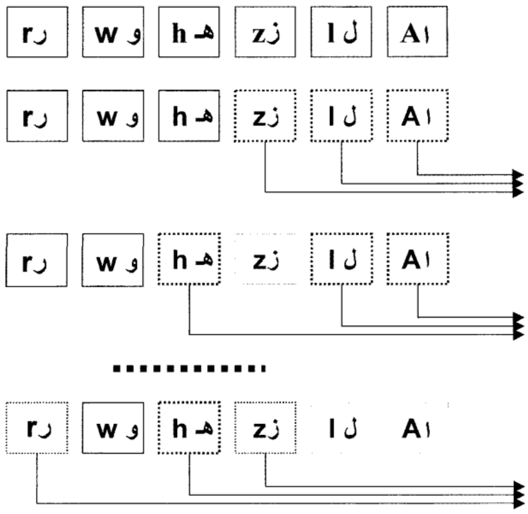


Figure 3.3: Example of extracting the "زهر" root from the word "الزهور" in a combinatorial system (Al-Sughaiyer and Al-Kharashi, 2004).

website. Downloading version 2.0 requires a LDC membership subscription, while version 1.0 can be downloaded without membership. Hajič et al. (2005) describe this analyser as "the most respected lexical resource of its kind". Many Arabic resources and processing systems have adapted and used it, including LDC Arabic POS-tagger, Penn Arabic Treebank and the Prague Arabic Dependency Treebank (Attia, 2008). The Morphological Analysis and Disambiguation for Arabic (MADA), a morphological tagger for MSA, also uses this analyser. This analyser has over 77,800 stem entries (Abouenour et al., 2008). It analyses Arabic text characters and displays the analysed output in a transliteration system, as shown in Figure 3.4.

According to Beesley and Karttunen (2003), "the term transliteration denotes an orthography using carefully substituted orthographical symbols in a one-to-one, fully reversible mapping with that language's customary orthography". The problem with a transliteration system that uses the definition of Beesley and Karttunen (2003) is that it is not a recognised written standard. Additionally, mixing Arabic and Roman text in the same document is not easy. Finally, this system fails to represent Arabic punctuation. Despite these drawbacks, Attia (2008) notes that the advantage of using this analyser is that it has less ambiguity than Xerox's morphological analyser. In addition, this analyser provides an

ء	'	ذ	*	ل	l
آ	|	ر	r	م	m
أ	>	ز	z	ن	n
ؤ	&	س	s	ه	h
إ	<	ش	$	و	w
ئ	}	ص	S	ى	Y
ا	A	ض	D	ي	y
ب	b	ط	T	ـً	F
ة	p	ظ	Z	ـٌ	N
ت	t	ع	E	ـٍ	K
ث	v	غ	g	ـَ	a
ج	j	ـ	_	ـُ	u
ح	H	ف	f	ـِ	i
خ	x	ق	q	ـّ	~
د	d	ك	k	ـْ	o

Figure 3.4: Buckwalter transliteration system.

English glossary of the meaning of each analysed Arabic word. Therefore, many non-Arabic speaker researchers prefer this analyser.

3.1.2 Xerox Arabic Morphological Analyser

Beesley (2001) and Beesley and Karttunen (2003) developed the Xerox Arabic Morphological analyser, which can be accessed on-line through the Xerox website (`http://open.xerox.com/Services/arabic-morphology`), and was developed in Java programming language as a web applet. This analyser uses a finite-state technique, employs a list of 4,930 roots and 400 patterns and can produce 90,000 stems (Attia, 2008). The analyser has two levels. The first level extracts roots and patterns, while the second level extracts affixes and other forms such as conjunctions, definite articles and prepositions. The advantage of using this analyser is its large coverage. In addition, as with Buckwalter's Analyser, it provides an English glossary for each analysed word. The drawback of this system is that it classifies the analysed words into only four categories: verbs, nouns, participles, and function words such as prepositions and conjunctions. This analyser also uses its own transliteration system.

3.1.3 Al-Khalil Arabic Morphological Analyser

Al-Khalil, released in 2010, is an open-source analyser written in the Java language (Boudlal et al., 2012) and can be downloaded from this link: (http://sourceforge.net/projects/alkhalil/). The Al-Khalil analyser was officially adapted by ALECSO after it won the award for best ALECSO Arabic morphological analyser. This analyser does not use any transliteration system, only accepts Arabic text and generates analysed words presented in Arabic characters. It does not provide an English glossary feature. Since its adoption by ALECSO, it has been rapidly updated. After an evaluation of many analysers, Altabbaa et al. (2010) concluded that "Alkhalil could be considered as the best Arabic morphological system". This analyser employs a very large set of rules. In addition, it uses very large lists of roots, patterns, suffixes and special words. The noun and verb patterns contain about 28,000 patterns with full vowelisation. These patterns were extracted from the Sarf, (see http://sourceforge.net/projects/sarf/), and NEMLAR (Sawalha, 2011) corpora. In addition, the root list, extracted from the Sarf corpus, contains approximately 7,000 roots. The advantages of using this analyser are that it analyses words into a large number of categories and subcategories and its lists are frequently updated to include more morphological information.

3.2 Related SL Translation Systems

This section will introduce some related translation systems from written text in other languages into other SLs. In addition, it will also introduce some attempts to build an Arabic text into ArSL translation systems. We are presenting the related systems as three different categories: rule–based, corpus-based and existing proposed ArSL systems.

3.2.1 Rule–Based Systems

Many SL translation systems have been developed. One of the first attempts at tackling SL translation was rule-based. Veale et al. (1998) proposed English to ASL, ISL and Japanese SL systems. The Zardoz system is considered one of the earliest serious attempts to address the challenges of building a written to visual language translation system. The system used a blackboard control structure (Huenerfauth, 2006). It also employed AI knowledge-based reasoning to produce the sign output. The output was presented in glossing text format. The system was not fully implemented due to the fact that it was designed to address translation challenges. Therefore, it has not been used or evaluated. In this section, we present some well-known, rule-based translation systems.

The TESSA system is the UEA's first attempt at building a translation system. This experimental system aims to translate a clerk's speech at the post office into BSL animation signing via an avatar (Wray et al., 2004). The system is funded by the Royal Mail Office. Its evaluation was funded by the EU as a part of the ViSiCAST project. The system deals with three main problems:

1. Speech recognition that focuses of how to make an accurate English speech into English text conversions.

2. MT concerned with how to make an accurate translation from English text into BSL.

3. A signing avatar responsible to show these BSL written representations in a realistic BSL performance.

The speech recognition problem is not a new research problem. There has been significant work in this area that has resulted in efficient speech recognition software that can be used on a daily basis such as IBM ViaVoice (http://www-01.ibm.com/software/pervasive/embedded_viavoice/).
With regard to the last problem, the TESSA researchers have already proposed a prototype signing avatar called SignAnim (Bangham et al., 2000b). This avatar uses the subtitles for television as the input and then translates them into Sign Support English, which is a sign system that depends on the English language with the same word order and vocabulary, etc. Therefore, the new research area that has not been tackled yet is how to translate from English text into BSL.

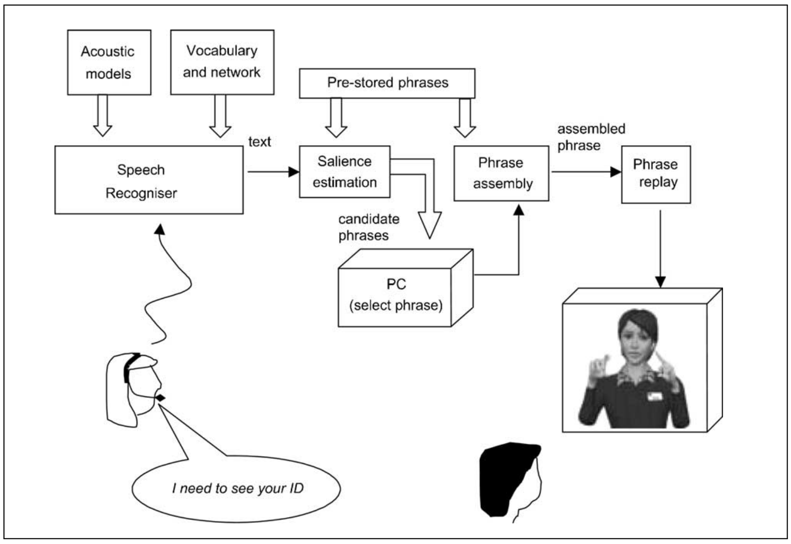


Figure 3.5: The components of TESSA system.

Figure 3.5 shows the components of the TESSA system. The first component is the speech recogniser. For this component, at the beginning of the work on the system, they used the Entropic HAPI system (Odell et al., 1997). This system includes the Hidden Markov Model Toolkit (HTK) recogniser (Young et al., 1995). Using this system enabled the system researchers to use acoustic models that they had already prepared in their laboratory using HTK. Later, they switched from using the Entropic HAPI system to using the IBM ViaVoice recogniser, which has flexible features so it can be used either as a speech model that requires an hour for setting up each speaker or a network model that does not require a setup speaker voice dataset since it uses an existing dataset over the network. Hence, the speech model has higher recognition accuracy.

Since people at the post office use highly constrained discourse, the researchers decide to use a phrase lookup approach for the translations. This approach is a straightforward technique that stores all the possible discourse used in a post office in English and a BSL lookup table. Therefore, when a clerk enters a new input text, the system searches for matched phrases and then concatenates them and presents them via the avatar.

Each BSL phrase is stored with its associated English text and was captured using a motion capture technique with the following sensors to capture different MF and NMF details:

1. Cyber-gloves to capture the parts of the MFs. It has 36 resistive elements to capture the movements of the fingers and thumb positions relative to the hands.

2. Polhemus, which is a magnetic tracking sensor, to capture the other details of MFs. It captures the wrist, upper arm, hand and upper torso positions.

3. NMFs capture expressions by using a helmet-mounted camera with infrared filters and surrounding infrared light-emitting diodes to avoid scotch light reflectors from the face. About 18 reflectors are placed in the areas of the main NMFs such as mouth and eyebrows.

The signing avatar was built using C++. The researchers developed a container system to hold all the components using TCL in which the speech recognition is included as a TCL extension. The communication between the avatar and the container system is done via a TCP/IP socket connection using a remote procedure call.

With regards to the evaluation of the system's translation accuracy, the researchers did not evaluate the system using automatic metrics but decided to conduct a human evaluation study. Therefore, in terms of the acceptability of transactions among clerks and the deaf, Figures 3.6 and 3.7 show the ratings that the system received. They used a 3-point scale for evaluation where 1 means low and 3 means high. The average rating received from the deaf was 1.9 with TESSA and 2.9 without it. In addition, the rating received from clerks was 2.5 with TESSA and 2.6 without it. In terms of the quality of signing, the average identification accuracy of complete phrases was 61%. The average accuracy of identifying a sign unit in a phrase was 81%. The previously presented evaluation is for output that goes through three main components: speech recognition, translation, and signing animation. Therefore, I believe the results can be considered as encouraging and deemed acceptable. The reasons for the deaf and the clerks preferring not to use the system might be because they establish an understanding connection between themselves through body language. However, the identifications accuracy shows that the results are acceptable. The drawback of this system is it can be used only in a very limited domain.

Later as a part of ViSiCAST project, Cox et al. (2002b) and Marshall and Sáfár (2005) developed a rule-based translation system from English into BSL. The

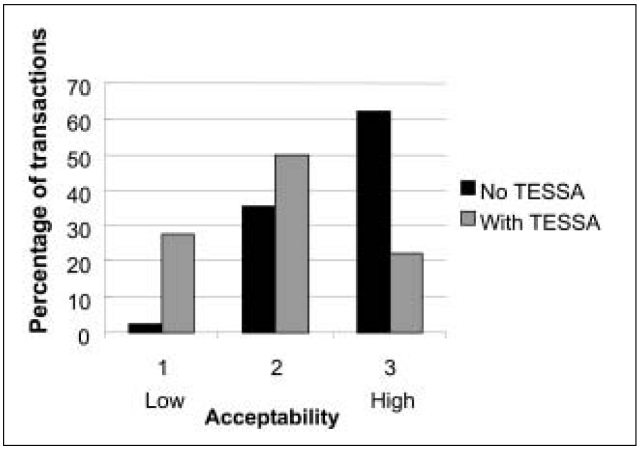


Figure 3.6: This figure shows the ratings that the system received from deaf evaluators (Cox et al., 2003).

architecture of this system is presented in Figure 3.8.

As shown in the figure, the system first employed Carnegie Mellon University's link parser (Sleator and Temperley, 1991) for parsing English text and creating linkages of English syntactic dependency based on the selections of a user intervention option. After that, the linkages' output format was transferred to the Discourse Representation Structure (DRS), the transformation process described in (Kamp, 1981). Then the system converts the format to Head Phrase Structure Grammar (HPSG) representation (Pollard and Sag, 1994). Finally, SiGML generated from these HPSG is based on 250 lexical items in BSL. The problem with the proposed system is it requires user intervention in three modules. In addition, it does not take into account the NMFs. Lastly, no evaluation was conducted to test this prototype.

Zhao et al. (2000) proposed a prototype machine translation system to generate ASL using a signing avatar from English text. Their system was called TEAM, which stand for Translation from English to ASL by Machine. It consisted of two steps. First, English sentences were translated into a tree representation using Synchronous Tree Adjoining Grammars (STAGs). The tree identified grammatical information, morphological information, and the sentence type. Second, the trees were parsed. In addition, the ASL trees were assembled. That is, each English word was replaced by an ASL gloss from an English-to-ASL lookup table that stored English words and ASL signs in pairs. This approach is

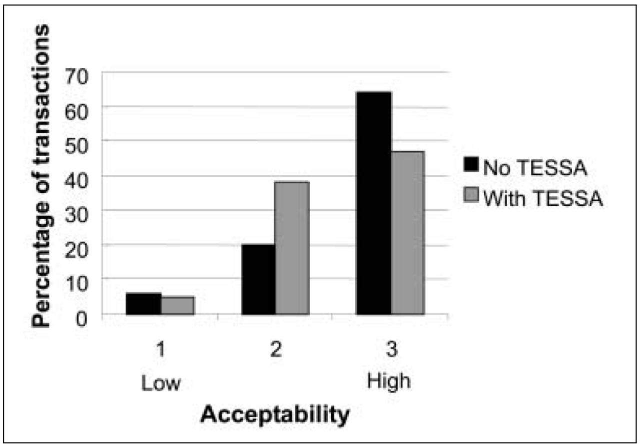


Figure 3.7: This figure shows the ratings that the system received from clerks (Cox et al., 2003).

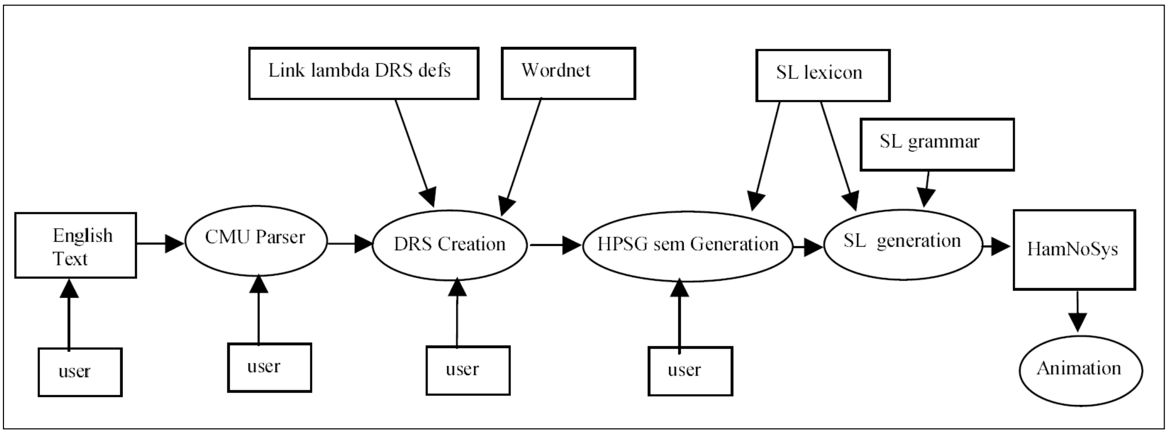


Figure 3.8: The rule-based UEA translation system (Marshall and Sáfár, 2005).

considered to be a syntactic transfer approach, as the tree presented syntactic and grammatical information. This information enabled the direct mapping of English sentences to ASL. To evaluate the system's animation output, Native American sign-language signers were consulted (Huenerfauth, 2006). Unfortunately, the evaluation scores of the participant signers were not published.

Another translation system is the SASL–MT project. This project (van Zijl, 2006) is an English–to–SASL machine translation system that was proposed and partially implemented at Stellenbosch University in South Africa. The aim of this project was to increase the accessibility of information for the South African deaf community. The project was designed for use in domains in which the need was greatest, such as hospitals and police stations. A rule-based approach was used for English-to-SASL translation, and XTAG rules were employed to build the SASL syntactic structure necessary for the parsing of an English text (van Zijl and Barker, 2003). The work was divided into different phases. The first phase was data collection. In this phase, the researchers collected about 800 signs and built an English–to–SASL look-up table containing frequently used signs. The words were collected by SASL interpreters who analysed sign-language video data. The domain of the collected data is unknown. The second phase was machine translation. Firstly, a STAG parser was designed. Then an SASL grammar tree and a rule-based tree were manually built from prototype sets of sentences. Secondly, the MT phase was evaluated and the results tested in the deaf community. The evaluation scores were not published. The final phase was building a pluggable signing avatar component. The avatar was based on a Humanoid Animation Standard (van Zijl and Barker, 2003). However, the researchers plan to extend the standard to include facial expressions in SASL (van Zijl and Barker, 2003).

3.2.2 Corpus-Based Systems

Many corpus-based studies on SL translation have been conducted recently. In this section, we introduce two of the most closely related works to our present works in this dissertation.

Morrissey (2008) developed the first EBMT prototype system for sign-language translation. This system is based on the work of Veale and Way (1997), with further extensions by Way and Gough (2003, 2005), which used a small set of tags for sub-sentence segmentation to represent the grammatical structure, called the Marker Hypothesis. We also refer to it as Marker-Based EBMT. Hence, this latter work was designed for tagging both source and target (where there are parts of speech taggers for both the source and target languages) and is consequently not well suited to ISL due to non-existence of an ISL tagger; this will be discussed later in Section 5.2. The goal of the researchers was to develop an EBMT system for English-to-Irish sign-language. However, when they began developing their system, there was no corpus of ISL available. At that time, there was one under construction at the Centre for Deaf Studies in Dublin (see Section 3.4.1.2). As a result, they began to experiment with translating English into NGT by using the ECHO corpus (see Section 3.4.1.1).

Table 3.1: Manual evaluation scores for the EBMT system (Morrissey, 2008).

	Good	**Fair**	**Poor**	**Bad**
Group 1	40	0	0	0
Group 2	4	20	16	0
Group 3	0	12	20	8
Group 4	0	0	12	28

The researchers split the translation process into three phases. Firstly, the system searched the source text to find the closest matches and translations. Secondly, it found the sub-sentential translation relations in the retrieved examples. Thirdly, it recombined the relevant parts of the target translation relations to derive the translation. To evaluate and test the system, the researchers created test sets manually and divided them into four groups:

1. The first group evaluated full sentences taken directly from the corpus.

2. The second group evaluated grammatical sentences formed by combining chunks taken from different parts of the corpus.

3. The third group evaluated sentences made of chunks not in the corpus and the combined chunks from the corpus.

4. The fourth group evaluated sentences made up of words present in the corpus. Table 3.1 summarises the manual evaluation scores.

Further, 55 text sentences were evaluated using automatic MT evaluation metrics. Those sentences achieved 96% Sentence Error Rate, 78% Position-independent Word Error Rate (PER), and 119% WER (Morrissey, 2008).

The Joining Hands Translation System is another system by Morrissey and Way (2007). It converts English text into ISL using a data-driven MT engine called MaTrEx (Stroppa and Way, 2006), which stands for Machine Translation using Examples. This engine was developed at Dublin City University. It combines SMT and EBMT. As in Figure 3.9, the main modules that this system contains are as follows:

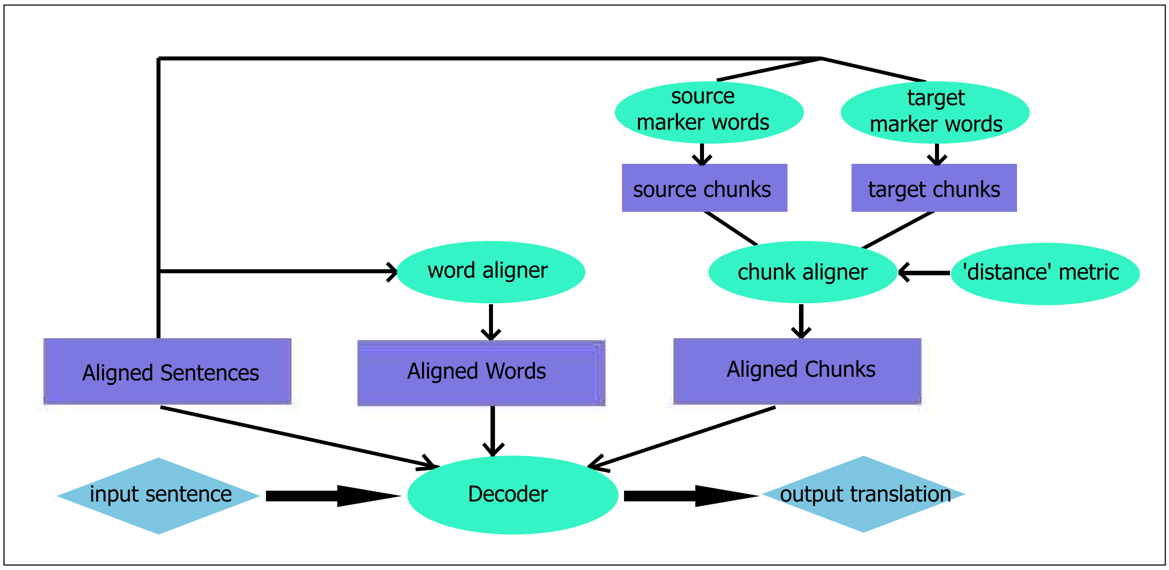


Figure 3.9: MaTrEx main components (Stroppa and Way, 2006).

- Aligned Word Module: This module uses the statistical word alignment toolkit GIZA++ (Och and Ney, 2003) to drive word/sign alignments.
- Aligned Chunks Module: This module employs Marker Hypothesis, see Green (1979) for more information, to drive a chunk which is a sub-sentence alignment between English and ISL.
- Aligned Sentence Module: This module as well as a word and chunk module feed the decoder module. However, the alignment in this module is done manually in the pre-processing stage.
- Decoder Module: This module takes input sentences. Then, it searches for sentence, chunk, and word matches using the previous modules. MOSES (Hoang and Koehn, 2008) is employed for selecting the most similar phrases for translation and these phrases are then used to generate the output sentence.

They used airport information announcements as the domain of their constructed English-ISL corpus. This corpus is now part of ATIS corpus, see Section 3.4.2.3. It contains 577 sentences. Two native ISL signers from the Irish Deaf Academy were used for translating the English sentences into ISL. They used gloss notation to annotate the ISL signs using ELAN software. The annotations were in basic form; in other words, they didn't include details such as NMFs. The goal of this system is to develop a practical translation system that can be used by the deaf community. The work was done with a strong collaboration with members of the Centre for Deaf Studies in Dublin. They helped with human translation of the sentences that were used as reference sentences during automatic evaluation. They also provided valuable advice with regards to ISL grammar and linguistics. Finally, they did the manual evaluation of the translated output. For evaluating this system, they experimented using three different techniques. The baseline was SMT only. Chunking method 1 used markers in both English and ISL sentences for the segmentation and alignment that later fed the statistical decoder. Since ISL and SL, in general, have a natural lack of linguistic studies to determine variety of the ISL marker, in method 2 they started making each ISL sign unit from its own segment and then aligned it with the English segment that was produced using Marker Hypotheses. Table 3.2 shows the evaluation scores obtained. 118 sentences have been used as testing sentences. As shown in the table, method1 and 2 have similar translation accuracy.

Table 3.2: Evaluation scores of the three techniques using WER and PER metrics (Morrissey and Way, 2007).

	WER (%)	**PER (%)**
Baseline	41.68	32.53
Chunking Method 2	40.96	29.75
Chunking Method 1	40.60	31.80

3.2.3 Proposed ArSL systems

Most systems wrongly assume that ArSL is dependent on the Arabic language. Therefore, these systems perform word-to-sign translation without consideration of ArSL's unique linguistic characteristics, such as its grammar, structure, idioms, and regional variations (Abdel-Fateh, 2005). Therefore, ArSL translation must be done in close collaboration with the deaf community and ArSL experts. We deem this collaboration by engineers and researchers with the Arabic deaf community and ArSL experts essential for understanding fundamental challenges in ArSL translation. Most of the previous works in other SLs, such as Morrissey (2008) and Cox et al. (2002a), were created through strong collaboration with the local deaf community. However, the existing ArSL research in the field of ArSL translation shows a poor or weak relationship between the Arab deaf community and researchers. We can sum up all the existing works into the following:

Mohandes (2006) developed a web application for translating an Arabic text to ArSL. Their application simply received an Arabic input and then searched the signs dictionary for each words of the input. Once the application finds a match, it replaces it with the corresponding sign video clip. If no match occurs, it translates the word using Arabic sign alphabets. The produced signs will have the same order as the Arabic word input. Their application uses only a sign dictionary and the number of signs in this dictionary was not mentioned in the published work. This application is clearly making word-to-sign replacements without regard to ArSL's unique linguistic characteristics, such as its own grammar, structure, and idioms. The researchers did not conduct any evaluation of their application.

Halawani (2008) proposed his implementation design of a translation system on mobile devices for Arabic text to ArSL. The working scenario of his system is as follows:

1. The user enters a text input on his mobile via the installed translation system on the user device.

2. Then, this input text is sent to a web server translation application.

3. This server application has a sign dictionary so it can make word-to-sign replacements like the previous work. It also translates using a sign alphabet for a non-existing word match.

4. The generated signs are sent back to the user's device.

Al-Khalifa (2010) proposed a system similar to Halawani (2008), except that there is no need for an Internet connection since the translation system is installed in the user device. However, her system only accepts 50 letters input as a maximum. In addition to the shared misconceptions with all the previous systems, we downloaded her system and we found that it contains only 15 signs in its dictionary.

Ameiri et al. (2011) developed a system similar to Halawani (2008) and Al-Khalifa (2010), except that it translates from Arabic SMS message text to ArSL. Their system uses a sign dictionary and doesn't employ rules or a corpus.

The Tawasoul translation system was developed as Masters' project by Al-nafjan (2008). Tawasoul is an Arabic word that means *interactivity*. This system has three features. First, it has a dictionary that can be browsed. The signs in this dictionary are categories. The second feature is that it makes a finger-spelling translation. The third feature is the translation that makes word-to-sign replacements like the above systems. However, if the word does not exist, the system uses Buckwalter's morphological analyser to get the stem and root. Therefore, first it searches for a matching stem in the dictionary; if nothing is there, it will search for a root; if nothing is there, it will do a finger-spelling translation (sign alphabet). The interesting part of Tawasoul is that the generated sign will not be presented as a sign video clip, but using a signing avatar. She used Vcommunicator Gesture Builder 2.0 for creating signs; see `http://www.vcom3d.com/vcommunicator.php`. Then, she used Smith Studio for exporting these signs to MOV video files which are converted to flash files to reduce the size of the signs. These translation features share the same misconceptions about ArSL with the systems above and assume the translations depend on the Arabic language. On the other hand, the other two features are valuable for hearing learners.

3.3 Notation Systems

As noted in Section 2.2.1, SL has no standard written form. Since 1960, many unsuccessful attempts have been made to develop a standard writing form (Kato, 2008). Several methods of notation exist for SL, and each was invented for a particular reason. Stokoe (1960) developed one of the earliest notation systems, which utilises a linear representation of signs. The structure of this representation starts with the *tabula* (sign location), followed by the *designator* (the hand shapes), and then the *signification* (orientation and movement). It uses special character symbols for signification and alphabet letters for tabula and designators. The problem with this system is that it does not represent NMFs. In addition, it does not have enough hand shape letters to describe all possible gestures in SL.

Therefore, recent SL resources and avatars do not employ this system and do not consider it the correct choice for their needs.

The Stokoe system has opened the door for more attempts to develop greater efficiency, leading Kakumasu (1968) to propose another system called Kakumasu's notation system. In addition to Stokoe and Kakumasu, Friedman considered the first researcher who analysed ASL from a phonological point of view invented Friedman's notation system (Kato, 2008).

Next, we will provide a description of three dominant sign notation systems: Sutton SignWriting, Hamburg Notation System (HamNoSys), and gloss notation. We will show the strengths of each and discuss their differences.

3.3.1 Sutton SignWriting

In 1974, Sutton created SignWriting (Sutton, 2002). The inspiration for Sutton SignWriting came from her dance notation system that represents movements as they are visually apparent in a format that is easy to read and understand. Sutton SignWriting is language-independent (see Figure 3.10); most international SLs, such as ASL, ArSL, and Brazilian Sign Language (LIBRAS), can be represented in the SignWriting system. The system is used in everyday communication such as writing letters and e-mails, articles for magazines and newspapers, and SL dictionaries. Symbols are written in relation to the form they actually take in Sutton SignWriting, whereas characters in HamNoSys are written linearly from left to right. In order to represent the signs, Sutton SignWriting uses a combination of hand shape, palm orientation, and hand movement. There is no known translation system using this type of notation.

3.3.2 Hamburg Notation

In 1998, HamNoSys was made publicly available (Hanke, 2004). It was developed by a group of hearing and deaf researchers at the University of Hamburg in Germany. HamNoSys is not intended for everyday communication. It was designed for research into sign generation, machine translation, and SL dictionaries. HamNoSys is based on the linear representation of signs and uses a standard structure of special characters (see Figure 3.11). The structures used to depict SL are based on hand shape, palm orientation, location, and movements.

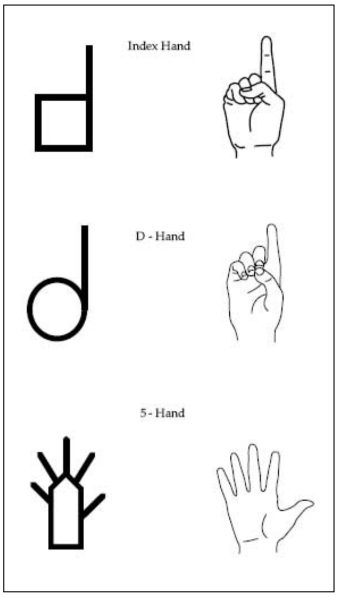


Figure 3.10: Some Sutton SignWriting hand shapes (Sutton, 2002).

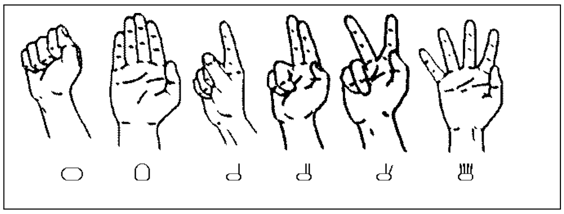


Figure 3.11: Some HamNoSys hand shapes (Prillwitz et al., 1989).

HamNoSys system is insufficient to express all of the NMFs, such as the movement of the head or body or changes in gaze. Another limitation is its inability to represent multi-channel signs. In case of an overlap between signs during SL performance, there is no way to synchronise the time between signs. Therefore, researchers have combined the HamNoSys for MFs with gloss notation for NMFs to overcome these limitations.

The group developing this notation is continuing to overcome its limitations. This notation is widely used in SLMT, especially in Europe.

3.3.3 Gloss Notation

Gloss notation describes signs by providing the meaning of the sign using an English word or several hyphenated English words (for the ArSL corpus which

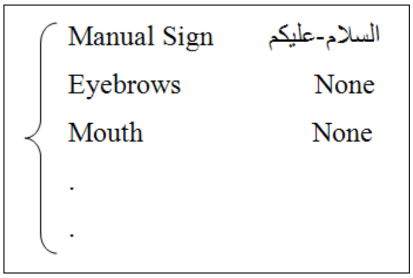


Figure 3.12: An example of multi-gloss notation. The two words "السلام عليكم" have one equivalent sign in ArSL.

will be described later in Chapter 4, Arabic words have been used instead of English words). It describes all NMFs, such as raising the eyebrows, as well as MFs. It is a simple system since the other notation systems are complex and require a great deal of time to learn. Multi-channel signs can easily represented by multi-gloss notation. This notation system has been used in our ArSL corpus (see Chapter 4). Figure 3.12 presents a simple example of gloss notation.

3.4 Related Corpora

An annotated SL corpus is useful for SL studies and learning, and is essential for corpus-based machine translation systems. For this reason, many SL corpora have been developed. Unfortunately, none of these corpora are for ArSL. In this section, we will survey some of the existing general purpose non-ArSL corpora that are primarily used for language studies and learning, and those specifically designed for MT systems.

3.4.1 General Purpose Corpora

This Section surveys of some of the existing general purpose corpora primarily used and designed for SL linguistic studies and for SL learning. As will be shown, the methodologies and tools for use are similar for these corpora.

3.4.1.1 European Cultural Heritage On-line SL Corpora

In 2002, the European Cultural Heritage Online (ECHO) was established to 'create an infrastructure to bring cultural heritage on the Internet, and build up a network of institutions, research projects and other users which provide content and technology for the common infrastructure, with the aim to enrich the "agora" and to create a future Web of Culture and Science'. ECHO is funded by the European Commission. One part of the ECHO project targets SL, resulting in many SL resources and tools being introduced, such as the well-known annotation software ELAN (Brugman and Russel, 2004), which stands for the EUDICO Linguistic Annotator (EUDICO stands for European Distributed Corpora). In addition, ECHO supports and publishes a fully annotated SL corpus for three European SLs: BSL, constructed by Woll et al. (2004), SL of the Netherlands by Crasborn et al. (2004) and Swedish SL by Bergman and Mesch (2004). Each corpus was constructed from five stories, each translated into all three SLs. Each story was signed by one signer and contains a large number of signs. Each sign is then divided into tiers that represent NMF, such as cheeks, eyes and mouth, or MF, the left and right hands. The representations of NMF and MF use gloss notation and include a translation to natural language.

Figure 3.13 provides a sample from the ECHO data corpora. ELAN was used as the annotation tool for this project. It was developed to analyse sign languages, gestures and natural languages and can handle video and audio data. The corpora can be accessed on-line via the project's website (`http://www.let.kun.nl/sign-lang/echo/`).

3.4.1.2 Irish SL Corpus

The Centre for Deaf Studies at the School of Linguistics, Speech, and Communication Sciences at Trinity College in Dublin built an ISL corpus (Leeson et al., 2006) containing children's stories. The corpus took approximately three years to build and 40 signers were involved. The participants' ages were between 18 and 65, and they came from different regions in Ireland. The recorded videos are approximately 20 hours long, and were annotated using ELAN. The sign sentences were divided into tiers that represent the MF and NMF features in gloss notation. In addition, English sentence translation was included for each sign sentence.

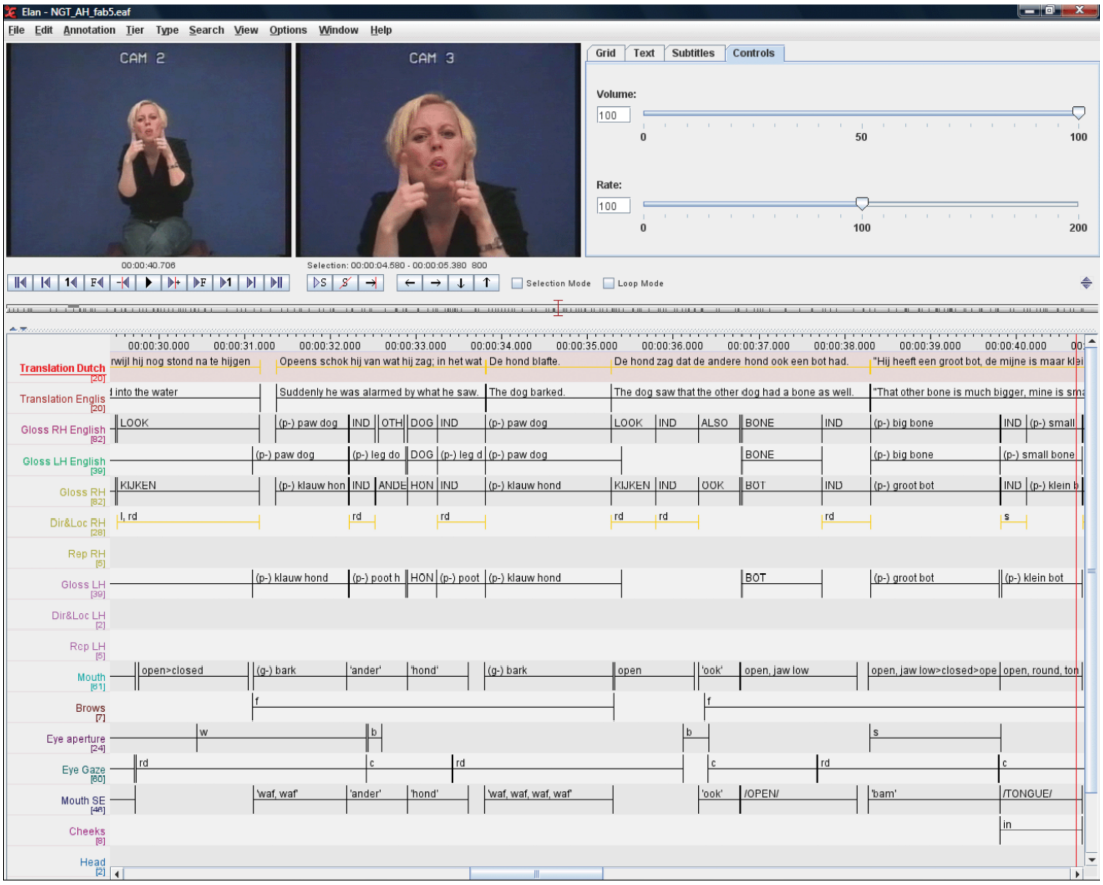


Figure 3.13: Screenshot from one story in the ECHO data corpora.

3.4.1.3 British SL Corpus Project

This project received funding from the Economic and Social Research Council (ESRC) to conduct a large BSL corpus (Schembri, 2008). The project began in January 2008 and was planned for completion in 2010 but was extended to June 2011. The main objective of the project is to collect a large corpus that allows researchers to understand BSL by conducting investigations of aspects of vocabulary, grammar and sociolinguistics. The total number of collecting sentences is unknown. However, 249 deaf signers participated in collecting this corpus from eight different cities across the UK. In addition, these participants belonged to different age groups. The project classified ages in the three different groups of 16-40 years, 41-65 years and 65+ years. For quality control, the filming was done without any hearing people present against a blue background screen using two lights. The signers wore plain-coloured clothing and used chairs with no arms. A high definition video camera was used, and filming sessions usually lasted for about sixty-five minutes, including five minutes for telling personal experience stories as a warm-up activity. Then, thirty minutes was used for free conversation, followed by a twenty-minute interview and ending with a ten-minute vocabulary task. The corpus was partly annotated using the ELAN annotation tool. The conducted video can be accessed on-line through the project's website (http://www.bslcorpusproject.org/data/region/). However, ELAN annotation files have not been published yet and the corpus has yet to be used for machine translation experiments.

3.4.1.4 Netherlands SL Corpus Project

The Netherlands Organisation for Scientific Research funded this two-year project to collect data from deaf signers using Netherlands SL, known as NGT (Crasborn et al., 2008; Crasborn and Zwitserlood, 2008). The work was completed in 2008. The project targeted five groups. The first is SL researchers, as the corpus enables them to study SL linguistics from different aspects. The second is NGT teachers who lack NGT material, as they can benefit from the collected videos within the corpus to help NGT learners practice the language in different situations. Third, interpreters will find this corpus useful because it has been collected from different regions, allowing them to get to know and practice the language of a particular region. Because the corpus includes many discussions and arguments among the deaf on different issues, the fourth group includes deaf individuals who can learn about these differing opinions on various NGT issues such as language standardisation. Finally, parents of deaf children can watch the included signed stories with their children. The conducted data consist of synchronous recordings using multiple video cameras. The data contain more than 2,000 sign clips categorised in regions. The ELAN tool was used for annotating the conducted data; Figure 3.14 shows a screenshot of annotated NGT using ELAN.

3.4.2 MT Corpora

This Section presents a survey of some MT corpora that have been collected specifically for SL translation systems and designed to be accessible by a computer for use in translation processes.

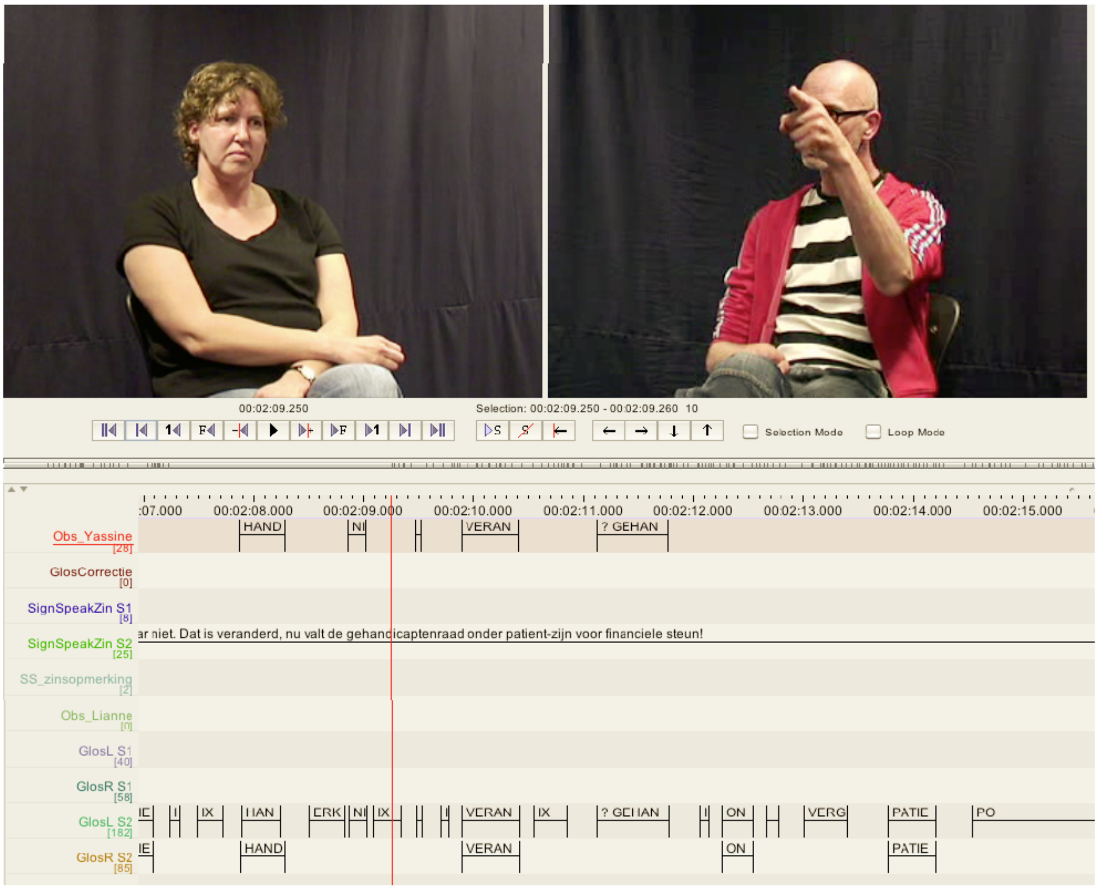


Figure 3.14: Screenshot of ELAN screen showing the annotating details of the NGT example.

3.4.2.1 Weather Cast DGS Corpus

Bungeroth et al. (2006) built a German sign-language (DGS) corpus of the weather report domain. They constructed their corpus by extracting the German subtitle text and DGS translation from a German daily weather news television channel called Phoenix broadcasts. The signs were collected by extracting the lower right corner of the broadcast frame, which shows the DGS interpreter, and ELAN was used to analyse the DGS sentences. These sentences were separated into five tiers: gloss notation of the sign sentences, word classes (such as verb, noun, adjective and adverb), DGS sentence boundaries, German sentence translation and German sentence boundaries. Two thousand four hundred sixty-eight sentences were collected and the corpus was designed for statistical machine translation and sign recognition. The problem with this corpus is its low-quality sentences compared with other corpora because the sentences were extracted from live weather reports, which failed to apply quality control. Also, it is worth mentioning that I met researchers from RWTH University at the second SLPAT Workshop in Edinburgh and was told that they only used about 700 sentences for the training datasets because of quality issues.

3.4.2.2 American Sign Language Linguistic Research Project (ASLLRP)

The National Science Foundation funded this project, which aimed to collect ASL sentences to facilitate both linguistic and computational research on ASL. The project began in 1999 at the National Centre for Sign-Language and Gesture Resources of Boston University (Stein et al., 2007) and Rutgers University. This project recorded signs simultaneously using one colour video camera and three black-and-white video cameras. Two of the black-and-white cameras were placed in front of the signer to record his entire front side, and the colour camera was placed between the two cameras and zoomed to record the signer's face. The last camera was placed at the side of the signer. To annotate the ASL videos, ASLLRP developed annotation software called SignStream (Neidle, 2002, 2007). The collected corpus was released in August 2007. Two signers were recorded in this corpus. Ben Bahan signed the following three stories: 'Close Call', 'Speeding' and 'Three Pigs'. Mike Schlang signed twelve different stories, including 'Accident', 'Biker', 'Boston, LA', 'Ali', 'Dorm Prank', 'Whitewater', 'Football', 'Movies', 'Siblings', 'Roadtrip1', 'Roadtrip2' and 'Scary Story'. Figure 3.15 shows a SignStream screenshot from the corpus.

3.4.2.3 Air Travel Information System (ATIS) Corpus

Bungeroth et al. (2008) collected a SL corpus for the air travel information domain. The corpus includes three SLs: ISL, DGS, and South African Sign Language (SASL). The ELAN annotation tool was used to analyse the corpus. English and German gloss notation was added to the annotation. This corpus contained 577 sentences of the three SLs.

3.4.2.4 Czech SL Corpus

Campr et al. (2008) built a Czech SL corpus for testing SL recognition systems. The domain of this corpus is the train timetable dialogue and it is being translated

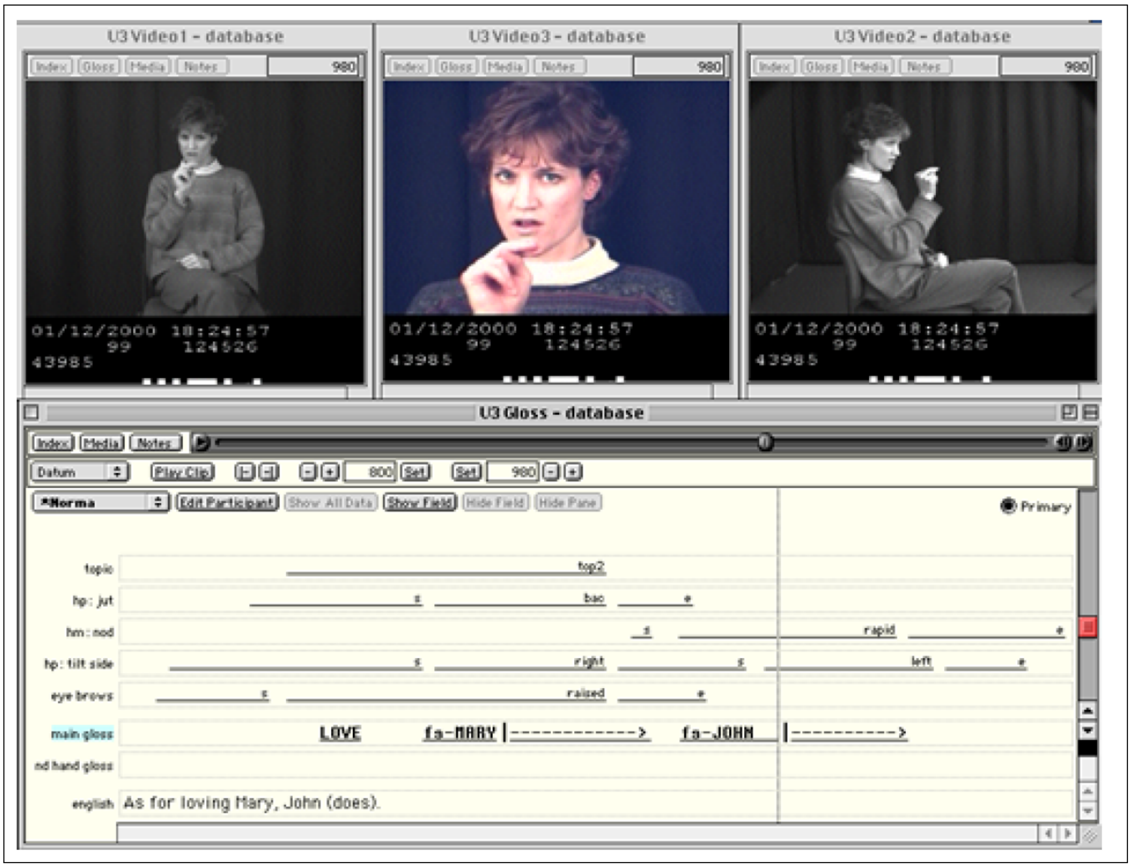


Figure 3.15: SignStream screenshot from the ASLLRP corpus.

into Czech SL. Four signers were recorded from three different camera angles. Two angles recorded the whole body and provided 3D movement data on the hands. The third camera focused on the face of the signer to record all NMFs, see Figure 3.16.

The ELAN tool was used to annotate the recording videos. A set of image processing algorithms was used for MF and NMF extraction. Each signer performed 378 of the 1,512 signs collected.

3.5 Sign Synthesis

Because SL is a visual language without a standard writing system, it is necessary to find a way to allow text–to–SL MT systems to present the translated SL output in space. Some attempts have been made to translate text into SL notation systems, such as the work of Morrissey (2008) and Othman and Jemni (2011),

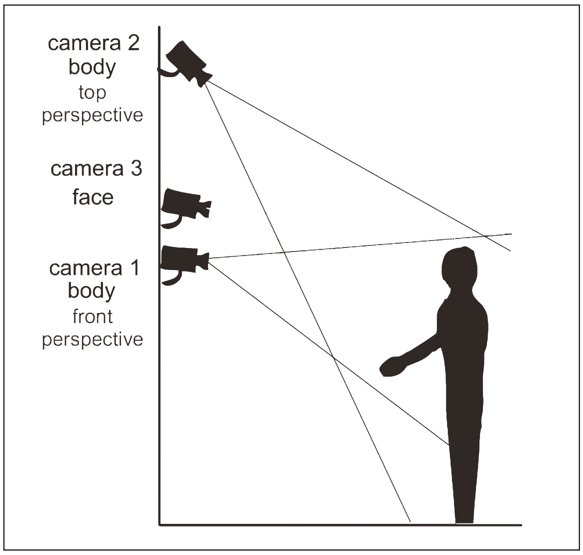


Figure 3.16: Illustration of the three cameras positions. One only camera captures the NMFs while the other two record the whole body.

who translated SL to sign glossing text. However, without presenting signs in space, these notations are not usable by the deaf and they do not allow a direct human evaluation of the translated signs' output of MT systems because most deaf people are not familiar with the existing notation systems. A study conducted by Naqvi (2007) shows BSL signers prefers video-based sign synthesis over signing avatars and SignWriting notation. This is likely because avatars are still in the development stage and they lack the ability to show NMF details. SignWriting notation, on the other hand, is unfavourable because signers are not familiar with the notation systems.

In this section, we discuss the two types of sign synthesis used in MT systems. Sign synthesis that uses concatenated videos to produce the signs output is easy to engineer, because each sign is isolated, then concatenated to other signs to produce a sign sentence. However, this method is limited to the recorded signs in the corpus. In addition, the delivered sentence is bound to be stilted, no matter how much effort is put into engineering, because the delivered sentence is produced from isolated signs, and there is no way to make the transition movement between these signs natural. In addition, concatenated videos require high quality control to ensure that the recording is done under the same conditions, such as lighting, clothes, and camera view. Some programs aim to make sign sentences look more natural by employing dynamic algorithms to address the problem of how to fill the transition movement between two isolated signs. The second type of sign synthesis is based on the use of avatars, which are animated signing characters. This method is more flexible than video-based sign synthesis, because it can perform any sign and does not require high quality control. In addition, with enough engineering, avatar sign synthesis can increase the natural appearance of sign sentences. However, this method is much harder to engineer than concatenated videos. Finally, the study of avatar sign synthesis is limited and avatars are still in a development stage which produces high error rates, so we choose to produce our translation output using concatenated video.

3.5.1 Concatenated Video-based Sign Synthesis

Using concatenated video clips of a real signer requires quality control. The clips required to produce a full sign sentence must belong to the same signer. In addition, all of these sign clips should be recorded under the same conditions, such as lighting and camera view. All of these quality factors aim to present fluent and clear signs to the viewer. The challenge of using concatenated sign video is in dealing with the transition movement problem. Figure 3.17, which illustrates this problem, shows a recorded sign sentence and determines the boundaries of each sign by showing the end point of Sign 1 and the starting point of Sign 2. In order to display a realistic transition movement, which is the movement between signs, the movement of the hand should start from the ending of Sign 1 and end at the beginning of Sign 2; otherwise, the sign sentence will not be clear or fluent. Solina et al. (2001) and Chuang et al. (2006) have made attempts to generate smooth and fluent sign sentences.

Solina et al. (2001) introduced two methods for producing a smooth sign sentence video from individual sign video clips. The first method starts and ends each individual sign at the same position: the centre of the signer's body, see Figure 3.17.

As seen in Figure 3.17, because the palm position and orientation are the same as the end of Sign 1 and the beginning of Sign 2, there is not a transition movement problem between clips. Therefore, joining these sign clips together to produce a full sign sentence will require no further processing. The drawback of this method is that the producing sign sentence video would present an unnatural

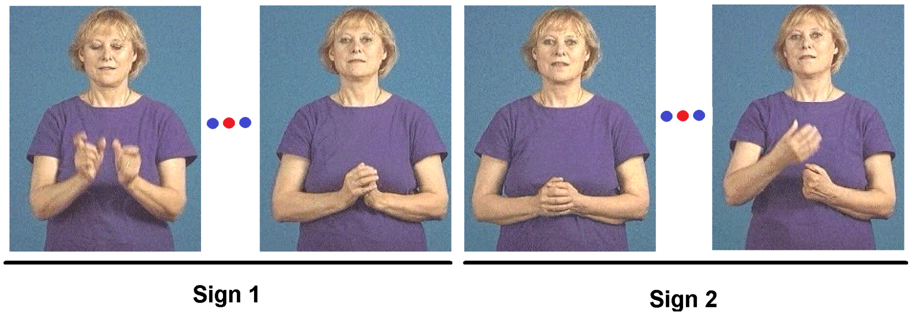


Figure 3.17: An illustration of the first method from France and Slake (2001) where the ending position of the palms in Sign 1 is the same position as the beginning of the palms in Sign 2.

way of signing even though the video is smooth. This method has been used by Sarkar et al. (2009) in their Bangla text to Bangla SL translation system.

The second method is to make joins based on four criteria, but before that it is necessary to pre-process all sign video clips to extract palm positions and orientations using computer vision methods invented by Marshall and Martin (1992) and Klette and Zamperoni (1996), and then saving the extracted information in a file associated with the video clip. During run time, when the system decides to produce a new sign sentence, the system call function called *DIFF* will receive $Sign_{(X)}$ and $Sign_{(X+1)}$ video clips and determine the optimal transition point in $Sign_{(X)}$ to join it with $Sign_{(X+1)}$. This function is based on four joining selection steps:

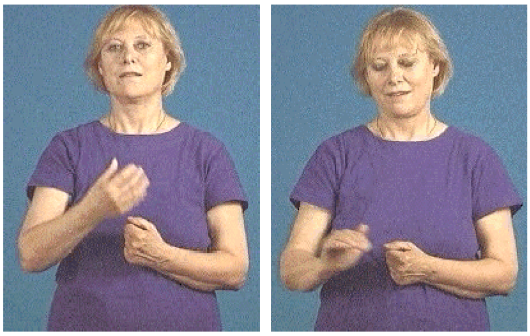
1. It tries to select joining points where the arms in the same position as the starting arms positions of the $Sign_{(X+1)}$ clip; see Figure 3.18(a).

2. If the previous case is not possible, the nearest arm positions of the starting arm positions of the next sign clips are selected. The distance between the selected position and the start position of the next clip must not exceed a certain value; see Figure 3.18(b).

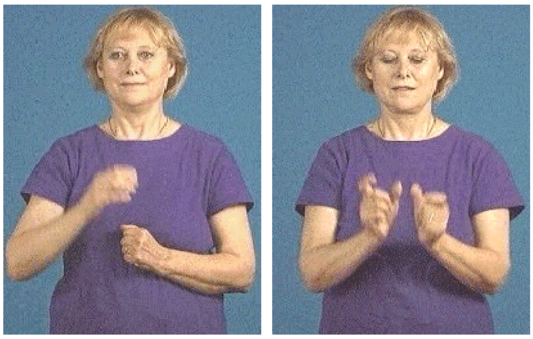
3. If it does exceed the distance value in the previous case, the function selects the position of palms when it is over the chest, as it shown in Figure 3.18(c).

4. When the previous step does not occur, the system selects the palms positions nearest to the palms position in the starting clips; see Figure 3.18(d).

(a)



(b)



(c)

(d)

Figure 3.18: Examples of automatic joining of video clips (Solina et al., 2001).

The system calls upon this function for each neighbour sign clip in order to combine all the clips to produce the final sign sentence. The problem with this method is that it is not fluid and the movement of the hands is not always smooth throughout the sign sentence.

Chuang et al. (2006) invented another method that searches through all sign video clips to find a suitable transition between the joining sign clips. The selection process is based on the non-uniform rational B-spline approach (Piegl and Tiller, 1997). The problem with this method is that it must have a very large clip database in order to find all possible transition movements between signs. In addition, the process of selection has a high performance cost. Furthermore, extracted transition movement may carry some NMFs that are not necessary for both clips, which can disturb the viewer.

3.5.2 Avatar

Many avatars have been developed in the past decade, and many of them have shown promising results. The research has progressed to produce more fluent and realistic signing avatars that can demonstrate smooth MFs movement. They have also started to show some NMFs. There are several benefits to using avatars, such as allowing users to select their preferred avatar character. Also, unlike video-based sign synthesis, avatars are not limited to a strict quality-control condition such as using the same signer.

In this section, we survey some well-known avatars. We present them in terms of how they can be beneficial to the MT system. Therefore, the challenges of creating such an avatar will not be included in this survey. We classify these avatars into two types. The first type includes avatars that receive standard input format such as XML for presentation. This type is very useful because any SL translation system can benefit from them. The second type includes avatars built exclusively for a specific MT system that cannot be used by other systems.

3.5.2.1 Integrated Avatar

The following is a survey of some well-known avatars. These types of avatars can receive signs input in written form as linear notation texts or in a predefined XML format, and then perform the corresponding signs in space.

- University of East Anglia (UEA) Avatar: UEA is one of the leading research groups in the world in sign synthesis and sign translation. Their first attempts to build a signing avatar began when they built a plugged-in avatar for TESSA (Cox et al., 2002a) and "Simon the Signer" (Bangham et al., 2000a). TESSA, which stands for the TExt and Sign Support Assistant, aims to translate a postal clerk's speech to BSL using an exclusive avatar built specifically for this project. The whole system, including the avatar, has been developed as collaboration between the School of Information Systems at UEA, Televirtual Company, the UK Post Office and the Royal National Institute for Deaf People. In 2000, UEA won a gold medal and the overall IT Award at the prestigious British Computer Society's Information Technology Awards.

  TESSA attracted the attention of the European Union (EU), which then funded a new UEA project called ViSiCAST, or "Virtual Signing: Capture, Animation, Storage, and Transmission," under the fifth framework program (Bangham et al., 2000a). The project was funded from 2000 to 2002 (Elliott et al., 2008). The main goal of this project was to study the feasibility, in principle, of building a prototype system for translating a text into SL animation (Elliott et al., 2008). In addition, it identifies the aspects of deaf life that would be improved if SL communication with hearing people was available, such as access to public services, learning services, and the web. During the project, the researchers decided to build an interface for their avatar to allow other systems to use it. They chose HamNoSys notation as the standard input format for this interface. However, HamNoSys has to be converted into an XML format based on the HamNoSys phonetic model, a format known as Signing Gesture Markup Language (SiGML). For this purpose, they built a conversion tool. Figure 3.19 shows the "Mug" sign in both HamNoSys and SiGML.

  After the completion of the ViSiCAST project, UEA received additional funding from the EU for another project called eSign, which ran from 2002 to 2004 (Elliott et al., 2008). It aimed to provide tools and resources to create low-cost SL performance content for websites. As a part of this project, UEA built the VANESSA system (Glauert et al., 2006), or Voice Activated Network Enabled Speech to Sign Assistant. It is a communications system

(a) "Mug" sign in HamNoSys

```
<hamgestural_sign gloss="mug">
  <sign_nonmanual>
    <mouthing_tier>
      <mouth_picture picture="mVg"/>
    </mouthing_tier>
  </sign_nonmanual>
  <sign_manual>
    <handconfig handshape="fist"
        thumbpos="across"
        extfidir="ol" palmor="l"/>
    <location_bodyarm
        location="shoulders"/>
    <par_motion>
      <directedmotion
          direction="u" curve="u"/>
      <tgt_motion>
        <changeposture/>
        <handconfig
            extfidir="ul" palmor="dl"/>
      </tgt_motion>
    </par_motion>
  </sign_manual>
</hamgestural_sign>
```

(b) "Mug" sign in SiGML

Figure 3.19: An example of "Mug" sign presented in (a) HamNoSys and (b) converted from HamNoSys to SiGML (Elliott et al., 2010).

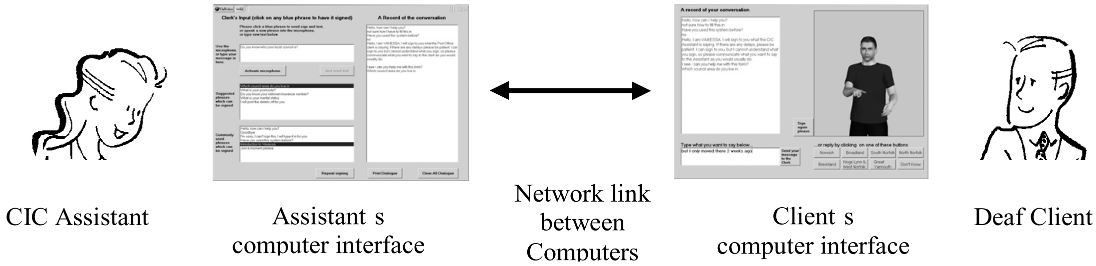


Figure 3.20: An overview of the VANESSA system (Glauert et al., 2006).

that connects deaf clients to assistants in an Information Centre. In this system, the assistant sends a text message over the network to the client's screen like chat software. Deaf clients can then see the text in addition to the UEA avatar, which translates the text into BSL. The system, like TESSA, uses a phrase lookup approach for translation. Figure 3.20 shows an overview of the VANESSA system.

Starting from TESSA and continuing today, UEA is improving its avatar. In 2007, they introduced a new version called AnimGen, which includes a large number of hand shapes that can be described in HamNoSys version 4.0. It also overcomes certain animation problems, such as including new NMFs and avoiding collisions between body parts (Elliott et al., 2008). In addition, it has the ability to sign in real time. The advantage of this new version is that it provides APIs to allow real–time HamNoSys/SiGML–to–Sign animation services via a standalone application or a web application. The JASigning (Java Avatar Signing) (Figure 3.21) is a SiGML Player Java Applet and the JASigning's animation generation module is Animgen.

- University of West Bohemia (UWB) Avatar: This has been developed and improved by the Department of Cybernetics at UWB since 2004 (Zdeněk et al., 2011). As a part of the Czech text to Czech SL (CSL) translation system, they built a signing avatar. This avatar has been developed separately and integrated with the CSL translation system via the avatar connection interface which receives HamNoSys notation as an input. The received HamNoSys is converted into specially designed trajectory signs using a rule-based conversion parsing system; see Figure 3.22. This conversion system uses more than 300 rules (Zdeněk et al., 2011).

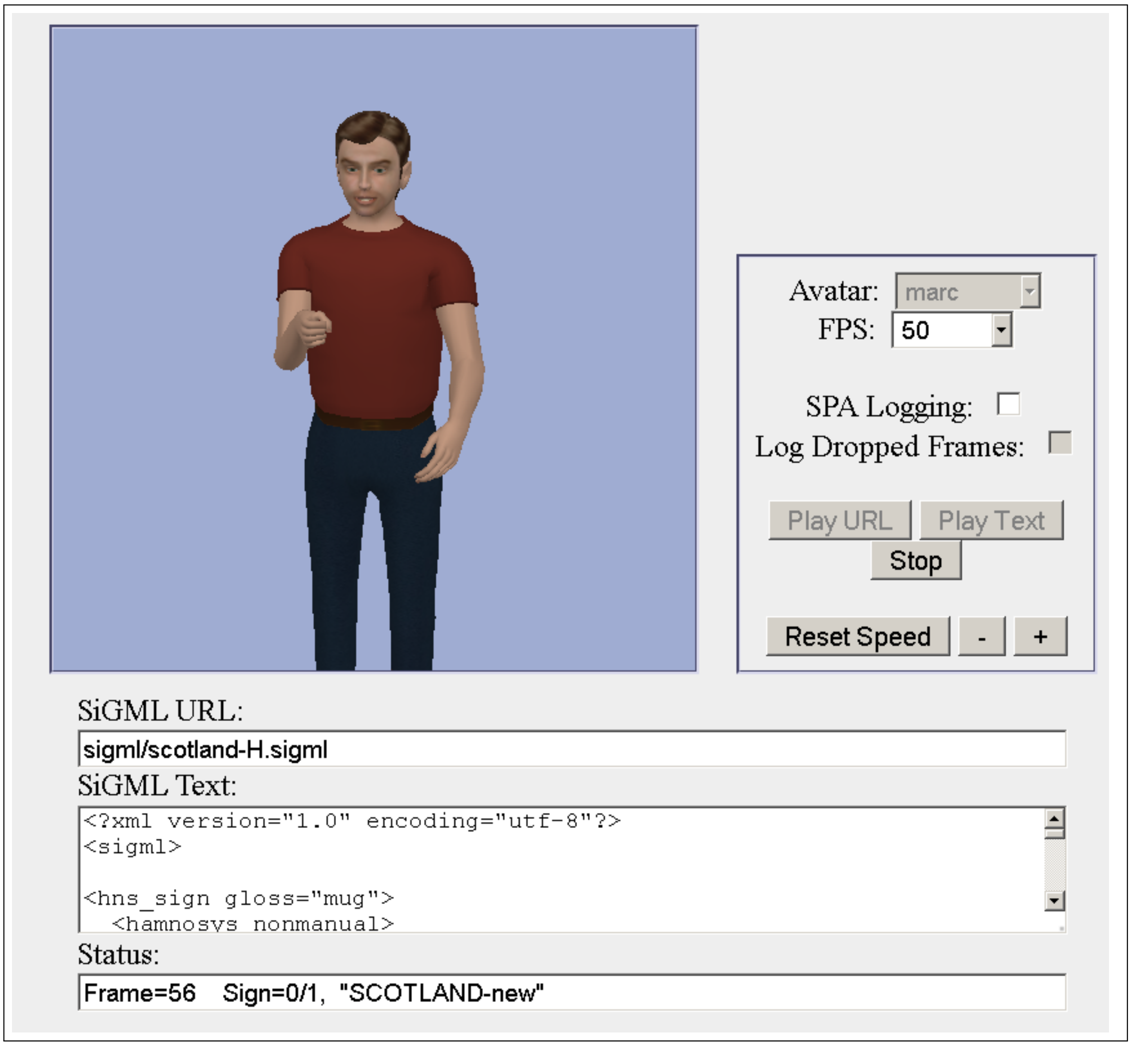


Figure 3.21: A screenshot of JASigning player performing the “mug” sign.

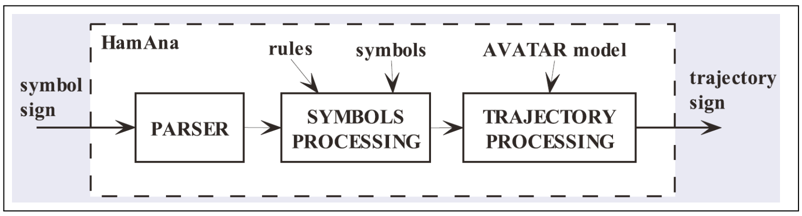


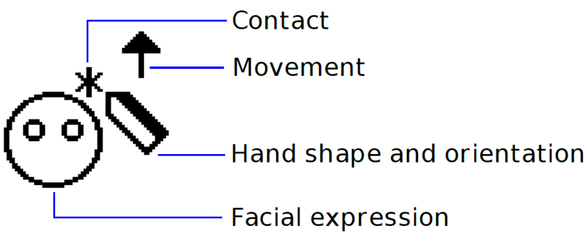
Figure 3.22: The UWB Conversion system based on rule-based parsing (Zdeněk et al., 2011).

In addition to MFs, the avatar provides some NMFs such as movements of the body, face, and mouth. The avatar was developed using C++ and OpenGL rendering. The avatar is still an ongoing development to show more realistic signs, including showing more NMFs.

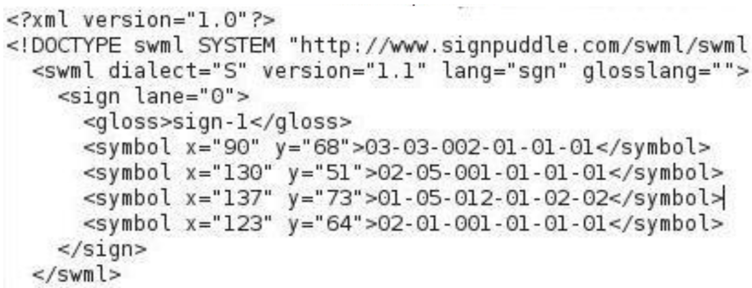
- SASL Project: Moemedi (2010) built an avatar as part of the SASL project at the University of the Western Cape. The avatar receives SignWriting Mark-up Language (SWML) input that is an XML-based format of the SignWriting notation system introduced by Sutton in 1974 (Sutton, 2002; Moemedi, 2010). Figure 3.23 gives an example of SignWriting in (a) and then it shows its SWML representation in (b). Then, it performs the sign animation. The benefit of this avatar is that it can perform any sign that can be written in SignWriting notation. In addition, the researchers conducted an evaluation test of their avatar with help from SignWriting's experts. This test consisted of eight signs. It received 62% animation performing accuracy and 82% recognition rate (where the experts were able to recognise the performed sign). This test was preliminary and the avatar is still in its infancy. It needs a comprehensive evaluation test of its capabilities and then the researchers can address its weakness in order to improve it.

3.5.2.2 Exclusive Avatar

There are some of the avatars that have been built exclusively for a specific MT system. These types of avatars cannot be used by other systems. ATLAS Project is one of these avatars. This project, which stands for Automatic Translation into Sign LAnguageS, is a three-year project that started in 2009 and is scheduled to finish in 2012 (Lombardo et al., 2011). It is funded by the Region of Piedmont of Italy within the Programme Converging Technologies. The aim of this project is to build an Italian text to Italian SL translation system. As part of this project, the researchers built a signing avatar that received a glossing input text from the translation system, called AEWLIS (ATLAS Written Italian Sign Language). This AEWLIS system consists of a sequence of lemmas associated with a description of the meaning of each lemma and its syntactic number and the link to the corresponding sign. Also, the gloss includes other information such as the initial/final locations. The avatar is still in the development stage and has not been fully evaluated yet. Figure 3.24, shows the avatar's components.



(a) “Mug” sign in HamNoSys



```
<?xml version="1.0"?>
<!DOCTYPE swml SYSTEM "http://www.signpuddle.com/swml/swml
  <swml dialect="S" version="1.1" lang="sgn" glosslang="">
    <sign lane="0">
      <gloss>sign-1</gloss>
      <symbol x="90" y="68">03-03-002-01-01-01</symbol>
      <symbol x="130" y="51">02-05-001-01-01-01</symbol>
      <symbol x="137" y="73">01-05-012-01-02-02</symbol>
      <symbol x="123" y="64">02-01-001-01-01-01</symbol>
    </sign>
  </swml>
```

(b) “Mug” sign in SiGML

Figure 3.23: The “Hello” SASL sign is presented in (a) SignWriting and in (b) its SWML representation. (Moemedi, 2010).

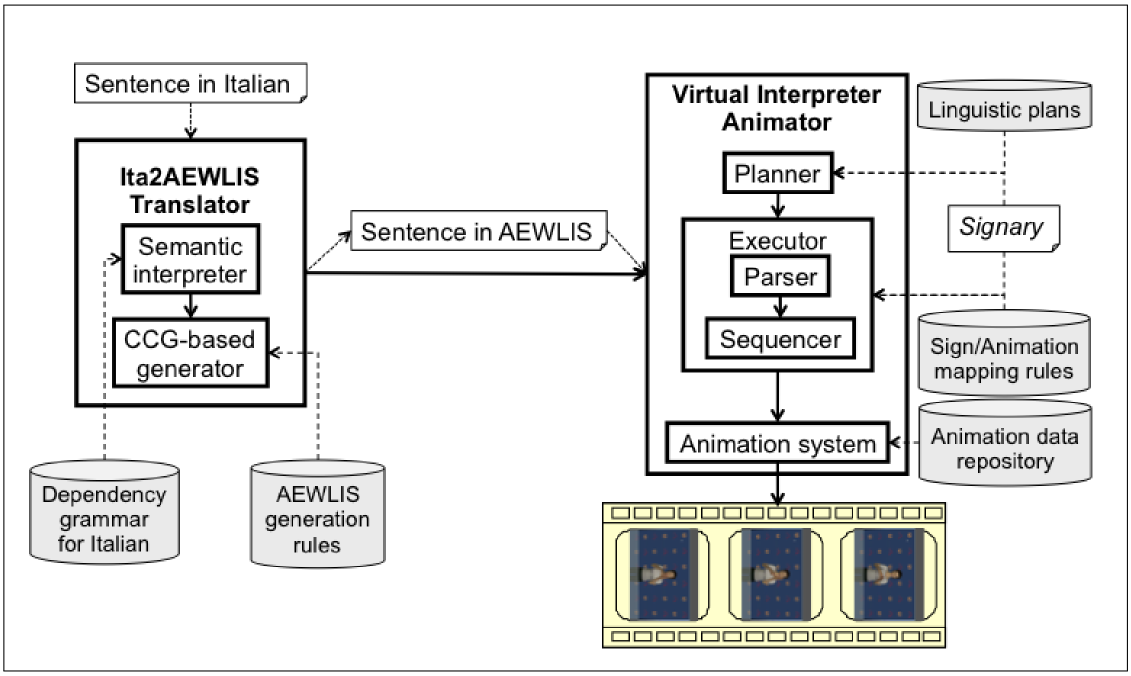


Figure 3.24: A component overview of the ATLAS system (Lombardo et al., 2011).

3.6 Translation Evaluation Techniques

Widely used automatic evaluation metrics have been designed to evaluate written language using a linear representation form, i.e., word by word. The problem with these metrics is that they fail to address multi-channel representations that include all of the details of MF and NMF. Figure 3.25 shows examples of linear and multi-channel representations. One way to solve this problem is to combine each sign's MFs and NMFs as a solid block and discard NMFs to enable existing metrics to be applied directly to these blocks as words. Many researchers follow these same methods such as Stein et al. (2007), Morrissey (2008), Othman and Jemni (2011) where they treat the sign as one block, use a metric such as WER to evaluate signs and call it Sign Error Rate. In many cases, as explained in Almohimeed et al. (2009), when the analysts treat a sign as a single block, for example, when an essential NMF is discarded, the metric score shows unrealistic results. Another way to overcome the metrics problems is by modifying these evaluation metrics to take care of all MF and NMF details. In Chapter 6, we will introduce our evaluation technique that uses this method.

	Sign 1	**Sign 2**
Right Hand	THEFT	FORBIDDEN
Left Hand		
Eye	closed	

Figure 3.25: This figure shows the sentences 'Theft forbidden' in ArSL. It has a multi-channel representation in ArSL.

Chapter 4

Building ArSL Corpora

As mentioned in Section 2.3, many efficient machine translation approaches have been proposed in recent years (e.g., SMT and EBMT). These approaches are corpus-based, which means that the accuracy of translation directly correlates to the size and coverage of the corpus. The corpus is a collection of translation examples constructed from existing documents such as books and newspapers. However, constructing an SL corpus is not a straightforward task. It is a complicated procedure due to the reasons previously cited. For example, no SL documents or standard writing system exist. As discussed in Section 3.4, a number of SL corpora have already been completed and used for SLMT. Unfortunately, no existing ArSL corpus is available. Therefore, we constructed two ArSL corpora. These corpora were heavily influenced by Saudi local SL due to the background of the signing team. This Chapter will present the two constructed corpora and their construction process.

4.1 Considerations

SL corpora are unlike the much more common corpora between written languages (French, English, Japanese, etc.) in that SL is visual and has no accepted, standard written form. Therefore, instead of a parallel text, we have parallel corpora of the text of the source language associated with the target SL language as video signs. This means that constructing ArSL corpora poses some novel and difficult problems not seen in more conventional written language corpora. One of the problems not seen in written language is the non-existence of SL documents. Therefore, to collect signs, members of the deaf community and SL experts must be involved to produce a high standard SL by signing a clear SL in a natural way. Another purpose of the collaboration is to make sure all deaf age groups can understand these signs. The next step is assigning and annotating these collected signs with the text of the source language, Arabic, using a special purpose annotation tool. Luckily, several SL annotation tools have been developed for annotating some SLs. In Section 4.1.1, we will show our selected tool has been used in annotating our ArSL corpora. For transcription of these collected signs using the selected annotation tool, one of the existing notation systems will be used. The selection will be based on time and cost factors. Section 4.1.2 will discuss our decision regarding the notation system used in our corpora. Annotating SL is not enough to prepare corpora for MT. Therefore, a compiling tool for delivering and preparing MT corpora from the annotating corpora is required. In Section 4.1.3, we introduce the compiling tool we built for this purpose. The size and domain of MT corpus are other issues of concern in building any corpus-based MT system. Section 4.1.4 will discuss our decision regarding these issues.

4.1.1 Annotating Tool

With existing annotation tools, it is possible to transcribe SL videos in a meaningful and machine-readable way. This can help for studying SL with either a qualitative or quantitative approach. In addition, and more importantly, using such a tool will help us prepare MT corpora.

Many tools exist, including Anvil (Kipp, 2001), SignStream (Neidle, 2002), ELAN (Brugman and Russel, 2004). The annotation tool will be a great help to us in constructing the first ArSL corpora. Therefore, we looked at earlier tools' features and determined how they can meet our needs. We ended up deciding to use the ELAN tool. This tool has several attractive features, one of which is that it can link the signs video with the transcription based on the timeline of the video. The sign videos can analysed to a different number of tiers (see Figure 4.1).

Also, the ELAN tool archives this annotation in an XML file, which has an EAF extension file. This EAF file will allow us to later, after annotation, compile and prepare the final MT corpora. Furthermore, one important feature of this

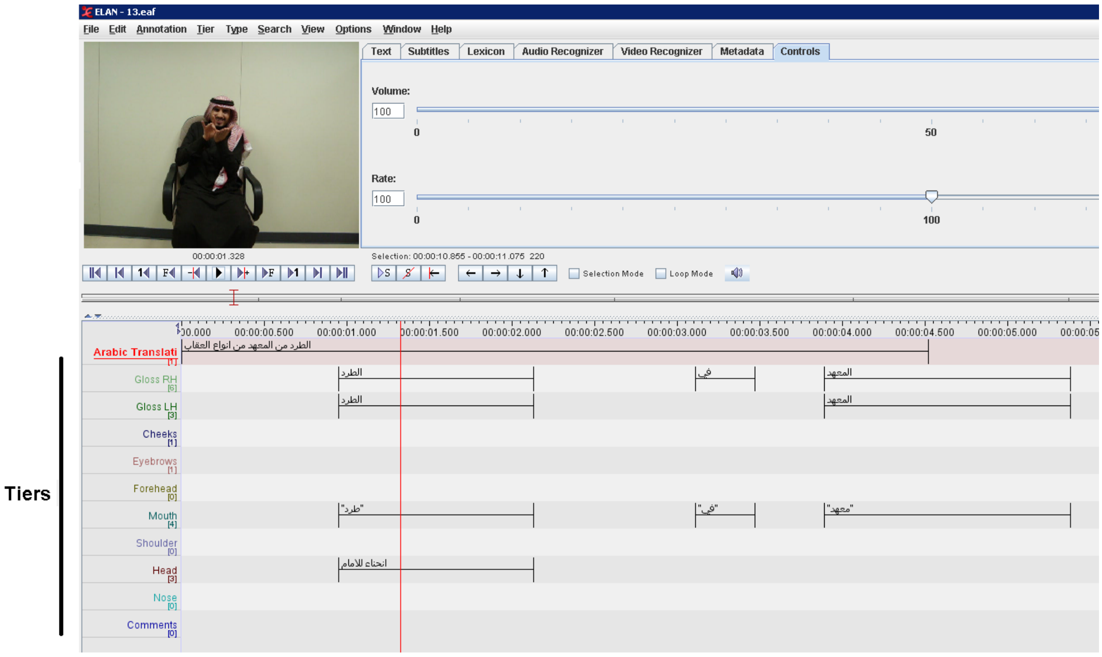


Figure 4.1: A screen shot of an ELAN tool showing one of our ArSL annotating examples. Each sign has clear boundaries of where it starts and ends; in addition, all the MFs and NMFs have been described separately by different tiers.

tool is that it enables us to write glossing text in Arabic script, since it uses UTF encoding so an EAF file is saved as a UTF encoding file format. ELAN is a free and open source tool that can be downloaded (`http://www.lat-mpi.eu/tools/elan/download`). As we have seen in Section 3.4, many corpora have been employed by ELAN for annotation. Therefore, by using it with our corpora, we can share it with many researchers. More importantly, they can benefit from our compiling tool since it is built to parse EAF files.

4.1.2 Notation System

As we described in Section 3.3, the widely used notation systems among researchers are SignWriting, HamNoSys, and Gloss Notation. SignWriting is not suitable for our corpora since our collection team is not familiar with it and we do not see any benefits to including it in our corpora due the non-existence of an efficient SignWriting avatar. On the other hand, HamNoSys uses a UEA avatar. However, at this time, the UEA avatar suffers from MFs collision and cannot perform NMF details (Morrissey et al., 2010). In addition, the team members are not familiar with this notation system, so they would need training to become proficient in using it. Therefore, we decided not to use this notation because we have only a short time to build the corpora; in addition, there are cost limitations to consider. Ultimately, we decided to use Gloss Notation and, in the future, we may add the HamNoSys details to our constructed corpora. Since there is no standard convention for glossing text, we designed our own glossing standard that will be used in our corpora; Section 4.4.2 will describe it. One benefit of using Gloss Notation is that everyone can read it, and in particular, it can be used for linguistic studies. This is especially true in our case; the corpora will have high impact on research in ArSL since there is no existing ArSL corpus that includes a pair of real ArSL and Arabic sentences. The only aid existing at the present is the ArSL dictionary, which is not sufficient for conducting linguistic studies of ArSL. These ArSL corpora may open the door for more ArSL corpora involving different domains and coverages.

4.1.3 Compiling Tool

The annotated corpora that we constructed using the freely available ELAN tool were not yet suitable for use by our built MT translation. Unfortunately, there was no compiling tool to deliver the final MT corpora and help with the subsequent steps in constructing an Arabic text to ArSL corpus. Accordingly, we had to build a special-purpose tool to handle this task. It is written in Delphi XE. To date, this tool has been used only for Arabic, but it may be adapted to other parallel text/SL pairings. This tool (see Figure 4.2) parses the EAF annotation file, isolates each sign clip for the original ArSL sentence videos, and associates this clip with its MF and NMF details. The isolated sign clips have great benefit in terms of producing a concatenated-video translated ArSL sentences.

4.1.4 Size and Domain

One of the concerns we faced before starting to build the two corpora is how many examples we needed to start experimenting with ArSL translation, since adding more examples should improve performance (Sumita and Iida, 1991). Therefore, we did a survey of a number of working EBMT systems that use a small number of real examples. Table 4.1 shows text-to-text EBMT translation

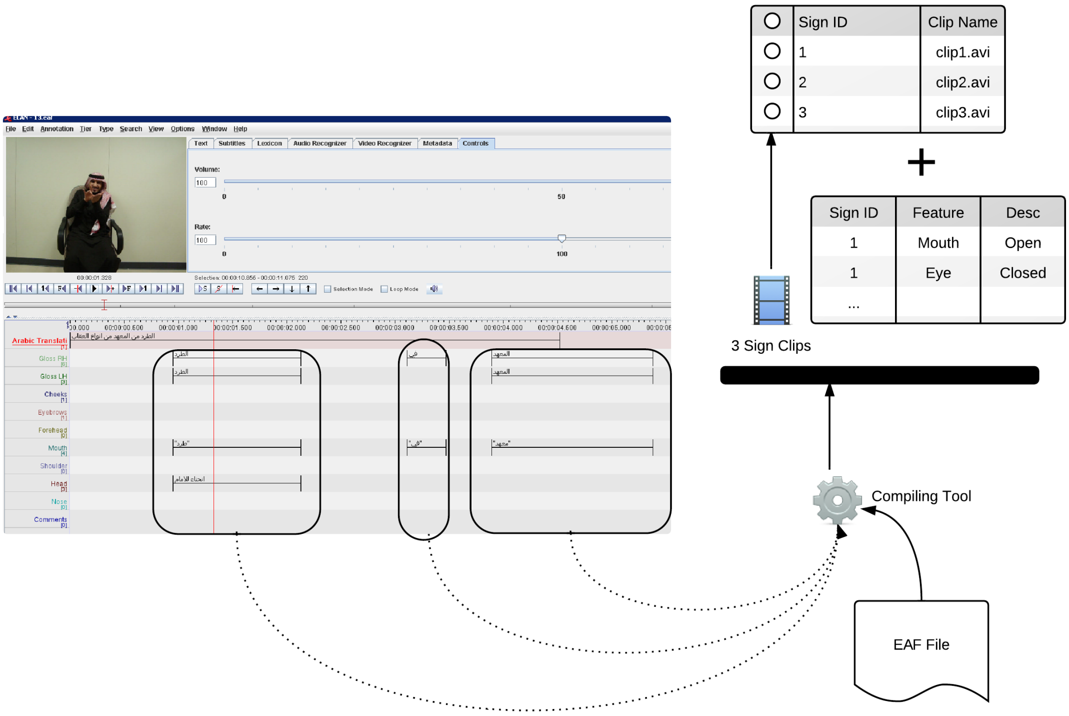


Figure 4.2: Illustration of the information prepared by the compiling tool.

systems that employ a small number of corpora. Table 4.2 shows some text-to-SL translation systems for both SMT and EBMT. As we mentioned in Section 2.3.2, SMT consumes more examples than EBMT (Gough and Way, 2004).

Cost and time limits were taken into account while making decisions. As a result, we constructed our first corpus with 203 sentences, which include 710 signs in the dictionary (Almohimeed et al., 2010). The domain of this corpus

Table 4.1: This table shows the size of corpus that has been used for each of these text-to-text EBMT systems.

System Reference	Language Pair	# of Sentences
Carl and Hansen (1999)	German $\longmapsto$ English	303
Furuse and Iida (1992)	Japanese $\longmapsto$ English	500
Sobashima et al. (1994)	English $\longmapsto$ Japanese	607
Collins (1998)	English $\longmapsto$ German	214

Table 4.2: This table shows the size of corpus that has been used for each of these text-to-SL EBMT and SMT systems.

System Reference	Trans. Approach	Language Pair	# of Sentences
Morrissey (2011)	EBMT, MaTrEx	English $\longmapsto$ ISL	391
Morrissey (2008)	EBMT	English $\longmapsto$ NGT	561
Morrissey and Way (2007)	MaTrEx (SMT+EBMT)	English $\longmapsto$ ISL	577
Bungeroth and Ney (2004)	SMT	German $\longmapsto$ DGS	167

was restricted to the instructional language used in schools for deaf students. Building the first corpus and developing the compiling tool took about 5 months. Later on, we did some experiments using this small corpus, but we decided to construct an average-size corpus to compare with the existing SL corpus. Therefore, we constructed a second corpus that contains 813 sentences and 2,478 signs in its dictionary. This corpus was constructed from a children's story called "Rabbit and Fox". Building this corpus took about 4 months since the team then had experience in building such a corpus; in addition, we were able to use the compiling tool that had been developed for the first corpus.

4.1.5 Spatial Reference Points

In SL, the signer, after introducing a new object, puts it in a particular location in the 3D space at front of him. Then, each time the signer wants to refer to this object, he/she points to its location in the 3D space. For our system, we are unable to follow these reference points, especially since we used concatenated video. Therefore, we reduced these reference points by introducing the object each time it is mentioned. As in the second corpus, we keep using the signs "rabbit" and "fox" again and again each time we mentioned them (see Figure 4.3).

For other spatial references, we used a general spatial reference that can isolate and be used in any sentences. For example, we used a go sign that can be used in any sentence without restricting it by original and destination points.

(a) “rabbit” sign

(b) “fox” sign

Figure 4.3: The “rabbit” and “fox” signs, in which the signer shows these signs without defining reference points.

4.2 Composition of the Team of Signers

Ideally, the signed video in this work would be produced by an individual signer with first-language competence in both Arabic and ArSL. This procedure would ensure that the Arabic text was fully understood and signed with native signer fluency and quality. However, only a very rare individual would have native competence in both languages, if such a person exists at all. Hence, based on the advice of the experts involved, we decided that we needed an interpreter whose first language is Arabic but who has learned ArSL as an additional language. In addition, we needed one or more native signers whose first language is ArSL and who have learned Arabic as an additional language.

The purpose of the interpreter is to tell the signer the content that must be signed. The native signer is better able than the interpreter to produce signs that are understandable to deaf and hearing-impaired people from a range of educational backgrounds. Only a single signer is recorded on the corpus. However, using more than one signer in the team enables us to check the quality and understandability of the signed materials. This is necessary because signers are all individuals; consequently, individuals may interpret or embellish the content according to their individual preferences or tastes. Moreover, they might misunderstand the interpreter.

Using a large team of four people has implications for the cost of collecting the corpus. The size of the team could certainly be reduced; however, we think this could be done only at the danger of compromising quality. The chosen team members for our corpora are as follows:

- Ahmed Alzaharani is an expert in ArSL and has more than 20 years of interpretation experience using ArSL and Saudi local SL. He has a Master's degree in deaf studies from King Saud University, Riyadh. He works for the Ministry of Education in the field of deaf education. He is one of the 17 experts chosen to build the hard-copy ArSL sign dictionary version 1 and version 2. In addition, he is a member of the deaf community club in Riyadh. He also works part-time as an interpreter for Saudi Television.

- Kalwfah Alshehri and Abdulhadi Alharbi have been deaf since birth. They each have a Bachelor's degree and are active in the deaf community club in Riyadh.

- Ali Alholafi has been hard of hearing since birth. He is fluent in Arabic. He has a Tawjihi (high school) degree. He is active in the deaf community club.

In addition to the four members, Mohamed Luhaidan helped us in constructing the first corpus by correcting the grammatical mistakes of the Arabic sentences. In addition, he manually performed the morphological analysis of these sentences. He has a Bachelor's degree in Arabic language and literature.

4.3 Domain

Given the expense and time involved in recording experts signing textual material and in ensuring high-quality signs that can be easily understood, it is clear that any parallel SL corpus must be limited in size. In this work, we have restricted the subject domain of the MT system so that we can achieve reasonable coverage with the small corpus that can be collected in practice. In addition, each word can have more than one meaning depending on the context; therefore, a restricted

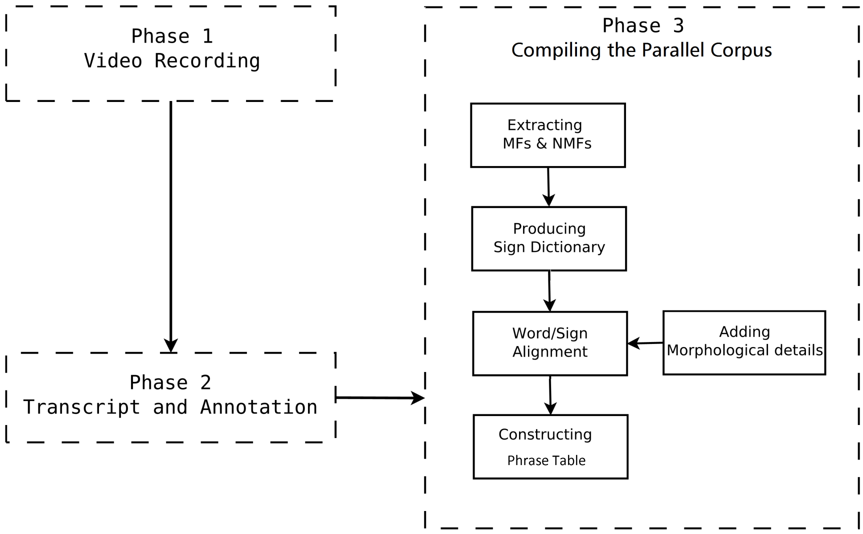


Figure 4.4: Three main phases of building the parallel corpus.

domain will help to reduce this ambiguity. The first corpus domain was restricted to the instructional language that is used in schools for deaf students. It can be described as a one-directional instruction that communicates sentences from teachers to students. This corpus contains more than 710 signs. In this corpus, the signing team prepared a list of 203 Arabic sentences that comprise the most common instructions given by teachers to deaf students. The second corpus was constructed from a children's story written in Arabic called "Rabbit and Fox." The signed version of this story contains more than 2,478 signs.

4.4 Overview of Corpus Construction Process

Figure 4.4 shows the three main phases involved in building the parallel corpus. Recording signed sentences is the first phase. In the second phase, the recorded videos are annotated with the help of ELAN by adding the Gloss Notation to all MFs and NMFs to all signs in the videos. In the third phase, the parallel corpus is constructed and added to the morphological details; the ArSL sign dictionary is also built. This final phase is accomplished using our special-purpose compiling tool.

4.4.1 Video Recording

As detailed in Section 4.2, the interpreter is responsible for reading the Arabic text and explaining the meaning to the native signers. Then, a designated signer produces an equivalent ArSL sentence intended to have the same meaning. The interpreter reviews it; if he accepts it, the two other native signers (not including the designated signer) will read the Arabic text sentence and judge whether or not it is equivalent to the signed version. If not, after discussion to resolve any difference of opinion over the meaning, the designated signer will re-sign the sentence. Later, we record the designated signer's signed sentence; after each recording, the team will again review the recorded video to ensure that it is fluent, complete, and clear for all ages of native signers and understandable by natives with different educational backgrounds. This process is illustrated in Figure 4.5. The final two corpora contained 1016 ArSL sentences in total. Corpus 1 has 203 sentences, and corpus 2 has 813 sentences; both were recorded using a Sony DSC-W120 digital camera and stored in MPEG format. The size of the recorded video frame was 640×480 pixels.

4.4.2 Transcript and Annotation

After the signed sentences were recorded, the videos were annotated using ELAN. This tool possesses all of the functions needed for this task, and it is a well-known tool in the field. A corpus annotated using ELAN can be easily shared with other researchers. Then, Arabic translation was added; each sign in the recorded video was isolated, and extra information on the MFs (hands) and NMFs (e.g., mouth, cheeks, eyes, head, shoulder) was appended. Figure 4.6 shows an example of a sentence annotated in this way. NMFs were described using the Gloss Notation discussed later in this Section. After isolating and adding the MFs and NMFs for all of the signs in the sentences, the annotated data were saved in EAF XML format. See Figure 4.7 for an example.

In the annotation, manual features are well represented by the corresponding Arabic word; however, we require a representation (i.e., gloss) for the non-manual features. Such a notation is needed to store and process the signs; if using a signing avatar, this notation is needed to represent and animate the signs by passing details of the MFs and NMFs to it. Arabic letters, rather than English,

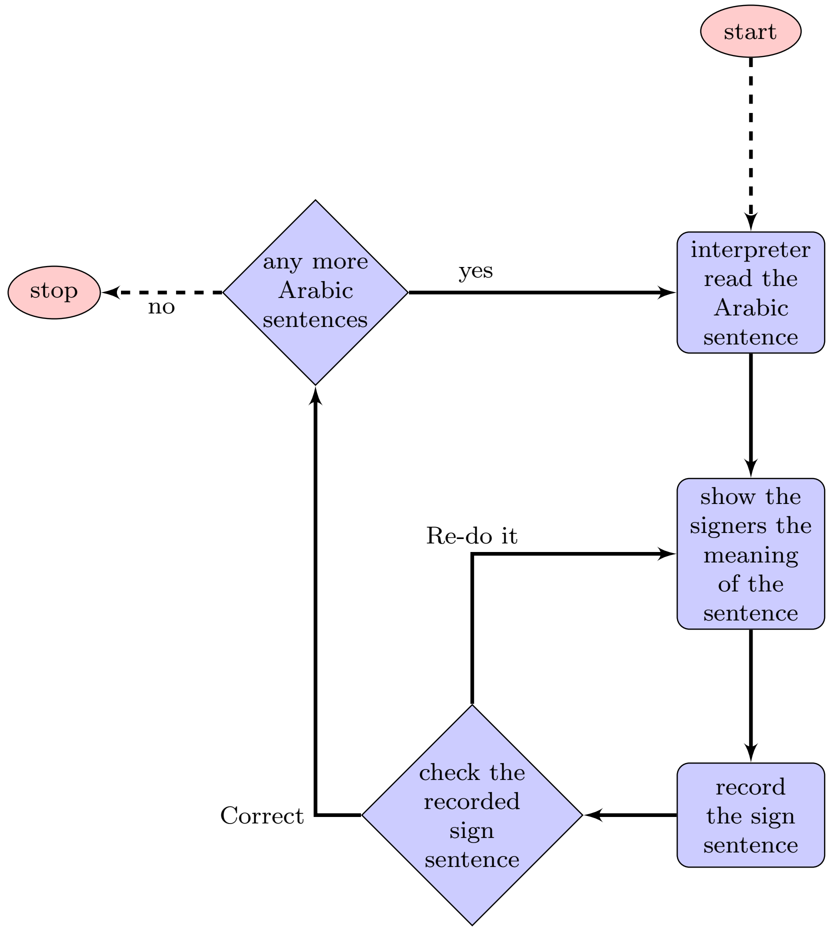


Figure 4.5: A flow chart of the quality control steps for recording the sign videos.

are used for the text part of the parallel corpus during annotation. None of the corpus team members have the ability to write the gloss in English. Therefore, a new specification for writing the Gloss Notation in Arabic has been created.

In general, each NMF feature is represented as follows:

`(NMF Part) -- "Action" -- Action Description`

Example: `(Mouth)` – "جاء" – شد

where "جاء" means the signer's pronouncement, the word "جاء" "*jaa*" and شد stands for stretching the lips.

An example is the textual representation of the sign sentence of the Arabic sentence "السرقة حرام":

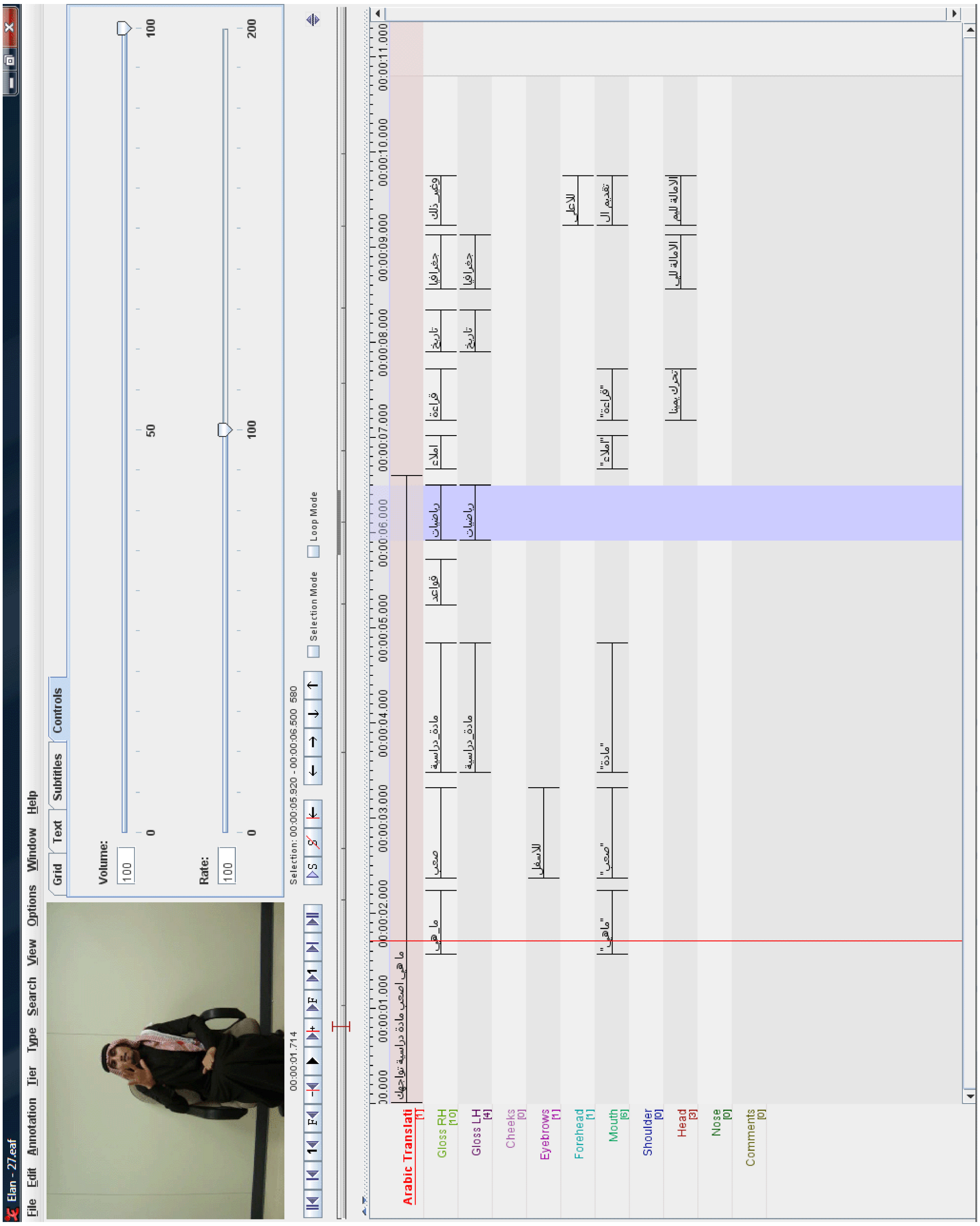


Figure 4.6: An example sentence annotated with ELAN.

```
- <ANNOTATION>
  - <ALIGNABLE_ANNOTATION ANNOTATION_ID="a10" TIME_SLOT_REF1="ts2" TIME_SLOT_REF2="ts4">
      <ANNOTATION_VALUE>ما_هي</ANNOTATION_VALUE>
    </ALIGNABLE_ANNOTATION>
  </ANNOTATION>
- <ANNOTATION>
  - <ALIGNABLE_ANNOTATION ANNOTATION_ID="a11" TIME_SLOT_REF1="ts6" TIME_SLOT_REF2="ts9">
      <ANNOTATION_VALUE>صعب</ANNOTATION_VALUE>
    </ALIGNABLE_ANNOTATION>
  </ANNOTATION>
```


Figure 4.7: An example EAF XML file. This file stores all of the MFs and NMFs, as well as the original Arabic sentence and the sign video location in time.

```
(Mouth) "جاء" شد
(Head)
(Eyes) اغلاق
(Nose)
```

The empty tiers mean no action exists. These annotation tiers can be combined as:

```
(Mouth) "جاء" شد (Eyes) اغلاق
```

where the empty feature will not be taken account. Table 4.3 summarises all of the Gloss Notations used for the corpus.

4.4.3 Compiling the Parallel Corpus

At this stage, we employ Arabic text and annotated sign videos. However, much remains to be done to make the corpus usable for corpus-based translation. This section will discuss all of the stages involved in Phase Three to accomplish this task. No software support existed previously for this work. Therefore, we built our own special-purpose tool to compile the parallel corpus. First, we extract the MFs and NMFs from the XML file. Then we isolate the signs, produce a sign dictionary, and perform word-sign alignment. Finally, we add necessary morphological details to the Arabic sentences. In corpus 1, first, these morphological details were added manually using six main tags (see Section 5.3). Later, we used an Al-Khalil morphological analyser to analyse sentences that produce more detailed morphological information than the manual one. For corpus 2, it was done automatically using only the Al-Khalil analyser.

Table 4.3: Summary of the Arabic Gloss Notation used in the corpus.

NMF	Action	Gloss
Eyes	Closing	"اغلاق العين"
	Opening	"فتح العين"
	Blinking	"ومض"
Nose	Wrinkling	"تجعد"
Mouth	Opening	"فتح الفم"
	Closing	"اغلاق الفم"
	Tongue out	"اخراج اللسان"
	Stre. lips	"شجد الشفاه"
	Sucking air	"شفط الهواء"
	Blowing air	"اخراج الهواء"
Shoulders	Forwards	"تحريك للامام"
	Backwards	"تحريك للخلف"
	Left	"تحريك لليسار"
	Right	"تحريك لليمين"
Cheeks	Puffing out	"ملىء بالهواء"
	Sucking in	"سحب للداخل"
Eyesbrows	Raising	"تقوس للاعلى"
	Lowering	"تقوس للاسفل"

4.4.4 Extracting Features

The first step is parsing the XML files and extracting the MFs and NMFs for each sign. These features are then stored in a feature table together with corresponding necessary information. The feature name field indicates which part of the body is involved in the sign (e.g., right hand, left hand, mouth). The text shows the Gloss Notation of the particular part. The EAF file name and Video fields identify the EAF and Video locations for this part. The start and finish times give the exact location of the feature in the source video, which will be used later in constructing the signs-to-Arabic dictionary to extract the sign video clip from the source video. This process is illustrated in Figure 4.8.

```
- <ALIGNABLE_ANNOTATION ANNOTATION_ID="a20" TIME_SLOT_REF1="ts13"
  TIME_SLOT_REF2="ts16">
    <ANNOTATION_VALUE>مادة_دراسية</ANNOTATION_VALUE>
  </ALIGNABLE_ANNOTATION>
</ANNOTATION>

- <ALIGNABLE_ANNOTATION ANNOTATION_ID="a21" TIME_SLOT_REF1="ts21"
  TIME_SLOT_REF2="ts23">
    <ANNOTATION_VALUE>رياضيات</ANNOTATION_VALUE>
  </ALIGNABLE_ANNOTATION>
</ANNOTATION>
```

Features | MF & NMF Features Table

ID	eafFileName	VideoFileNam	Part	aID	Stim	Ftim	txt
2175	D:\Corpus\EAF\9.	MOV00011.MPG	Arabic Translation	a9	0	7524	لا تضرب من هو اصغر منك
2176	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a10	1140	2320	لا
2177	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a11	2490	3280	تضرب
2178	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a12	4200	6100	اصغر_منك
2179	D:\Corpus\EAF\9.	MOV00011.MPG	Eyebrows	a13	1140	2320	"تقوس للأسفل"
2180	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a14	1140	2320	"لا"
2181	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a15	2490	3280	"تضرب"
2182	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a16	4200	6100	"اصغر منك"
2183	D:\Corpus\EAF\9.	MOV00011.MPG	Shoulder	a17	4390	5210	انحناء

Figure 4.8: Parsing EAF XML and storing the extracted features in the feature table.

4.4.5 Sign Dictionary

The next step is extracting signs to produce a sign dictionary using the extracted MFs and NMFs (See Figure 4.9). All features that occur in the same period of time and have the same video source are grouped to produce a sign. The signs in this sentence are linked to this corpus using the video name as a key. In addition, the order of the signs in the sentence is indicated using a simple integer position indicator.

Sign video clips can now be extracted from the source video files using the start and finish times. After extracting each clip, its location is appended in the sign table as shown in Figure 4.10. At the end of this process, corpus 1 dictionary contained 710 signs; corpus 2 dictionary contained 2,478 signs.

4.4.6 Word-Sign Alignment

The next phase is to align the sign video clips in the MPEG files with the Arabic text delivered from the XML files. This procedure is essential for translation, and

Features — MF & NMF Features Table

ID	eafFileName	VideoFileNam	Part	aID	Stim	Ftim	txt
2175	D:\Corpus\EAF\9.	MOV00011.MPG	Arabic Translation	a9	0	7524	لا تضرب من هو اصغر منك
2176	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a10	1140	2320	لا
2177	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a11	2490	3280	تضرب
2178	D:\Corpus\EAF\9.	MOV00011.MPG	Gloss RH	a12	4200	6100	اصغر_منك
2179	D:\Corpus\EAF\9.	MOV00011.MPG	Eyebrows	a13	1140	2320	"تقوس للأسفل"
2180	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a14	1140	2320	"لا"
2181	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a15	2490	3280	"تضرب"
2182	D:\Corpus\EAF\9.	MOV00011.MPG	Mouth	a16	4200	6100	"اصغر منك"
2183	D:\Corpus\EAF\9.	MOV00011.MPG	Shoulder	a17	4390	5210	انحناء

Signs — SignsDic — Signs Table

ID	Txt	VideoFileName	Gloss RH	Gloss LH	Eye	Eyebrows
5488	لا	MOV00011.MPG	لا			للاسفل
5489	تضرب	MOV00011.MPG	تضرب			للاسفل
5490	اصغر_منك	MOV00011.MPG	اصغر_منك			للاسفل
5491	السرقة	MOV00012.MPG	السرقة		غلاق-العين-اليمنى	
5492	حرام	MOV00012.MPG	حرام		غلاق-العين-اليمنى	
5493	الابتعاد	MOV00013.MPG	الابتعاد	الابتعاد		
5494	ترك	MOV00013.MPG	ترك	ترك		
5495	مشاكل	MOV00013.MPG	مشاكل	مشاكل		
5496	شجار	MOV00013.MPG	شجار	شجار		للاسفل

Figure 4.9: The extracted features are used to build a sign table.

it is complicated by the fact that these two streams of information have different orders; there is no possibility of one-to-one mapping, and the MFs and NMFs overlap in time. After alignment, morphological information about each Arabic sentence is added to the corpus. In addition, we implement a plug-in component in our compiling tool. This component interfaces our compiling tool with the Al-Khalil Arabic morphological analyser. This analyser is, of course, language-specific and would have to be changed if we wanted our system to handle other languages in the future.

Finally, it is important to note that Arabic sentences usually tend to be long

Signs — SignsDic

ID	Txt	eafFileNam	ClipFileName	Gloss RH	Gloss LH	Cheeks	Eye	Eyebrows
5488	لا	D:\Corpus\EAF	c:\clips\1.wmv	لا				للاسفل
5489	تضرب	D:\Corpus\EAF	c:\clips\2.wmv	تضرب				للاسفل
5490	اصغر_منك	D:\Corpus\EAF	c:\clips\3.wmv	اصغر_منك				للاسفل
5491	السرقة	D:\Corpus\EAF	c:\clips\4.wmv	السرقة			غلاق-العين-اليمنى	
5492	حرام	D:\Corpus\EAF	c:\clips\5.wmv	حرام			غلاق-العين-اليمنى	
5493	الابتعاد	D:\Corpus\EAF	c:\clips\6.wmv	الابتعاد	الابتعاد			

Figure 4.10: Sign table after appending the sign clip's location information.

`Sentence=`“آه، ارحمني أيها الثعلب، أتوسل إليك”

(a) Before Separating

`Sentence 1=`“آه” `Sentence 2=`“ارحمني أيها الثعلب” `Sentence 3=`“أتوسل إليك”

(b) After Separating

Figure 4.11: Examples of sentences in corpus 2 before and after separating task.

and combine more than one sentence separated using one or more punctuation marks such as a comma “,”. The first corpus is already separated and includes no punctuation marks except the end full stop mark because the team prepared it. However, the second corpus has many long sentences since it was taken from a story. Examples of the sentences in both corpora are given in Figure 4.11.

After the word/sign alignment, we decided to split the long sentences in corpus 2. Figure 4.12 shows the distribution of sentences in corpus 2, according to the number of signs that they contain, before the separation process. It has 489 sentences and 2,478 signs.

Finally, after splitting corpus 2, we ended up with 813 sentences and 2,478 signs. The distribution of both corpora is shown in Figure 4.13: corpus 1 has 203 sentences and 710 signs and corpus 2 has 813 sentences and 2,478 signs. Figure 4.14 shows the program interface for the sign-to-Arabic dictionary.

4.5 Dealing with Size Limitation

Due to the complexity and cost of building SL corpora, all SL translation systems suffer from the size limitation of their corpus. In our system, we decided to extend our corpora coverage by using morphological details. As described early in Section 2.1.2, the Arabic language is based on root-pattern schemes in which one root can deliver tens or hundreds of words, and this root provides the basic meaning of a word. In other words, the non-existing word in the dictionary can use the sign of a word that shares either the same stem or root. For this purpose, we employed a morphological analyser to extract all the morphological details. Figure 4.15 shows an example of a root and its delivered stem and words.

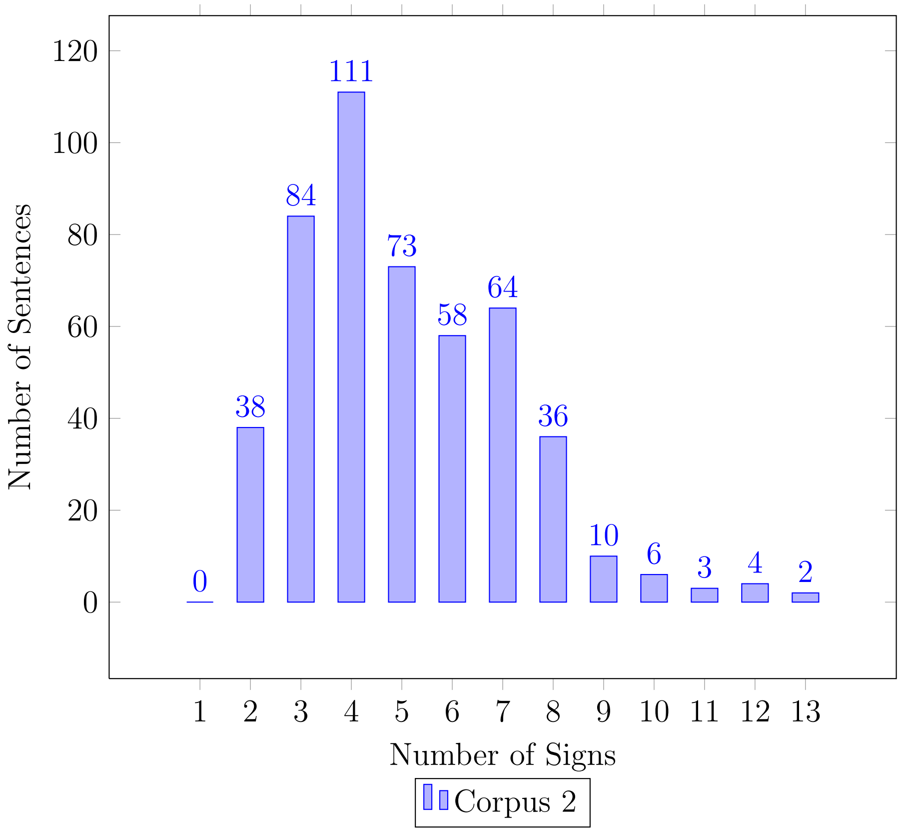


Figure 4.12: Distribution of collected sentences in corpus 2 before the splitting of long sentences according to the number of signs they contain.

In addition to using the morphological details, we did not employ automatic alignment tools such as GIZA++ (Och and Ney, 2000) to make word/sign alignment. We decide to make a manual alignment between word/sign. Figure 4.16 show an example of manual alignment. The reason for this decision is that as Kit et al. (2002, pp. 64) stated, “Manual alignment by experts can, of course, produce quite reliable examples”. Since the sizes of SL corpora in general are limited, it is feasible to make a manual alignment.

4.6 Discussion

Corpus size and coverage has a significant impact on the translation accuracy of any corpus-based MT system. Further, the quality of a parallel corpus is considered the main factor that affects translation quality. We decided to ensure

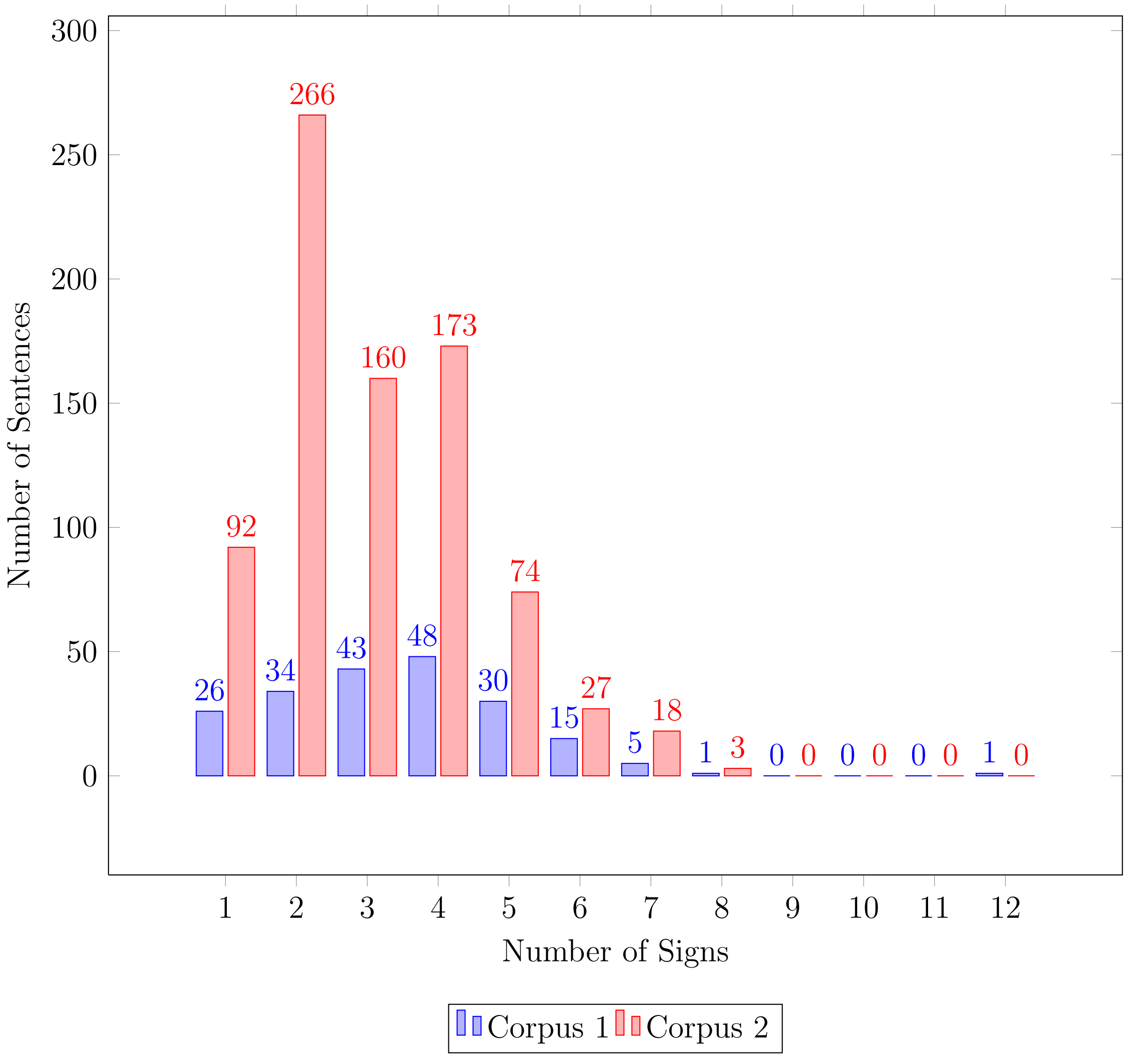


Figure 4.13: Distribution of collected sentences in corpus 1 and 2 after the splitting of long sentences in corpus 2 according to the number of signs they contain.

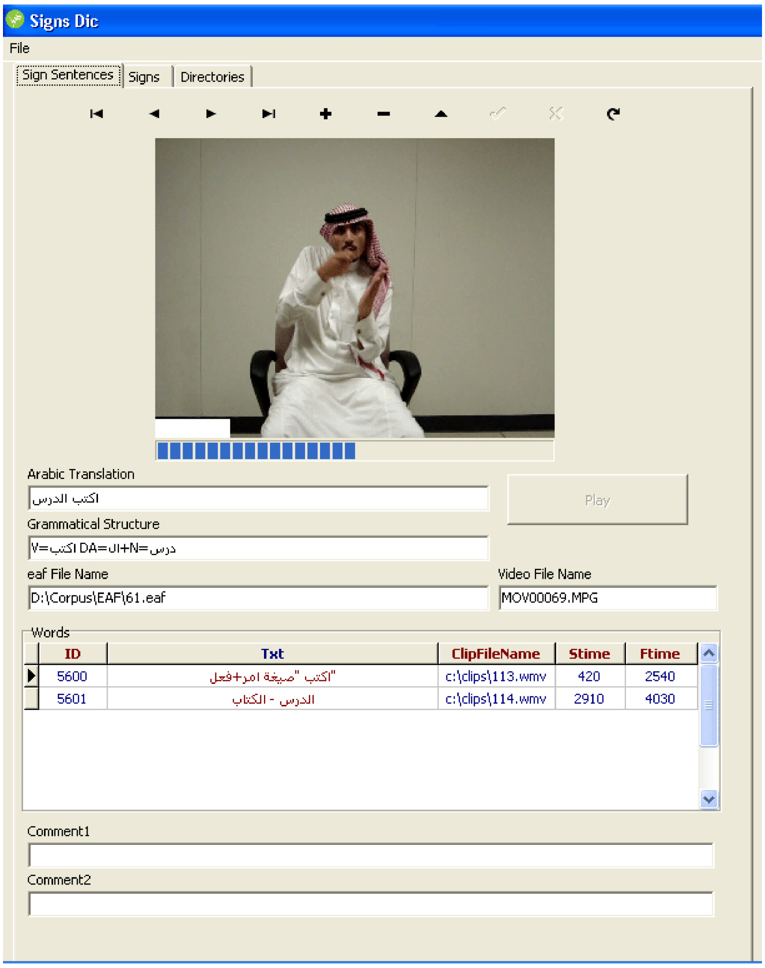


Figure 4.14: The interface of the bilingual corpus of ArSL program. The tagged sentence is shown in the grammatical structure field.

that our corpus is high quality by establishing a corpus team consisting of one professional ArSL interpreter and three native signers with strong educational backgrounds to help manage the quality control tasks while collecting the corpus. In Section 4.2, we justify this number of interpreters and signers.

Meeting the quality standards we set for the corpus is very time consuming and expensive. Corpus 1 required more than 5 months of full-time effort from the expert team to record the sign sentences, annotate them, and develop and use the compiling tool. Building the second corpus and developing the compiling tool took about 4 months although this corpus is larger. The compiling tool was already built, and the signing team had the experience of recording and annotating ArSL sentences. This process also required us to find a suitable

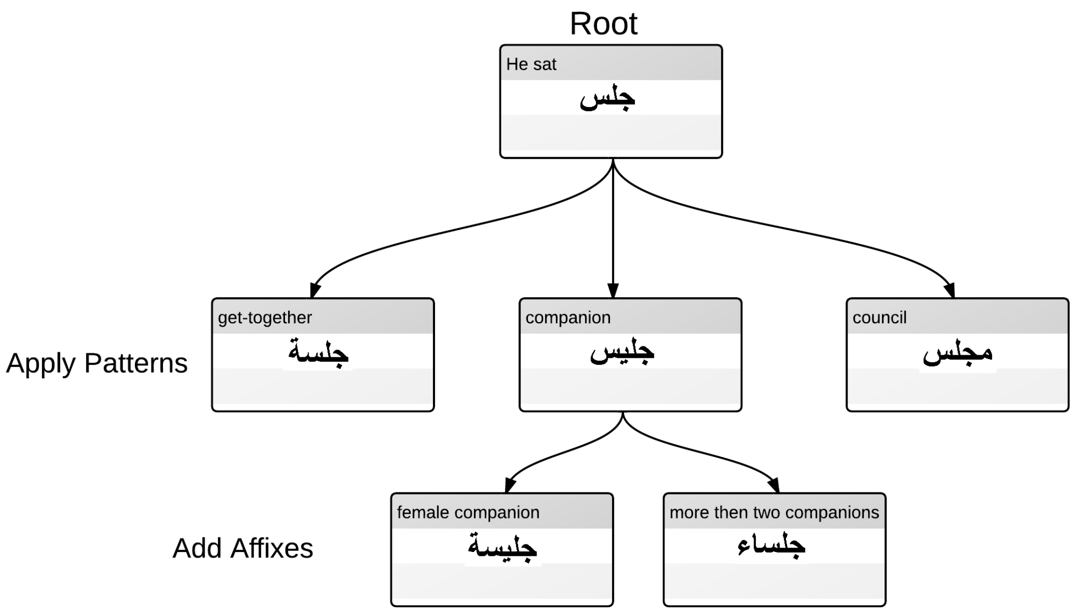


Figure 4.15: An example of root and its delivered stems and words.

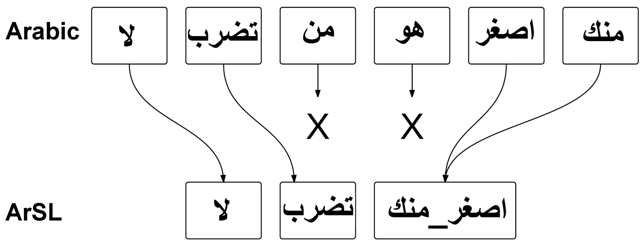


Figure 4.16: An example of Arabic-to-ArSL word alignment. Each block in the Arabic row represents an Arabic word, while each block in the ArSL row represents an ArSL sign. An '×' signifies the fact that the Arabic word does not have an ArSL-equivalent sign.

place for recording and annotating the signs, to purchase equipment (e.g., video cameras), and to travel twice to the Middle East to build each corpus. Researchers who may want to build an SL translation system for another language pair cannot escape the expense and effort of collecting their own parallel corpus. Costs might be saved by reducing the size of team of signers-perhaps just to a single native signer. However, we think reducing the size of the team potentially compromises corpus quality.

The compiling tool that we developed and have described here can in principle be used for any SL translation system. The only language-specific aspect is the plug-in Arabic morphological analyser. An appropriate morphological analyser would have to be provided to suit the specific source language. Finally, it is worth mentioning that our constructed corpora are designed for unidirectional translation. Our approach was created for a system that only translates from Arabic to ArSL, rather than from ArSL to Arabic. This aspect is unlike other parallel corpora that can be used for bidirectional translation. For example, an English-French parallel corpus can be used to translate from English to French and vice versa. Furthermore, the MF features in our corpora might be extracted, depending on the recording conditions of quality requirements for this process, by a sophisticated video-processing algorithm.

Chapter 5

Implementing the ArSL System

As we explained in Section 2.3.2, we favour EBMT over the RBMT and SMT approaches for numerous reasons; for example, it can be easily extended by adding extra sign examples to the corpus. In addition, there is no requirement for linguistic rules. Also, unlike SMT, EBMT can translate using a limited corpus, although performance is expected to improve with a large corpus. The accuracy of the translated output primarily depends on the quality of the examples and their degree of similarity to the input text. Therefore, it is necessary to choose a similarity matching algorithm that can select the most helpful guided suggestion example.

This chapter will start by presenting a new method to reduce word ambiguity. Then, since there are different ways of implementing EBMT systems, we built four EBMT methods and it will be presented in this Chapter, as well as a combination method of two of the presented methods, which improves translation accuracy. The system architecture of each method, including the components, will be explained in detail in this chapter. The discussion of how we evaluate these methods as well as the conducted evaluation results of these methods will be shown later in Chapter 6.

5.1 Arabic Processing Unite (APU)

In Arabic, short vowels usually have diacritical marks added to distinguish between similar words in terms of meaning and pronunciation. For example, the word كُتُب means *books*, whereas كَتَب means *write*. Most Arabic documents are

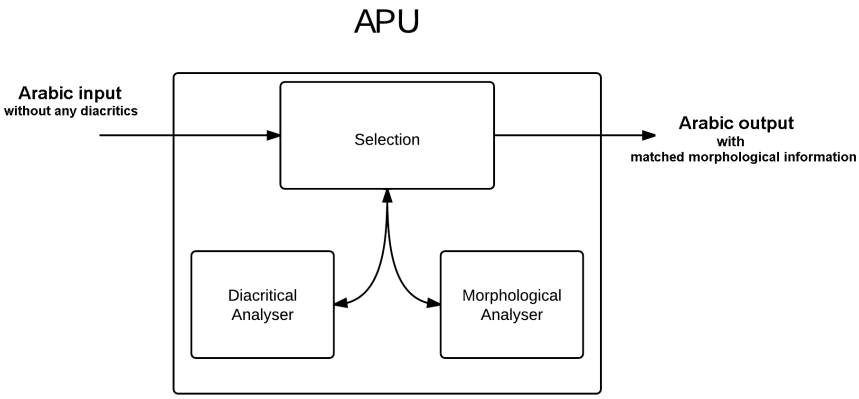


Figure 5.1: The main components of the Arabic Processing Unit.

written without the use of diacriticals because Arabic speakers can naturally infer them from the context. On the other hand, systematically determining the correct diacriticals is very difficult. Generally, Arabic morphological analysers used in Arabic language processing accept Arabic input without diacriticals, but they produce many different analysed outputs by making different assumptions about the missing diacriticals. In the end, the system selects one of these analysed outputs, but it might not be the same as the input meaning.

To increase the chance of forcing the SLMT system to select the correctly analysed output that matches the meaning of the input, we built an Arabic processing unit (APU) component. It is responsible for detecting the equivalent input meaning of words in a sentence. The component employs a morphological analyser, such as an Al-Khalil morphological analyser (i.e., Section 3.1 provides details about additional analysers), and an Arabic diacritical analyser (e.g., Google Tashkel) which adds the missing diacritical marks to the input sentences according to a diacritical corpus-based technique (Elshafei et al., 2006; Nelken and Shieber, 2005); see Figure 5.1.

Arabic input without any diacritics proceeds to the APU components. The APU then forwards the input to the morphological analyser, where the analyser scrutinizes each word in the sentence and returns a list of this word with different diacritics; see Figure 5.2. Each word might have a different meaning, stem, and root. Therefore, in order to determine which of these analysed words that contain diacritics is the one that matches the meaning of the word in the sentence, the

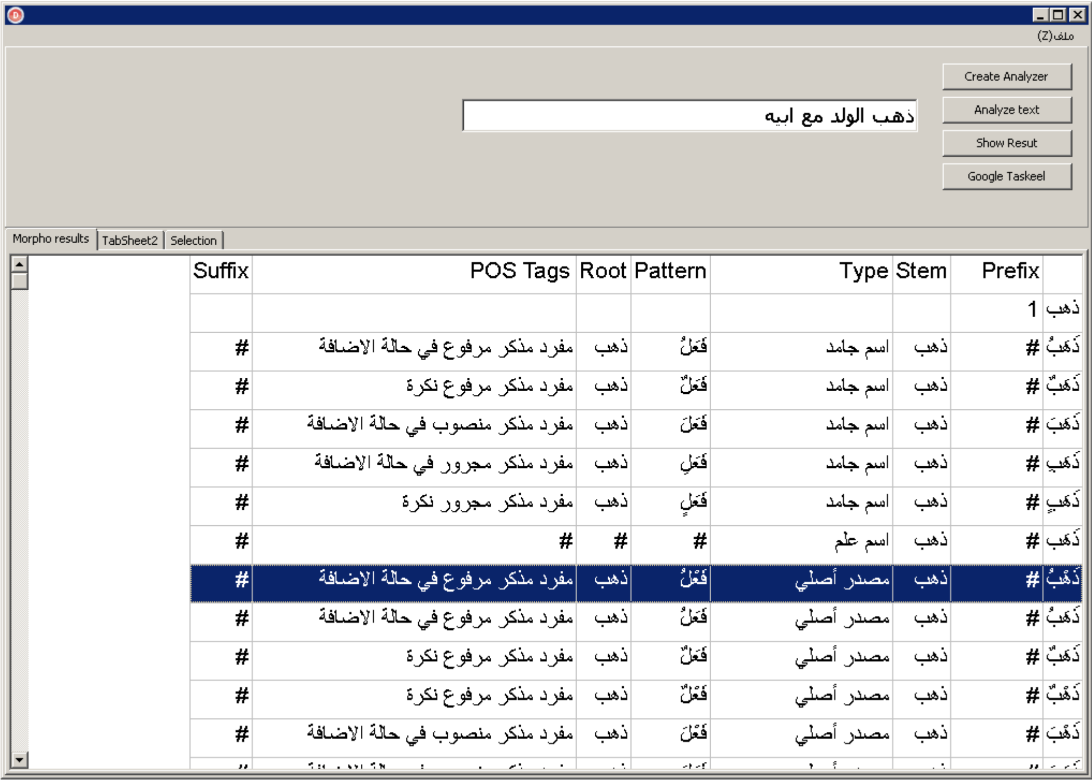


Figure 5.2: An example of a morphological list produced by the Al-Khalil morphological analyser that scrutinizes the word “ذهب”. The first column on the right is the list of all produced words with different diacritics. The word that is highlighted in blue means “go,” while the line above it means “gold.” Both of these lines contain words with different diacritics. Therefore, the morphological analyser has no ability to detect which diacritic-containing word matches the input word.

diacritical analyser is used to add the missing diacritic to the input sentence, as shown in Figure 5.3.

To get the correct meaning and morphological details of the input words, the diacritical sentence produced by the diacritical analyser must be linked to the correct corresponding and matching words in the analysed words from the morphological analyser's analysed list. However, the diacritical analyser produced marks for all letters in the input sentence, while the morphological analyser only added the essential marks that affected the meaning of the words. Therefore, it is not a straightforward process to link the two analysed sentences, so, to link the

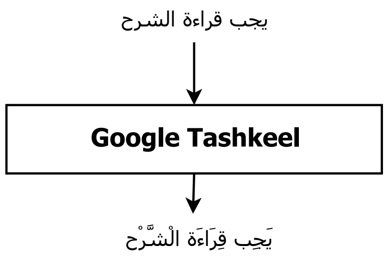


Figure 5.3: An example of an input and output text using *Google Tashkeel*. The input is a sentence without diacriticals; the output shows the same sentence after adding diacriticals.

two equivalent words, we used the Levenshtein distance algorithm (Levenshtein, 1966), also called edit distance, and the length of words to select the equivalent words. Figure 5.4 shows an example of selection using this technique.

The system starts by comparing the edit distance between the diacritical word against the words generated by the morphological analyser. Then, it selects the words that have less distance. After that, from these words, it selects the shortest word.

Our implemented APU component used an Al-Khalil morphological analyser. The original Al-Khalil analyser cannot be integrated with any NLP software. It only allows the end user to manually enter the text and then to click the 'Analyse' button. Therefore, since it is a Java open-source analyser, we rewrote it in Delphi XE and plugged it into our translation system. Furthermore, we developed another stand-alone version that allows other NLP software to benefit from this analyser. We kept the manual feature that allows the user to enter text directly into the program. We also added a new integration features that allow NLP software either to use an ActiveX plugged-in component or to integrate it by running a batch file. Figure 5.5 shows the original Java main screen and our re-written version screen.

5.1.1 Evaluation

An evaluation was conducted to show how well the automatic selections were correlated with selections made by human linguistic experts. The selections were chosen by one linguistic expert based on 1,038 sentences and a total of

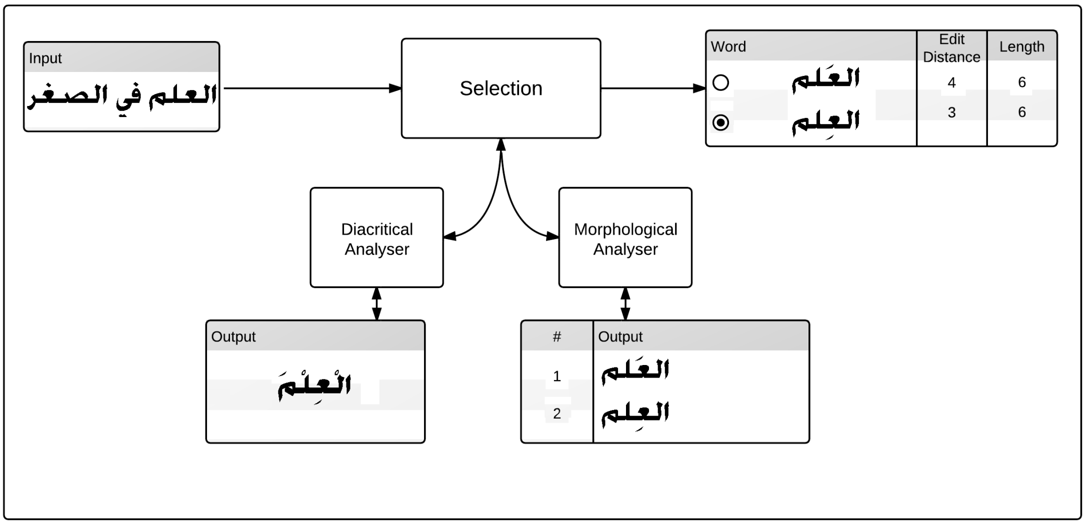


Figure 5.4: An example of linking equivalent analysed diacritical words with their equivalent morphological words.

4,740 words. The total number of wrong selections was only 66 words (1.4%), while the total number of correct selections was 4,010 words; this means that 84.6% of the selections were correct. The morphological analyser was not able to analyse 448 words (9.5%). In addition, the system failed to show a correct analysis in the analysed outputs of 216 words (4.56%). Figure 5.6 shows the main screen of the human evaluation selection software.

Due to Google's decision to shut down some of its API services such as *Google Tashkeel*, we switched to *Mishkaal*, an open source Arabic text vocalization project (`http://sourceforge.net/projects/mishkal/`). We ran automatic selections using *Mishkaal* based on the same 1,038 sentences that had been used previously to show the correlation between *Google Tashkeel* and human selection. The results show that the total number of wrong selections was 1,356 words (28.6%), while the total number of correct selections was 2,936 words (61.9%). This means that *Google Tashkeel* shows a better correlation with human linguistic experts than *Mishkaal*.

5.1.2 Summary

A new technique to reduce the ambiguity of Arabic words was proposed. This technique employs the power of an existing morphological analyser and an Arabic

(a) The main screen of the original Al-Khalil analyser

(b) The main screen of our improved version

Figure 5.5: In (a) the original screen of the Al-Khalil analyser in which (b) our Delphi version enables the feature of integration with it by other NLP software.

Figure 5.6: Main form of the selection software used by the human linguistic expert to choose the correct analysed word.

diacritical analyser. Then, it links the equivalent matched words between the analysed words generated by both analysers by using edit distance and word length. The evaluation shows that this new technique results in reduced word ambiguity, giving the machine a method of selecting the right morphological analysed words that match the input meaning. The results also show that the accuracy of both the diacritical and morphological analysers has a high impact on the accuracy of the selection.

5.2 Considerations

We adopt an EBMT approach for the translation. There are many ways in which to implement EBMT system (Carl and Way, 2003). Therefore, this chapter presents different versions of Arabic text-to-ArSL EBMT, which have been developed in order to yield a final, accurate translation-implemented method. Most of these versions utilise the previous APU component. Also, concatenated video-based sign synthesis, which will be discussed at the end of this chapter, will be used to produce the translation output.

To create our ArSL system, we decided to use four different methods. Efforts were made to combine some of them in order to improve translation accuracy. This chapter will explain each method in detail, but here we summarise the four methods and the reasons why they were selected.

- After traveling to Riyadh and putting together the signing team that helped us record the sign videos and make the annotations, we developed a website to allow the signing team to evaluate our translated examples remotely without having to get in touch with us in person. Therefore, our first method was built and tested for two main reasons: first, to test the first corpus constructed by us; second, to test the feasibility of having an evaluation website and to test the time required by the signing team to finish their evaluations. We developed the first method after studying the corpus we had collected. It has three matching phases that first employ the structure tags that were manually added to the corpus for selecting examples that have a similar structure. Then, based upon these examples, the method selects examples that share a high number of similar words, and, finally, it selects an example that has a similar length. Section 5.3 describes all these details.

- The second method we built is the classical method in which the system simply seeks out the closest matching sentence on the basis of the word edit distance (Levenshtein, 1966). Implementing this classical method is important since it shows the improvement of accuracy of other methods compared to this original method. Section 5.4 will describe all the implementation details of this method.

- The third method is the similarity method. Unlike the first method, this method employs all the morphological details obtained from the morphological analyzer to search for similar examples. Section 5.5 describes this method.

- The fourth method is a phrase-based method, which employs the refined method of Och and Ney (2003) to extract phrases and then use these phrases in the translation. This method is described in Section 5.6.

```
<PRON> you click apply   <PRON> vous cliquez sur appliquer
<PREP> to view           <PREP> pour visualiser
<DET> the effect         <DET> leffet
<PREP> of the selection  <PREP> de la selection
```

Figure 5.7: An example of marker-based fragmentation.

- Finally, on the basis of the evaluation results, we combined two methods, phrase-based and similarity, to construct a system that can deliver translations that are more accurate. Section 5.7 explains how this system works.

Before we finish presenting the methods we employed in building our ArSL translation system, It is important to note that we need to address the differences between the EBMT methods we have selected and the EBMT method selected by Morrissey (2008), because her system is the only other SL EBMT system that exists. As we mentioned in Section 3.2.2, her system is based on Way and Gough (2005)'s method, which uses a small set of POS tags for sub-sentence segmentation to represent the grammatical structure based on the marker hypothesis principle proposed by Green (1979), who said 'The Marker Hypothesis states that all natural languages have a closed set of specific words or morphemes which appears in a limited set of grammatical contexts and which signal that context'. Way and Gough (2005)'s method uses eight 'chunker' markers: Determiner, Quantifier, Prepositions, Conjunctions, WH-Adverbs, Possessive Pronouns, Personal Pronouns, and Punctuation Marks. Both the source and target texts should have tags in order to be able to extract POS fragments. In addition, each segment must include at least one non-marker word. Figure 5.7 shows an example of POS fragments. We cannot employ this method for ArSL translations because no ArSL POS tagger exists.

5.3 Three-Phase EBMT

After we constructed our first corpus, we developed an evaluation website that allows the members of the signing team to evaluate our translated examples from distance aiming to minimise the complexity and time consumption of communicating in person or via email or phone through an interpreter. So, after completing the corpus with the necessary information, such as adding the main structure tags and constructing the final sign dictionary and its associated sign clips, we decided to build our first method for two main reasons:

a. To make sure that our first corpus is ready for MT tasks and can delivered a concatenated-video signs translated output as planned.

b. To test the feasibility of an evaluation website, in terms of the time required for the team members to finish their evaluation or whether the team is familiar with technology and can easily access the site and evaluate without guidance or further help. This will be discussed later in Chapter 6.

5.3.1 System Design

In the first method, we decided to start by trying to build a specially-design method for ArSL translation. Therefore, after consulting the signing team, we came up with a new method that takes into account the following factors:

1. As seen in Section 2.2.3, ArSL has a flexible structure and it can be expressed in different orders such as SVO, VSO. Therefore, if we found a similar example that shared most of the words and structure tags, we may follow its sentence order, taking into account the similarity of length during the selection, and deliver a translation output.

2. After studying the conducted first corpus and consulting the team, we found that ArSL does not have a definite article, unlike other SLs such as ISL (Morrissey, 2008). It is worth mentioning that after we built our first method, the first ArSL linguistic reference was published and it clearly states that ArSL does not have definite articles (Al-Binali and Samareen, 2009, pp. 116), see Figure 5.8.

3. Intensively processing the corpus is possible for SLMT because all existing SL corpora are considered small. Therefore, this method selection process consists of three phases, hence, we called the method 3-phasess EBMT.

The system employs the manual structure tags that were already added to the first corpus to determine the structure of each example. These tags are the main structure tag set of Arabic that was proposed by Khoja et al. (2001), as shown in Table 5.1.

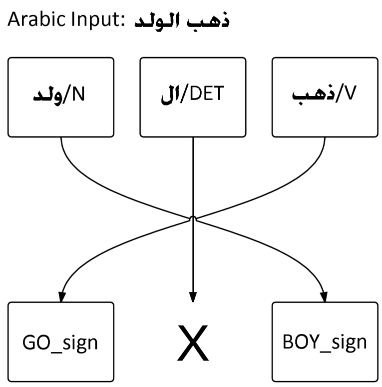


Figure 5.8: An example of words/signs alignment between an Arabic and ArSL equivalent sentence. An '×' signifies the fact that the Arabic word does not have an ArSL-equivalent sign.

Table 5.1: The five main tags that were manually added to the first corpus. In addition, we added an extra tag that is the definite article. In addition to the five main tags proposed by Khoja et al. (2001), we added a 'D' tag, which stands for definite article. Since a definite article does not exist in ArSL, we decided not to consider it during the selection process.

Abbreviation	Tag	Description
N	Noun	
V	Verb	
P	Particle	It is used as a prefix and connects words, phrases and clauses together.
R	Residual	It contains foreign words, numbers, and mathematical formulas.
PU	Punctuation	It contains all punctuation symbols both Arabic and foreign languages such as ? ! "
D	Definite Article	

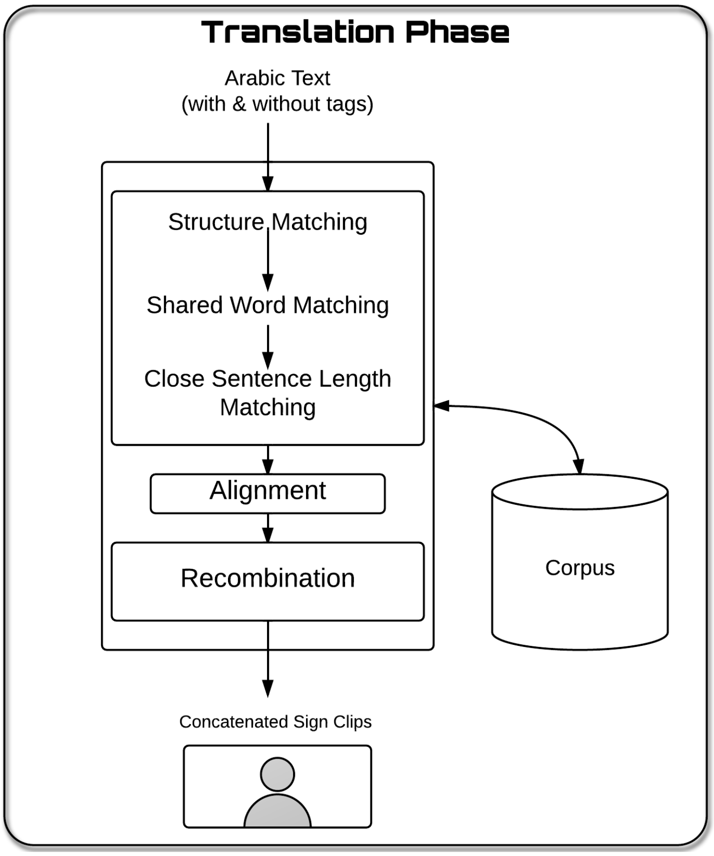


Figure 5.9: An overview of the main components of 3-phasess EBMT.

The architecture of the system and the main components of this method are described in Figure 5.9. As shown in the Figure 5.9, the translation process is done in run-time since the system is not required to do pre-processing tasks in order to start translation. In this method, the user first enters the input twice, once without adding any tags and then by manually adding tags using the previous six tags. Subsequently, in the matching component, the selection of a similar example occurs after first selecting the example with a similar structure, and then it selects the examples that share a high number of similar words. Finally, it selects the example that is most similar to the example length. The output will be produced using a concatenated video technique that was presented in Section 5.7.1.

Input without tags: خرج الرجل
Input with tags: خرج/V ال/D رجل/N

Figure 5.10: An example of two inputs of the same sentence. The first input is without additional tags. The second was tagged manually.

5.3.2 Matching

This component received Arabic input twice; once without adding a tag to the original example and once after the tags were added manually with the six main tags described before; an example is shown in Figure 5.10.

Then it's go throw a three phases as following:

1. It starts searching the corpus for the most structurally similar sentence. It applies the following formula and then selects the sentence with the highest score.

Algorithm 5.3.1: GETSCORE(R_{PoS}, C_{PoS})

$score \leftarrow 0$
if $|R_{PoS}| > |C_{PoS}|$
  then $n \leftarrow |C_{PoS}|$
  else $n \leftarrow |R_{PoS}|$
for $i \leftarrow 1$ **to** n
  do if $C_{PoS}[i] = R_{PoS}[i]$
  then $\begin{cases} \textbf{if } C_{PoS}[i] = \text{`N' or } C_{PoS}[i] = \text{`V'} \\ \quad \textbf{then } score \leftarrow score + 3 \\ \quad \textbf{else if } C_{PoS}[i] \neq \text{`D'} \\ \quad \textbf{then } score \leftarrow score + 1 \end{cases}$
return ($score$)

where all parts of speech involve in the reference and candidate sentences are in R_{PoS} and C_{PoS}. Also, $|R_{PoS}|$ and $|C_{PoS}|$ are the total length of each one of them. The 'D' tag is weighted at 0 because a definite article does not exist in ArSL. On the other hand, noun and verb tags will score 3 since they affect the meaning more than other structure tags. Increasing the score more than 3 will not affect matching results. Examples that have the highest scores will be selected for the next phase.

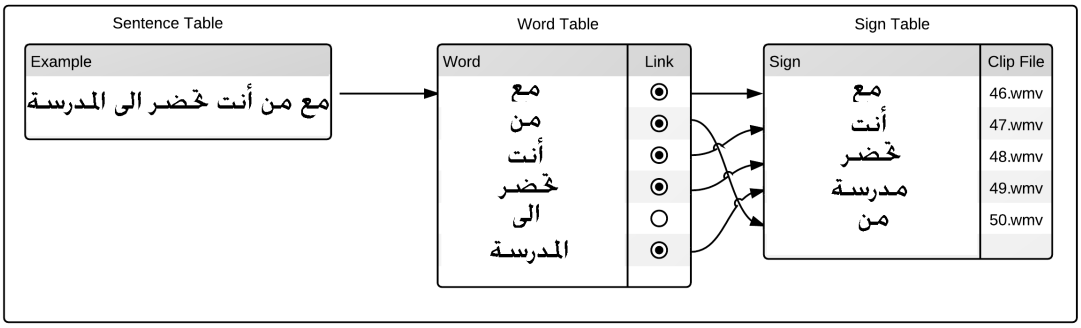


Figure 5.11: An example of a retrieved match from the corpus and its associated ArSL translation and sign clips.

2. In this phase, the system will calculate the number of shared words between the selected examples from phase 1 and the input sentence. Then, it selects examples that have the highest number of shared words for phase 3.

3. Phase 3 searches for the most similar example in terms of length of input. Therefore, it applies the following formula in order to calculate the score:

$$S = |input| - |example| \tag{5.1}$$

   Where S is the score and $|input|$ is the number of words in the input sentence. While $|example|$ is the number of words in the example sentence. Then, it sorts the examples based on their score. Finally, for selecting the most similar example, it select one positive score starting from zero; if no positive score exists, it selects one negative score starting from -1.

5.3.3 Alignment

After selecting the match example, the system retrieves it with its ArSL translation and all of its associated sign clips (see Figure 5.11). In addition, since the corpus has already manually alignment the word/sign, it necessary to use the input sentence without tags in order to perform the alignment task. During the alignment task, the system identifies the corresponding match words between the input and example, as shown in Figure 5.12.

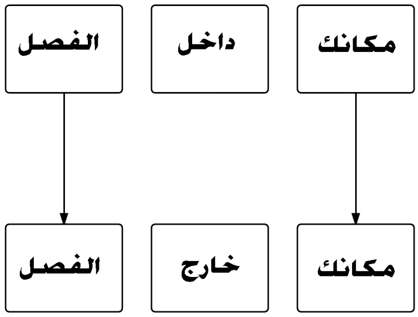


Figure 5.12: An example of identifying the corresponding match words between the input sentence and the match example.

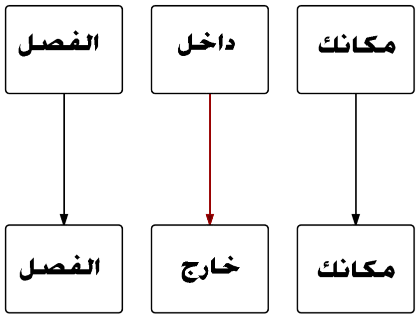


Figure 5.13: An example of linking between the input words and match sentence words. The black arrow means these link words are matched, while the red arrow means that we link between the two mismatch words for the reordering issue.

To address the reordering issue in ArSL, the mismatch words in the input search for the first available mismatch example word and link with it (see Figure 5.13).

5.3.4 Recombination

Once the corresponding match words in the example have been identified, the corresponding signs are used to replace the original word. In addition, the location of the corresponding signs of the linked, mismatched words will be used. Thus, the mismatched word in the input will search for an equivalent sign in the dictionary and then use the location that it already occupied. When the input for other mismatched words fails to link with any words in the example, their equivalent sign in the dictionary will be added to the end of the ArSL sentence that is produced.

5.4 Classical EBMT

This method uses the original examples paired in the corpus without employing their syntactic or structure details. Using this method, the system searches for the closest matching example on the basis of the word's edit distance (Levenshtein, 1966). We use this method as a baseline to shows the accuracy improvement of other methods. The following dynamic programming algorithm is used to calculate the similarity score of examples in the corpus:

Algorithm 5.4.1: LEVDIST(s, t)

$$
\begin{array}{l}
len_s \leftarrow |s| \\
len_t \leftarrow |t| \\
cost \leftarrow 0 \\
\textbf{if } s[0] \neq t[0] \\
\quad \textbf{then } cost = 1 \\
\textbf{if } len_s = 0 \\
\quad \textbf{then } \left\{ \begin{array}{l}
\textbf{return } (len_t) \\
\textbf{else if } \left\{ \begin{array}{l}
\textbf{if } (len_t = 0) \\
\quad \textbf{then return } (len_s) \\
\quad \textbf{else } \left\{ \begin{array}{l}
x \leftarrow minimum(LevDist(s[1..len_s - 1], t) \\
+1, LevDist(s, t[1..len_t - 1]) + 1, \\
LevDist(s[1..len_s - 1], t[1..len_t - 1]) + cost) \\
\textbf{return } (x)
\end{array} \right.
\end{array} \right.
\end{array} \right.
\end{array}
$$

Algorithm 5.4.2: MINIMUM(a, b, c)

$$
\begin{array}{l}
mi \leftarrow a \\
\textbf{if } (b < mi) \\
\quad \textbf{then } mi \leftarrow b \\
\textbf{if } (c < mi) \\
\quad \textbf{then } mi \leftarrow c \\
\textbf{return } (mi)
\end{array}
$$

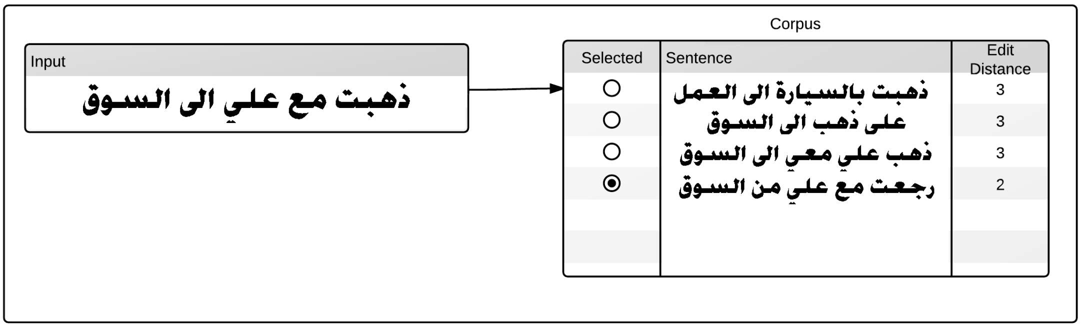


Figure 5.14: Matching example of classical EBMT.

The system selects the example that has the higher score as a matching sentence (see Figure 5.14). Then, the system performs the alignment and recombination as the previous 3-phasess method.

This method does not have a pre-processing phase. It is only relies on the original form of sentence pairs in the corpus.

5.5 Similarity-Based EBMT

In the 3-phases based EBMT, the system strictly employs a six structure tags to search for similar sentences. In this method, the system uses the power of a morphological analyser, which produces a high range of structure categories and sub-categories. In addition, since the corpus size is limited, the system avoids a vocabulary problem by using the stem and root of the word to locate a match word.

5.5.1 System Design

As shown in Figure 5.15, in the pre-processing phase, the system analyses the morphological details of all examples in the corpus by passing each example to the APU component. Once this information is acquired, there is no need to redo this phase again unless a new corpus or examples are used. Then, in the run-time phase, each time an input needs to be translated into ArSL, it is first analysed by the APU and then it passes to the matching component where it starts searching for the closest match example in the corpus based on the following criteria:

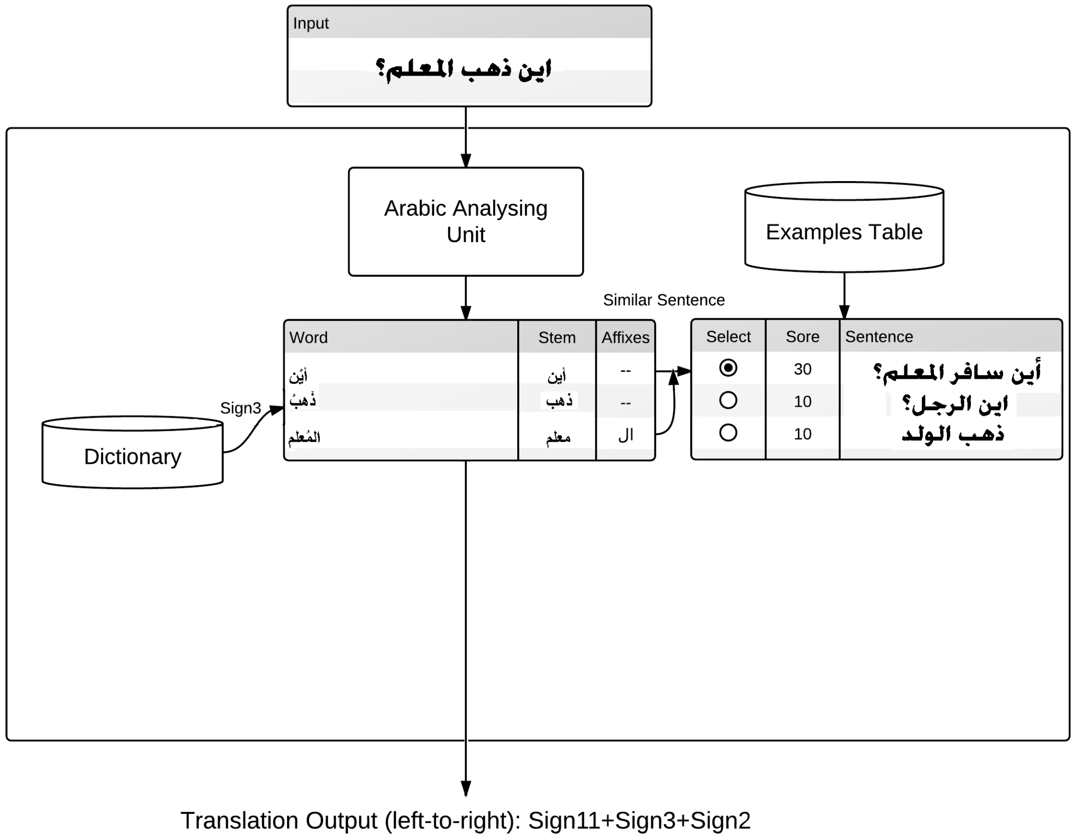


Figure 5.15: Similarity-based EBMT method architecture.

a. The number of matched morphological main categories;

b. The number of matched morphological sub-categories; and

c. The number of exact root matches.

Each criterion has its own weight, so the system selects the example that has the highest calculated score based on the previous criteria.

5.5.2 Matching

The similarity-based method searches for a sentence that is similar to the input. Therefore, the similarity formula proposed by Andriamanankasina et al. (1999) is extended and employed to find the most similar sentence to the input. The original formula only took into account the number of root and POS tag matches;

these matches were restricted and thus suitable for general translation. The original formula is

$$\text{Sentence Score} = \alpha \times \text{NR} + \text{NP} \tag{5.2}$$

where NR is the number of exact matches, and NP is the number of matches to the POS tag set as used by Andriamanankasina et al. (1999). They determine the value of α, which is 10. However, we are able to considerably extend the set of POS tags to cater specifically for Arabic. The following similarity score is then used to select the best-matching sentences:

$$\text{Sentence Score} = \alpha \times \text{NR} + \text{NP} + \frac{\text{ND}}{10} \tag{5.3}$$

where ND is the number of matches to the more detailed POS tag set defined for Arabic. A weight of 100 was used to score the presence of exact root matches as more significant compared to NP and ND, with a weight of 10, making NP matches more significant compared to ND matches. The Al-khalil morphological analyser produces 48 NP categories and 539 ND categories. It is important to note that it is not possible to justify α since we don't have enough sentences to determine the right value for Arabic and do cross-validation to verify it. Therefore, we rely on the Japanese value and we add the sub-categories ND to the formula and divide by 10.

5.5.3 Alignment and Recombination

Alignment and recombination in the similarity-based method is similar to the 3-phases and classical methods. However, when recombining the system after identifying the corresponding match word, it will search for a match tag between the mismatch word between the input and the example in order to use the example word location for the input word during the recombination task.

5.6 Phrase-Based EBMT

All the previous methods simply retain the corpus in its original form with some additional morphological details and compare the input against the corpus sentences, looking for partial matches of any size. In contrast, the phrase-based method starts by pre-compiling the corpus into a reduced set of phrases. All words/signs in a phrase are neighbours on both sides; Arabic and ArSL sides. We called this method as a phrase-based EBMT because we used Refined Method (Och and Ney, 2000) to generate these phrases that are also used in phrase-based SMT.

5.6.1 System Design

The system has two phases. Phase 1 is run only once; it pre-compiles and generates the phrases and their associated signs by using the Refined Method to parse the corpus sentences. Phase 2 is the actual translation system that converts Arabic text into ArSL output. Each component of Figure 5.16 is described below.

5.6.2 Matching

The first component is responsible for finding phrases that match the input. It starts matching words from the beginning of the phrase table and scans the table through to the end. Overlapping phrases have a higher priority for selection than separate phrases. Then, for any remaining unmatched input words, it starts matching stems from the beginning through to the end of the phrase table.

5.6.3 Alignment and Recombination

The second component is the alignment component, which replaces phrases with their equivalent signs. A sign dictionary is used to translate the remaining input words that do not have a phrase match. If the word does not appear in the dictionary, which is possible given the small size of the corpus, the system starts searching for the stem of the word and compares it with the stems in the dictionary. If the stem does not appear in the dictionary, the system searches for a matching root. This process will increase the chance of translating the entire input sentence. However, it is always possible that no match will be found, in

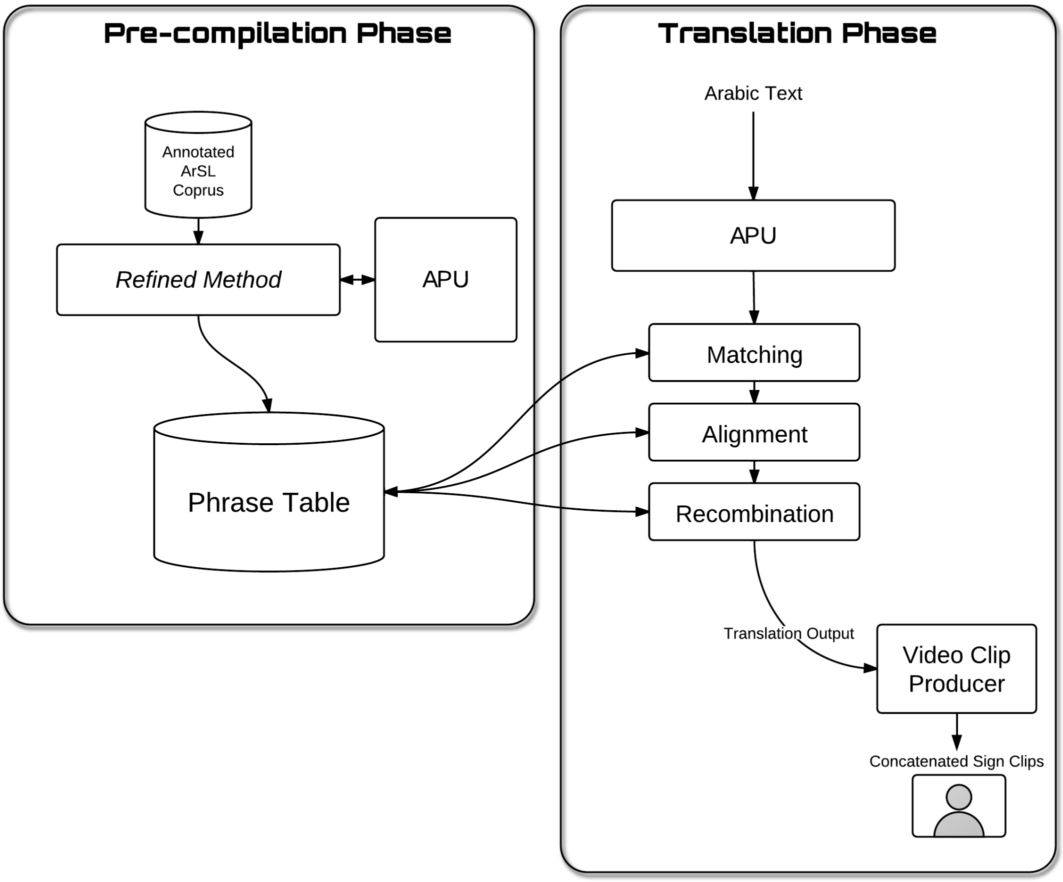


Figure 5.16: The main components of the phrase-based EBMT. Phrase 1 is the pre-compilation phase, and Phase 2 is the translation phase.

which case, the system cannot produce output. Figure 5.17 shows an example of this. The last component is recombination, which is responsible for delivering the sign output using the sign location on both the phrase table and the dictionary. This component will produce a series of sign clips, which are then concatenated as in Section 5.7.1.

5.7 Combining Similarity- and Phrase-Based EBMT

We adopted two EBMT approaches: a phrase-based method and a similarity-based method. Figure 5.18 shows an outline of how both approaches are implemented. The system has two phases. Phase 1 is run only once; it pre-compiles

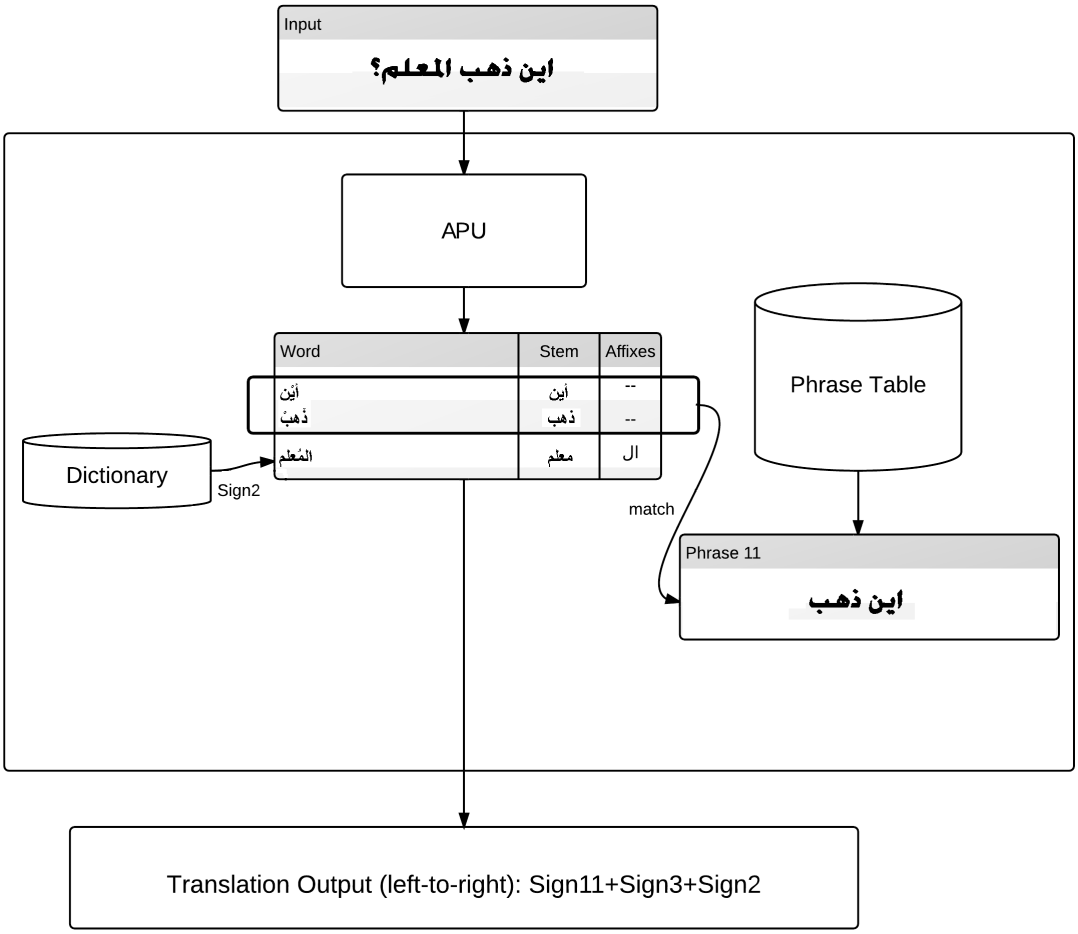


Figure 5.17: Example of matching in the phrase table.

the phrases and their associated signs and parses the corpus sentences. Phase 2 is the actual translation system that converts Arabic text into ArSL output.

5.7.1 Concatenated Video-based Sign Synthesis

Since ArSL is a visual language and does not have a standard writing system, it was necessary for us to find a way to present the translated ArSL output. Using a signing avatar is one way to present these translated signs. However, when we had to make this decision, avatars were showing promising results but not adequate performance to show all MFs' and NMFs' details. Also, the avatar integration cost was another issue that we took into account because it involves

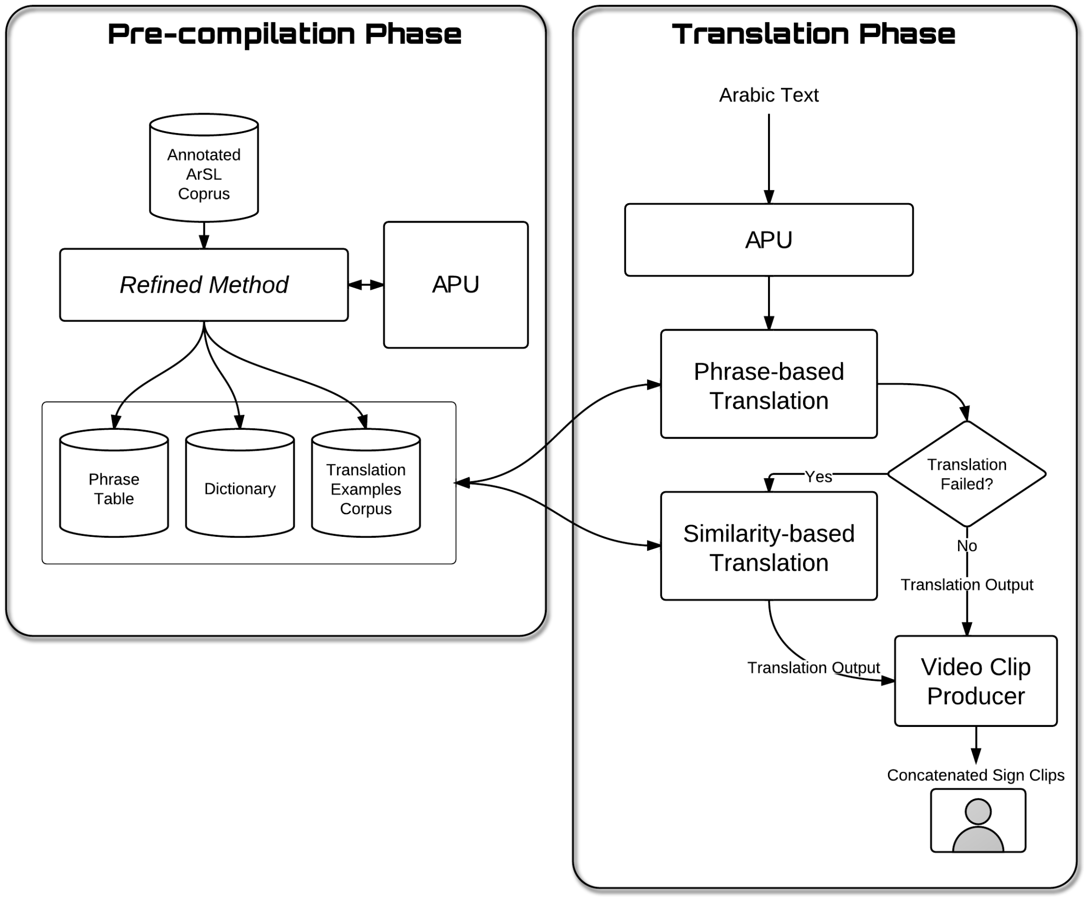


Figure 5.18: The main components of the ArSL EBMT system. Phase 1 is the pre-compilation phase, and Phase 2 is the translation phase.

learning the notation system that is used by the avatar; the process of entering and validating these notations to ensure that they are correct was also addressed. Most importantly, for a number of reasons the use of concatenated video is preferable to the use of an avatar in the present situation. First, it allows for an effective human and automatic evaluation of SLMT output-to-scale accuracy and enables translation problems to be addressed. This is because avatar research is still in the early stages of its development. The presence of an avatar means that the evaluator may not be able to distinguish between translation errors and errors committed by the avatar. Second, we believe that the number of researchers in the SLMT field will increase because most researchers have a lack of computer animation knowledge with respect to the way in which to address

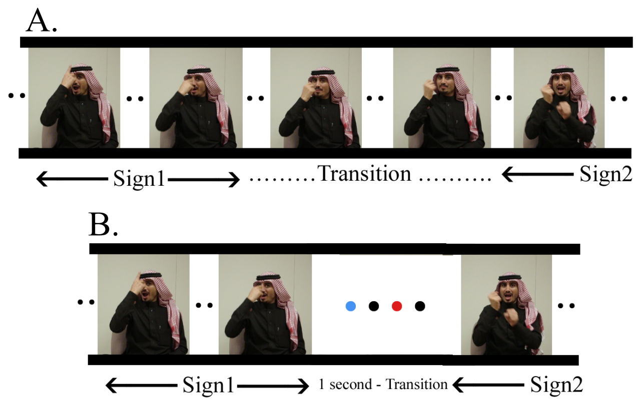


Figure 5.19: An example of natural representation (A) and the new method output representation (B).

the production of 3D signs and spatial reference points. Thus, we decided to use concatenated video-based sign synthesis. However, we decided not to use the methods described in Section 3.5.1, due to the problems that emerged when each of them was attempted. Hereafter, we present a simple and straightforward method for presenting the signs' output. This method produces a series of sign clips to represent the translation output; between each clip, a transition will be added (see Figure 5.19).

During MT processing, each translation system transfers the source words and phrases to their equivalents in the target language. In the case of SLMT translation, the translation system maps the source to intermediate representation data, which could be a notation system or actual geometric data of 2D and 3D representation to be used later for performing calculations and rendering avatar images to show signs. In contrast to the traditional SLMT translation that produces these intermediate representation data, SLMT systems that use concatenated video clips only map the source words and phrases to their equivalent sign video identifications (IDs). The ID is used later to retrieve the sign video clip from the corpus. Therefore, at the end of the translation process, the system will end up with a series of video IDs (see Figure 5.20). Each ID represents an isolated sign.

To show the output, a transition is added between each isolated sign. This transition is a one-second pause animation (Figure 5.19B) and can be changed upon the evaluator's request. The reason for adding this transition is that, in

Source Sentence	Word 1	Word 2	Word 3	Word 4
MT Processing	Word 1	ID 122	Word 3	Word 4
	Word 1	ID 122	ID 42	Word 4
	Word 1	ID 122	ID 42	ID 632
Sign Sentence	ID 932	ID 122	ID 42	ID 632

Figure 5.20: SLMT translation steps ending with a sequence number of sign video IDs.

reality, when a signer shows a sign and wants to move on to the next sign, the signer makes transitional motions to prepare MFs and NMFs for this next sign. This preparation is actually a transition, too (Figure 5.19A). However, because it is impossible to do this kind of preparation in real video, a pause has been added instead.

5.7.2 Evaluation

At the outset, we were concerned that the transitions would be visually disruptive and had the potential to upset comprehension. To explore this, the concatenated video output was tested by native signers on a team of three native signers plus one interpreter. Five hundred sign-test sentences were prepared, in which natural transitions were replaced by one-second pause transitions. Moreover, the sign in the actual sentences was replaced by the equivalent sign in the sign dictionary (see Figure 5.21). In addition, two five-point scales for *adequacy* and *fluency* are used for evaluation (LDC, 2005; Ma and Cieri, 2006), see Table 5.2.

Adequacy indicates the clarity of the meaning expressed in the produced concatenated video output, compared to the meaning of the original sign sentence. *Fluency* indicates the closeness of the produced concatenated video output to the natural signing way.

The test indicated that the meaning of the sentences was clearly expressed to the signers because all three of them scored all of the sentences a 5 (i.e., out of a possible 5-point total) when evaluating adequacy. Such an outcome is

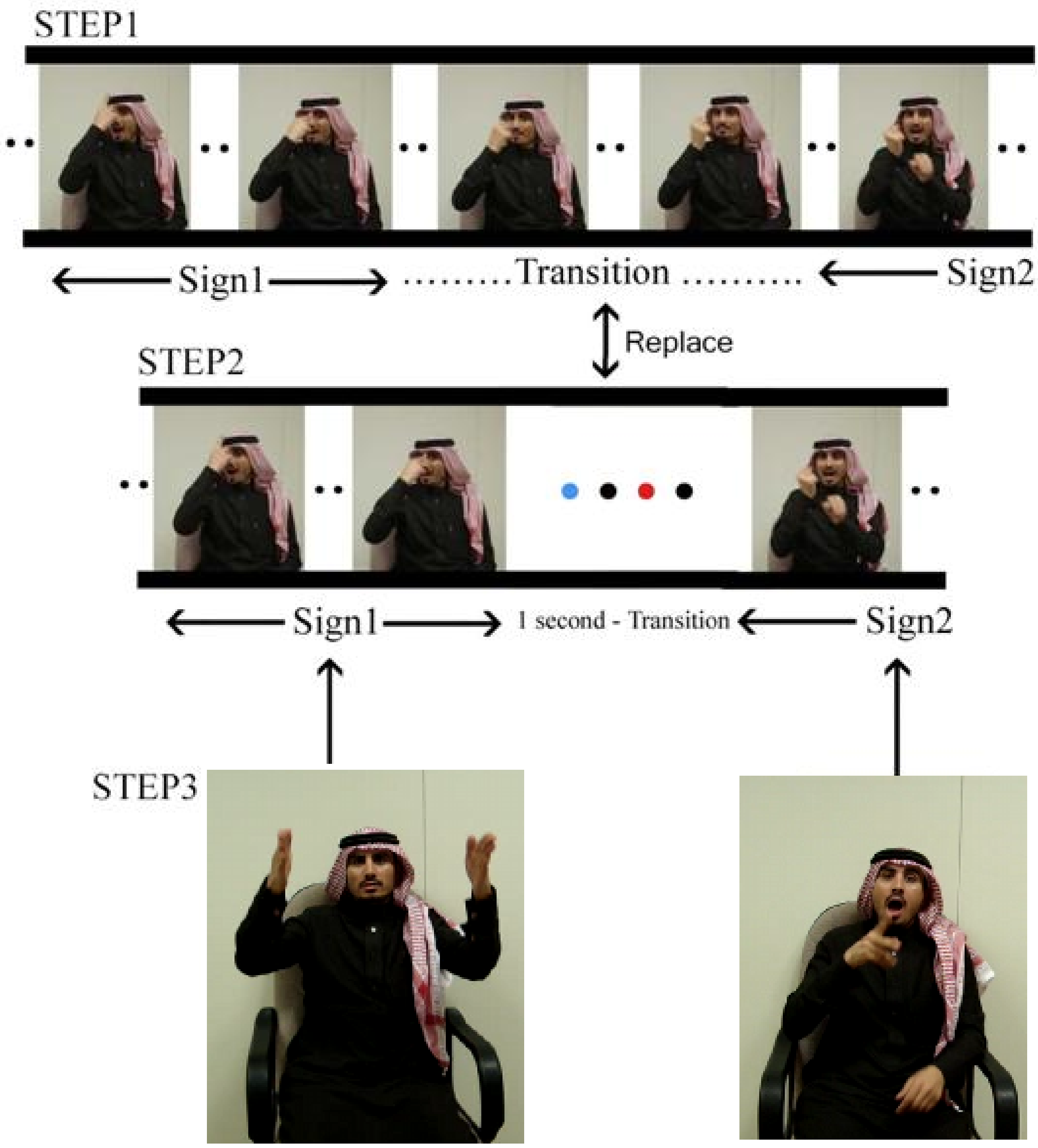


Figure 5.21: The signs in the original ArSL sentence replaced by an equivalent sign from the dictionary.

Table 5.2: The *adequacy* and *fluency* scales are explained in this table.

Adequacy		Fluency	
Scale	Description	Scale	Description
1	None	1	Not acceptable
2	Little Meaning	2	Diffluent (Not Fluent) Signs
3	Much Meaning	3	Acceptable
4	Most Meaning	4	Good Signs
5	All Meaning	5	Flawless Signs

proof that each sentence clearly expressed its meaning. In addition, to ensure that the team had grasped the correct meaning, we asked the team members to provide an explanation so that we could verify their evaluation points. Also, the fluency of the sentences was deemed acceptable, since the evaluators awarded 4 points out of 5. However, there is no easy way to interpret the fluency points. In terms of acceptability, it always appears that there were no issued gaps and that a rating of 4 points was given.

5.7.3 Summary

In this section, we proposed a straightforward method to present the translated ArSL output. It allows us to distinguish more easily between SLMT errors and sign synthesis errors. The method produces a series of sign clips; between each clip, it will insert a transition, a one-second pause animation, that can be changed upon request. The evaluation of the method showed that the meaning of the produced sentence was clearly expressed to the signers. Further, the fluency of the sentences was deemed acceptable. This method helps researchers to focus in the SLMT field. They may integrate their SLMT system later with a sophisticated signing avatar.

Chapter 6

Evaluation

This chapter will explain the two ways of evaluating a translated output: the human evaluation and using the automatic evaluation metrics. The benefits and drawbacks of each way have been discussed and explained. In addition, our methodology for evaluating our methods has been presented, too. Finally, we present and discuss the evaluation results of the five methods.

6.1 Human Evaluation

All of the automatic evaluation metrics aim to show a strong correlation with human judgment; human evaluation is also known as manual evaluation. This is because the human evaluators usually are people who will benefit from the tested translation system. Therefore, their judgments tend to be realistic and useful to test the accuracy of a system. However, it has some drawbacks, and these drawbacks can be summed up as follows:

- Human evaluation is expensive.
- It is time-consuming.
- It is subjective because the resulting evaluation scores typically are not identical to other evaluators' scores. Many factors can affect the score judgment, such as evaluator knowledge.
- It is subjective because the resulting evaluation scores typically may not match other evaluators' scores. Many factors can affect the score judgment, such as evaluator knowledge, ages, and so on.

Figure 6.1: The evaluation web site login page. The user and password box for login evaluator verification is located at the top left.

In addition to all of that for our system, sending the translated output to the evaluator is another problem. This is because the far distance between the ArSL deaf communities and us. As a result of that, we decided to test the feasibility of having an evaluation website to allow the deaf to log in from a distance and do the evaluation tasks. So we developed a website to test their ability to this. On this website, to upload the translated ArSL videos, we converted them to a SWF flash format and gave them a unique number that we increase by 1 each time we upload a new SWF ArSL video. Each evaluator has his own account and can log in to the website and start the evaluation (see Figure 6.1). Each evaluation score will be saved in a text file named by the evaluator user. One goal of this website was to allow each evaluator to freely log in without the need to gather with the rest of the team.

However, after uploading the translated output of the first EBMT, we found the following:

Figure 6.2: The evaluation page. The first edit box allows the evaluator and especially the interpreter to write the ArSL translation in Arabic. One of the four scores to judge the accuracy of the video can then be selected. The final box is for comments.

- Not all team members are familiar with technology. Before we left Riyadh, we trained the team to use the website, focusing on teaching the interpreter. However, deaf members faced difficulties, so they met with the interpreter and did the evaluation by accessing the website one by one under the supervision of the interpreter.

- Playing each SWF file takes time, as the uploaded files are large and the Internet connection speed was slow.

- More importantly, the evaluations took a long time to complete.

To overcome the connection speed problem, we developed evaluation stand-alone software that doesn't require a fast connection speed. The interpreter downloads this new software with the sign dictionary. Then, each time we want him to evaluate, we send the translation output as a text file that contains the number of sign clips so that the software automatically generates the translated video by concatenating these clips as well as the transition clip from the sign dictionary. Then from the same software, the evaluator performs the evaluation.

The evaluators did the first task, which is evaluating Three-Phase EBMT system. However, manual evaluation took a long time to finish by the deaf. Also, evaluation tasks are done only when the interpreter supervises the task. In addition, unfortunately, for the final attempt, which compared the next four real EBMT systems, the interpreter was out the country starting his PhD study; therefore, we only received his evaluation score for 60 ArSL sentences.

Regarding criteria of manual evaluation, in the first attempt, as shown in Figure 6.2, we were only concerned with the quality of the express meaning of the produced ArSL translation. In addition, we asked the interpreter to choose one of the four scores; the results are explained later in Table 6.2. We asked the interpreter to write the meaning of the ArSL output in Arabic in the top edit box.

After we developed the evaluation stand-alone software, we asked the interpreter to evaluate using two five-point scores for both fluency and adequacy (see Section 5.7.2). The evaluation of the video-based sign synthesis, as in Section 5.7.1, occurred during the collection of corpus 2. The SiER evaluation metric was also tested in person during the collection of corpus 1.

6.2 Automatic Evaluation

Automatic evaluation metrics were invented within the last decade. They are used to determine the accuracy of MT outputs. All of these metrics aim to show a good correlation with human evaluation. The advantages of using them over human evaluation are:

- They are systematic, so they run very quickly, finishing huge tasks in one press of a computer button.

- Very cheap and always available.
- They are consistent, as they will give the same results all the time. So they are very useful to compare two different translation systems if both are tested using the same automatic evaluation metric as well as same training sets and corpus.

Therefore, we start using automatic evaluation metrics and cross-validation between the first manual evaluation attempt using the website and the final manual evaluation attempt using the stand-alone software. We initially decided to choose the three commonly used evaluation metrics: Bilingual Evaluation Understudy (BLEU), Word Error Rate (WER), and Position-independent Word Error Rate (PER). In addition, we used our new SL evaluation metric, which we will see later in Section 6.3. BLEU (Papineni et al., 2002) is an n-grams-based metric. The score is calculated by comparing the candidate-translated output with the references-translated text. The summation of 4-gram, trigram, bigram, and unigram matches found for the candidate-translated output is divided by the summation of such matches found for the references. It was ideally design to use 3 or 4 n-grams. Also, it was ideally design to use more than one reference text. Since our corpora have included many sentences that have 1-sign or 2-sign lengths, the n-gram should be set to 1. However when we set the n-gram to 1, it will give the same result as the PER metric. Therefore, we end up deciding to use WER, PER, and our new metric.

WER is based on the Levenshtein distance (Levenshtein, 1966) on word level. This distance refers to the minimum number of word substitutions, deletions, and insertions one has to convert the candidate sentence into the reference sentence. WER is very sensitive to word location where word recording is not permitted. WER is calculated as follows:

$$WER = 100 \times \frac{I + S + D \times W_i}{T} \tag{6.1}$$

where I, S, D, W, and T are stand for insertions, substitutions, deletions, weight, and total signs in the reference sentence. WER assigns the same weight to insertions, substitutions, and deletions. In other words, consider the following example:

Candidate: `The woman saw little dog.`

Reference: `The man saw the little dog.`

We end up with one substitution, *'woman'*, and one deletion, *'the'*, and both, according to the WER metric, have the same weight. Also, it can be more than 100% if the number of words in the candidate sentence is higher than that of the reference sentence.

On the other hand, The PER metric is similar to WER, but it allows word reordering (Tillmann et al., 1997). It measures the differences in the count of the words occurring in the candidate and reference sentences. Therefore, PER is calculated as follows:

$$PER = \frac{max(|C|, |R|) - |C \bigcap R|}{|R|} \times 100 \tag{6.2}$$

Where C is the candidate sentence and R is the reference sentence. In addition, $|C|$ is the number of words in the candidate sentence. While $|R|$ is the number of words in the reference sentence.

6.3 A New Automatic Evaluation Metric for SL Translation

One of the major challenges for building any SL translation system is evaluating it. In general, two ways exist for evaluating SL translation output. First, one can use human judgment to assess the translation quality. This method is considered the most reliable way to evaluate any translation system, but it is expensive and time-consuming. Second, one can use existing automatic evaluation metrics. The problem with these metrics, however, is that they are designed to evaluate natural language, which has a different representation. Natural language representation is linear, while SL is multi-channel. Therefore, we present a new metric that is an extension of the WER metric, which is one of the most widespread evaluation metrics (Levenshtein, 1966).

6.3.1 Defining the Problem

Widely used automatic metrics have been designed to evaluate representations in linear sequence. These metrics fail to measure multi-channel sequences of SL; however, most research in the field of SLMT has evaluated using these metrics by discarding NMFs and combining MFs as one linear output (Morrissey, 2008) or by considering the entire sign as one block and then combining MFs and NMFs in linear representation (Segundo et al., 2007). In some cases, when NMFs are discarded, a completely unrealistic evaluation score results. For example, the sign for "theft" would be seen as the sign for "lemon" because "lemon" shares all of the manual features of "theft." In addition, when the sign is treated as one block, the metric score is usually unrealistic, specifically in cases in which NMFs are deleted or non-existent NMFs are inserted in signs. Measurements, in these cases, are equivalent to the score of signs that have extra MFs or to a sign that is completely different from the original sign and that shares no NMFs or MFs with the original sign. Therefore, an evaluation metric for SL that agrees with human judgments while automatically generating scores is strongly needed.

6.3.2 Sign Language Error Rate (SiER) technique

Before presenting the new metric, it is important to present the original WER metric since the new SiER is an extension to it. The formula for SiER is

$$SiER = 100 \times \frac{\sum_{i=1}^{n} \alpha}{T} \tag{6.3}$$

and

$$\alpha = \begin{cases} 1, & \text{if } I_{MF} = 1 \\ (D_{MFs} + S_{MFs}) \times \beta + NS, & \text{otherwise} \end{cases} \tag{6.4}$$

and

$$\beta = \begin{cases} \frac{\gamma}{T_{MFs}}, & \text{if } T_{NMFs} \geqslant 1 \\ \frac{\gamma+\delta}{T_{MFs}}, & \text{if } T_{NMFs} = 0 \end{cases} \tag{6.5}$$

$$NS = \begin{cases} 0, & \text{if } T_{NMFs} = 0 \\ (D_{NMFs} + S_{NMFs}) \times \frac{\delta}{T_{NMFs}}, & \text{if } T_{NMFs} \geqslant 1 \end{cases} \tag{6.6}$$

where n is the total number of candidate signs. I_{MF}, D_{MFs}, and S_{MFs} refer to the total number of the insertion, deletions, and substitution of MF for $Sign_i$. α is the total error score of candidate $Sign_i$. While γ is the total weight that falls between 0 and 1 for each existing MF. β is the weight given for each existing MFs. δ is the total weight given for all existing NMFs. NS is the total errors score given for all NMFs in candidate $Sign_i$, and it falls between 0 and δ. T_{MFs} is the total numbers of MFs in the reference $Sign_i$. T_{NMFs} is the total number of NMFs in the reference $Sign_i$. Factors considered when the formula was designed are: (1) the sign has at least one MF; (2) adding a new MF feature changes the meaning of the sign; (3) useless NMF features are usually inherited from previous signs and do not affect the meaning of the sign (these are naturally discarded by a native signer); (4) the quality of signs; (5) according to the ArSL corpus, there is at most one essential NMF in a sign; and (6) finally, it is important to note that the number of essential NMFs in the corpus compared with the number of quality NMFs is very limited.

First, in computing SiER, the automated procedure determines whether any NMFs or MFs have been substituted or deleted, or whether MFs have been inserted. Second, it distributes the weight of each NMF and MF, assigning γ to the right hand (RH) feature if the RH feature exists and assigning γ to the left hand (LH) feature if the LH feature exists; otherwise, it assigns β to the only existing MF. It then breaks up the δ weight for all NMFs that exist in the reference sign sentence. If no NMF exists, it adds the δ to the MF weight by adding δ divided by 2 to each MF, or it adds δ to the sole MF if only one exists. Third, it adds the difference for each NMF and MF, and then it multiplies that sum by the NMF and/or MF weight. Finally, it divides the final score by the total number of signs in the reference sentence. The result is then multiplied by 100 to transform the rate into a percentage. SiER can be more than 100% when the numbers of MFs insertions are high and the candidate has more MFs numbers of reference sentences. In addition, the Sign Recognition Rate (SiRR) can be calculated from SiER by subtracting 100 from SiER.

Figure 6.3 shows a simple alignment between a reference and candidate sign sentence from the ArSL corpus, including the types of differences between the two. Applying the traditional WER technique to evaluate the candidate sign sentence, first when we discard NMFs from signs, we get

	Sign1	Sign2
Reference – RH Gloss	THEFT	FORBIDDEN
LH Gloss		
Eyes Gloss	closed	

Candidate – RH Gloss	THEFT	CRIME
LH Gloss		
Eyes Gloss		

Figure 6.3: A simple alignment between a reference and candidate sign sentence. The Gloss Notations for ArSL are translated from Arabic to English.

Applying the traditional WER technique to evaluate the candidate sign sentence, first when we discard NMFs from signs, we get

$$WER = \frac{0+1}{2} \times 100 = 50\%$$

In this case, sign1 in the candidate matches sign1 in the reference sentence since NMF has been discarded. Also, we have one substitution, = *'Crime'*. On the other hand, when we take into account both MFs and NMFs, we end up with

$$WER = \frac{1+1}{2} \times 100 = 100\%$$

Here, we have one NMF deletion, =*'closed'*, and one substitution, = *'Crime'*, and we divided by the total number of signs in the reference sentence. On the other hand, when the SiER is applied and $\gamma = 0.7$ and $\delta = 0.3$ parameters have been chosen, when end up with

$$SiER = \frac{0.3+1}{2} \times 100 = 65\%$$

where the SiER is equal to 65%, which shows a better correlation.

6.3.3 Evaluation

To evaluate and test SiER, a set of Gloss Notations, as in Figure 6.3, was manually created. The set contained 8 groups, each with 5 Gloss Notations, with the groups in each set including (1) an extra MF, (2) an extra NMF, (3) a deleted MF, (4) a deleted essential MF, (5) a deleted quality NMF, (6) a substituted MF, (7) a substituted essential NMF, and (8) a substituted quality NMF.

The test was conducted by two native signers and one interpreter. The interpreter read the Gloss Notation and mimed it exactly as written in the notation, then received feedback. After that, he put it into context and received feedback (see Table 6.1).

Table reftable:result shows the average manual evaluation results for each group. Also, we calculated the WER for each group: WER1 takes into account both MFs and NMFs and deals with them as one block, whereas WER2 is the error rate for only MFs. In addition, SiER has been calculated for each group and with different δ and γ weights. To show the correlation clearly, WER, WER2, and SiER were calculated based on a reference that has one sign. The manual evaluation was given a score from 0 to 5 for each feedback, and the average feedback for each group is provided in the table.

As Table reftable:result shows, different δ and γ weights were tried for SiER. Also, a scatter diagram has been drawn between each SiER result and human judgment, as shown in Figures 6.4, 6.5, 6.6, 6.7, 6.8, 6.9, 6.10, 6.11 and 6.12. In addition, a linear regression line, which was forced to start from origin point zero, has been drawn for each figure. Also, R-squared values, which are the square of the Pearson product moment correlation coefficient, have been calculated and displayed in each figure. In the end, we chose the weights $\delta = 0.1$ and $\gamma = 0.9$ to be the ideal parameters because they give the highest R-squared value (0.82), as shown in Table 6.1.

Table 6.1 shows the average manual evaluation results for each group and the correlations between it and WER1, WER2, and SiER. WER2 is the error rate only for MFs, while WER1 considers both MFs and NMFs as equals. To show the correlation clearly, WER1, WER2, and SiER were calculated based on a reference that has one sign. The manual evaluation was given a score from 0 to 5 for each feedback, and the average feedback for each group was provided in the table. score 0 means the sign is of high fidelity and fluent; score 1 is given when the sign shows the correct meaning but the evaluators felt a little confused. score 2 is given when the sign demonstrates its main meaning but cannot be fluently fitted into the context. score 3 is given when a part of the sign is known. Based on the evaluator's knowledge, the missing part could hardly verify its meaning.

Table 6.1: Manual evaluation results and correlations between manual evaluation and WER1, WER2, and SiER with different δ and γ weights. The human judgment score of 0 is equal to a 0% error rate, 1 is equal to 20%, 2 is equal to 40%, 3 is equal to 60%, 4 is equal to 80%, and 5 is equal to 100%.

				SiER								
	Human Judgment	**WER1**	**WER2**	$\delta = 0.1, \gamma = 0.9$	$\delta = 0.2, \gamma = 0.8$	$\delta = 0.3, \gamma = 0.7$	$\delta = 0.4, \gamma = 0.6$	$\delta = 0.5, \gamma = 0.5$	$\delta = 0.6, \gamma = 0.4$	$\delta = 0.7, \gamma = 0.3$	$\delta = 0.8, \gamma = 0.2$	$\delta = 0.9, \gamma = 0.1$
Group 1	5=100	100	100	100	100	100	100	100	100	100	100	100
Group 2	0=0	100	0	0	0	0	0	0	0	0	0	0
Group 3	3=60	100	100	45	40	35	30	25	20	15	10	5
Group 4	1.5=30	100	0	10	20	30	40	50	60	70	80	90
Group 5	1=20	100	0	10	20	30	40	50	60	70	80	90
Group 6	4=80	100	100	45	40	35	30	25	20	15	10	50
Group 7	2=40	100	0	10	20	30	40	50	60	70	80	90
Group 8	0.5=10	100	0	10	20	30	40	50	60	70	80	90
		R-square:	0.70	0.82	0.79	0.62	0.30	-0.07	-0.38	-0.57	-0.68	-0.72

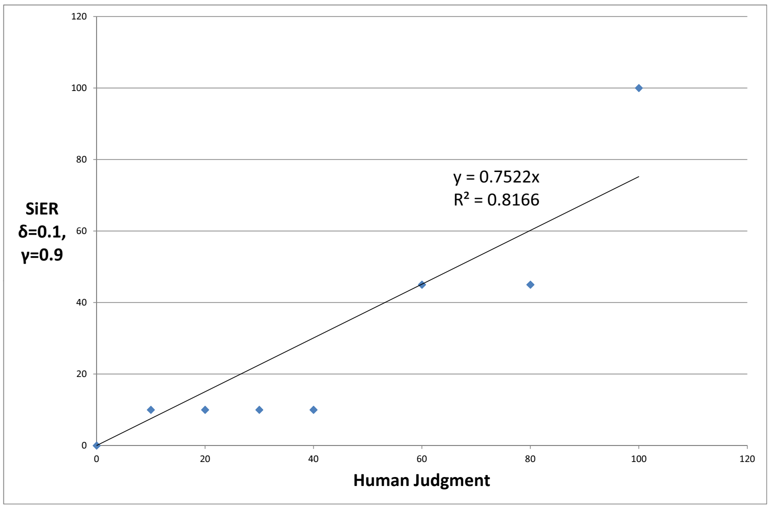


Figure 6.4: Scatter plots between human judgments against SiER, where γ=0.9 and δ= 0.1.

score 4 is given when a small detail of the sign is known but evaluators could not determine the meaning of the missing part. score 5 is given in the case of a completely unknown sign.

Regarding the results, for group 1, the evaluators had to distinguish between the extra MF in cases in which it is not at all related to SL, such as picking up a pen from the table while signing with the other hand (this case is discarded naturally by the viewer and does not affect the meaning of the sign), and between the actual movement as a part of the sign. Therefore, the first case was omitted from the results. In group 2, in general, adding NMFs to the sign was naturally discarded by the viewer and did not affect the sign. This has been clearly shown in the evaluators' feedback. For groups 4, 5, 7, and 8, the evaluators were able to determine the meaning when the sign was added to a context.

6.3.4 Summary

A new evaluation approach for SLMT was proposed to extend WER measurement methods for multi-channel representation. It also takes into account that each feature in the representation has a different impact on the evaluation score. The

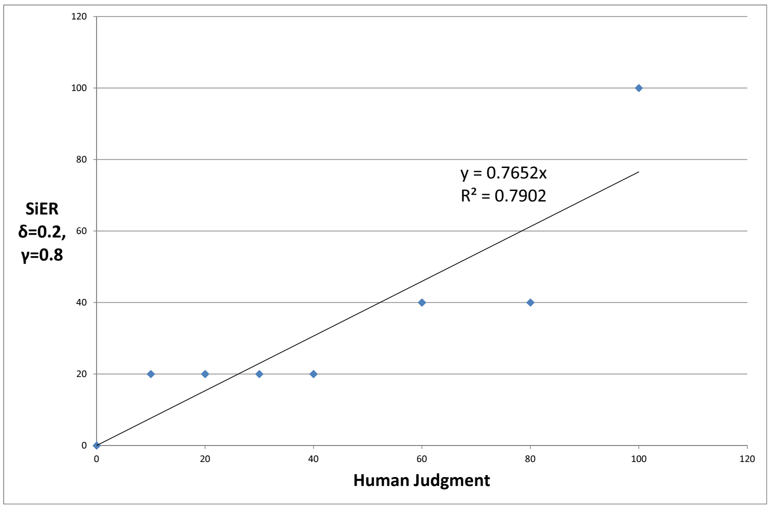


Figure 6.5: Scatter plots between human judgments against SiER, where γ=0.8 and δ= 0.2.

idea behind it is simply to assign a weight to each MF and NMF, considering some facts about SL, such as the insertion of NMF, which has no impact on the meaning of the sign. The experiments show the new approach is promising and, most of the time, more realistic than WER evaluation scores, especially for NMFs. This approach opens the door for further investigations into multi-channel evaluation techniques.

6.4 Cross Validation

Many SLMT systems, such as the one mentioned in Section 3.2.2, have created their own test sets manually, according to some criteria. In addition, the number of test sets is usually much smaller than the number of corpus training sentences, as in Morrissey (2008), who created 55 sentences as test sets with a system containing 577 sentences in the corpus. Using a single test set, which is common, introduces possible problems; for instance, using a single test set will provide only a single-point estimation score. This does not provide confident and precise evaluation scores that average many estimation points. In other words, using only a single-point estimation score will not provide a confident score of how the

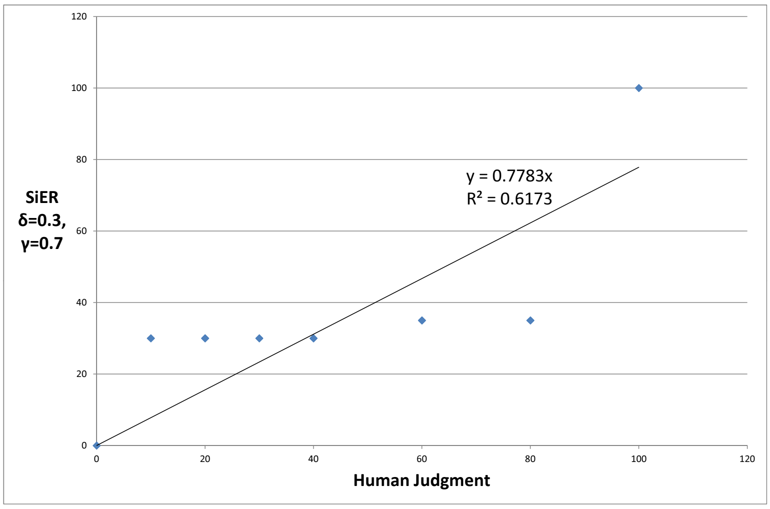


Figure 6.6: Scatter plots between human judgments against SiER, where γ=0.7 and δ= 0.3.

translation system will perform in practice. Therefore, we decided to employ cross-validation to test our implemented EBMT methods. We chose the Leave-One-Out Cross-Validation (LOOCV), since as it is a preferred technique and is computationally feasible because we use small corpora (TuBao, 2000).

We used this technique to perform the evaluation N times, where N equal to the number of sentences in the corpus. Figure 6.13 shows one sentence as a testing input and the remaining N-1 sentences used as training sets prior to evaluation. It then moves to the next sentence, using it as testing sentence. The remaining N-1 sentences are used as training sets, including the sentences before the selected testing sentence, and the system then perform the evaluation. The system repeats this procedure N times. It then averages the evaluation scores.

6.5 Results and Discussion

For evaluating the constructed five methods, we start by manually evaluating the 3-phases method. A test set was created for this purpose. In addition, corpus 1 was used for training the system. Hence, corpus 2 at that time had not

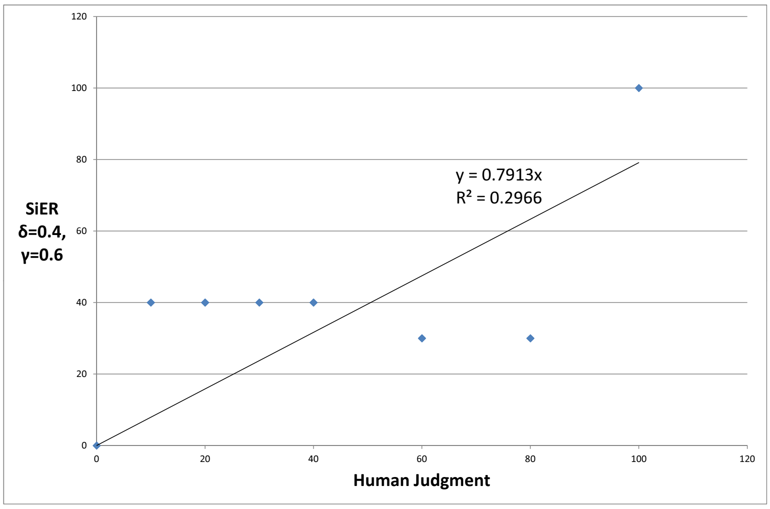


Figure 6.7: Scatter plots between human judgments against SiER, where γ=0.6 and δ= 0.4.

been constructed yet. We divided the test set into three groups and each group consists of eight sentences:

- The first group evaluates the sentences that were taken directly from the corpus.
- The second evaluates the sentences that have an identical structure with any example in the corpus.
- The third evaluates the sentences that have a structure not close to any example in the corpus.

Table 6.2 summary of the manual evaluation scores. The results show that the accuracies of group 1 and group 2 are acceptable, while in group 3, the system generally shows poor translation accuracy. The evaluation was based on four scores. The six tags used failed to give the system enough data, such as the person's gender.

Finally, as mentioned in Section 5.3, we implemented this method only to make sure that our first corpus was ready for MT and to test whether or not the evaluation Web site was practical for use. In addition, at that time, the

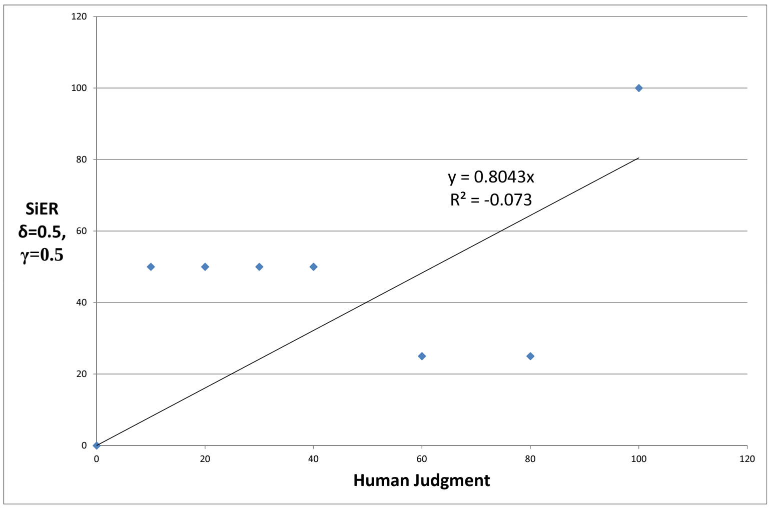


Figure 6.8: Scatter plots between human judgments against SiER, where γ=0.5 and δ= 0.5.

Table 6.2: Summary of the manual scores received for each signer including the interpreter.

	Good	Fair	Poor	Bad
Group 1	32	0	0	0
Group 2	6	9	13	4
Group 3	4	0	5	23

morphological analyser, which produces a variety of morphological details and the roots and stems of words, was not used to increase the accuracy of the translation.

For the remaining four methods, we acquired the complete evaluation results by LOOCV. Table 6.3 shows the evaluation results. The WER, PER, and SiER of the phrase-based method are 63.1%, 45.7% and 63.1%, respectively, for corpus 1 and 58.5%, 32.4% and 58.5%, respectively, for corpus 2. The number of failed translations using the phrase-based method was 94 sentences in corpus 1 and 235 sentences in corpus 2. Failure in phrase-based translation means that no phrase matches were found for any portion of the input sentences. After eliminating the failed translation sentences, the WER, PER, and SiER of the

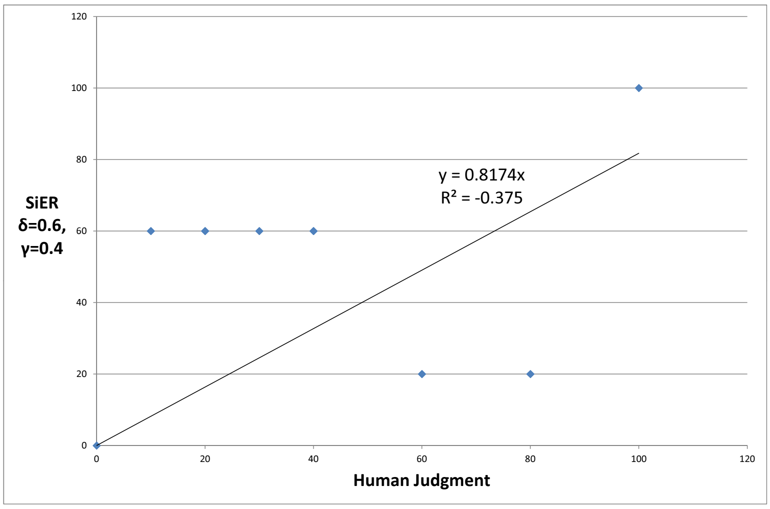


Figure 6.9: Scatter plots between human judgments against SiER, where γ=0.4 and δ= 0.6.

phrase-based translation are 31.3%, 24.7% and 31.3%, respectively, for corpus 1 and 41.6%, 21.3% and 41.6%, respectively, for corpus 2, with 109 successfully translated sentences for corpus 1 and 578 successfully translated sentences for corpus 2. Regarding the classical method, the WER and PER are high for both corpora, partly because this method does not employ the morphological analyser to increase the chances of matches.

It is clear from the results that the phrase-based method is more accurate than the similarity-based or classical methods for those sentences for which complete matches are found. However, the drawback to the phrase-based method is its high failure rate. The remaining failed input sentences require some additional process to produce an acceptable sentence. We decided to use the similarity-based method for this purpose, resulting in the "combined" figure in the last row of Table 6.3(a) and (b). These figures clearly show that combining the phrase- and similarity-based methods by forwarding the failed phrase-based translation to the similarity-based method significantly increases the level of accuracy.
The results also show the noticeable impact of the complexity of the corpus. A lower error rate was achieved by corpus 1 because about 75% of its sentences have four signs or fewer, whereas only about 25% of the sentences in corpus 2

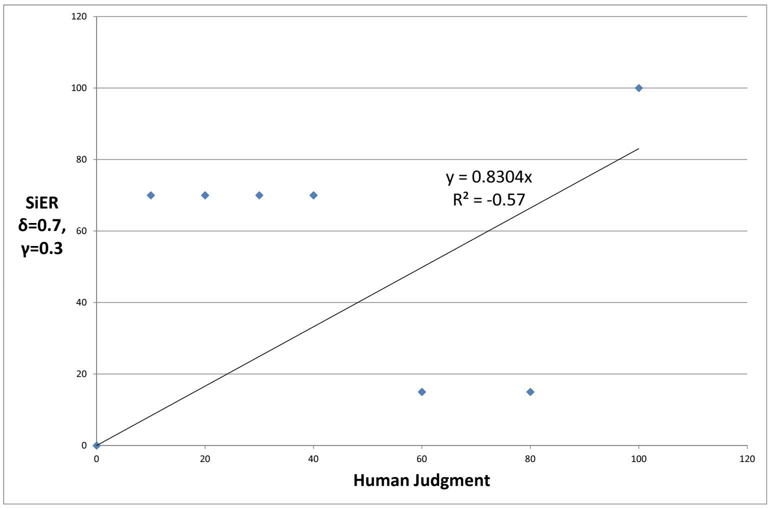


Figure 6.10: Scatter plots between human judgments against SiER, where γ=0.3 and δ= 0.7.

have four signs or fewer. In addition, since corpus 2 was constructed from a children's story, the sentences are highly correlated. This can be seen from the relatively low number of failed phrase-based translation sentences in corpus 2, namely 235, or about 28.9%, whereas the corresponding number in corpus 1 is 94, or about 46.3%. It is important to note that since sign output is produced using high-quality pre-recorded signs, each sign is guaranteed to have all of its NMFs and MFs. Therefore, the SiER metric results for our system are similar to WER results.

Finally, it is worth mentioning that we sent the full translation output of the combined method to the interpreter in order to allow a deaf team to do the evaluation task. However, as we stated in Section 6.1, the interpreter was out of the country beginning his PhD study, so we received only his evaluation score of 60 sentences. Human evaluation of a language must be done by a native speaker. Therefore, we cannot take into account the interpreter's evaluation sentences. Table 6.4 summarises the results of the received 60 sentences.

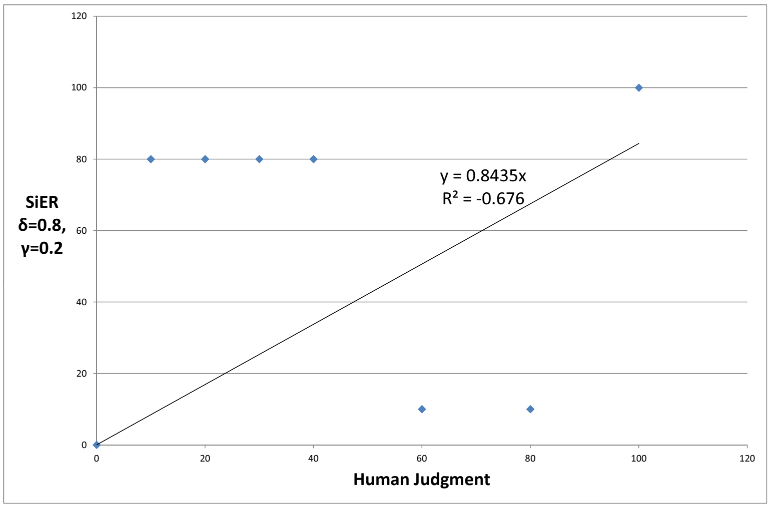


Figure 6.11: Scatter plots between human judgments against SiER, where γ=0.2 and δ= 0.8.

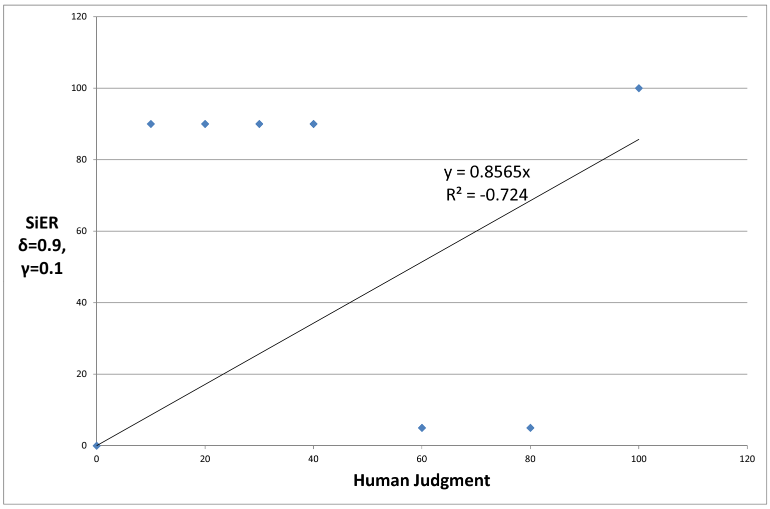


Figure 6.12: Scatter plots between human judgments against SiER, where γ=0.1 and δ= 0.9.

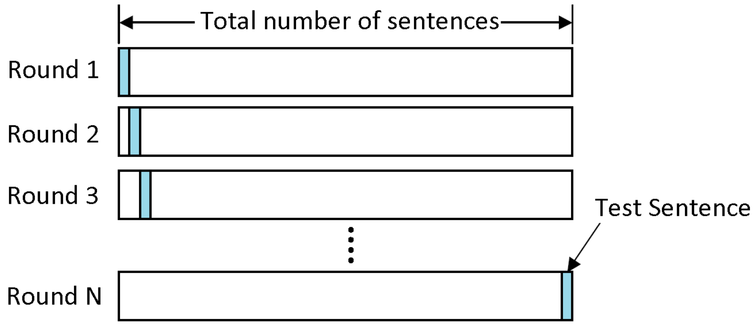


Figure 6.13: Illustration of LOOCV.

(a) Corpus 1 (203 sentences, 710 signs)

Translation Method	WER (%)	PER (%)	SiER (%)
Classical	68.4	49.4	68.4
Phrase-based	63.1	45.7	63.1
Similarity-based	43.5	36.7	43.5
Combining Similarity- and Phrase-Based	36.2	26.9	36.2

(b) Corpus 2: (813 sentences, 2478 signs)

Translation Method	WER (%)	PER (%)	SiER (%)
Classical	73.1	39.5	73.1
Phrase-based	58.5	32.4	58.5
Similarity-based	48.2	29.3	48.2
Combining Similarity- and Phrase-Based	44.0	28.1	44.0

Table 6.3: Evaluation scores of the classical method, as well as all the methods used in the system.

Adequacy

score	5	4	3	2	1
# of Sentences	38	13	5	1	3

Table 6.4: Shows the evaluation results of only 60 sentences that were taken from Corpus 1 and evaluated by the interpreter. The descriptions of the scores are in Table 5.2.

Chapter 7

Conclusions and Future Work

SL has started receiving more attention from many scientists in the fields of computational linguistics and computer animation. Scientists in these fields are attracted to SL because it is a visual language and has a multi-channel representation, unlike traditional written languages, and building a translation system for SL is required to overcome some major challenges. As a result, many efforts have been made to build local SL corpora for different purposes, such as helping researchers to study SL linguistic characteristics or being employed in a corpus-based SLMT system (for more details, see section 3.4). Many SL annotation tools, such as the widely used ELAN tool, have been released publically to help construct these corpora (see section 4.1 for more tools). Despite these great efforts, existing SL corpora are still very limited in size compared to the size of text-to-text corpora, due to the fact that text-to-text corpora is constructed from existing documents, whereas SL has no standard writing system and therefore no documents. Therefore, as discussed in section 3.3, some notation systems for daily use have been introduced, such as Sutton's SignWriting. However, these daily-use notations are not popular among the deaf community, and many in the community still do not recognise these notations. Other notation systems, such as HamNoSys, have also been introduced for research purposes to be used, for example, by SL avatars. HamNoSys is considered the most widely used notation, and the University of Hamburg continues to improve it and release new versions. Because SL is a visual language without a standard writing system, it is necessary to find a way to allow text-to-SL MT systems to present the translated SL output in space. As described in section 3.5, some attempts have been made to build

SL avatars. Most of these avatars still in the development stage and still have limitations, such as with performing some NMFs. In general, avatar studies are limited; however, the EA avatar seems promising, as it continues to improve and produce accurate SL animation output. Also, there have been many attempts to build translation systems between SLs and written languages (section 3.2 introduced many of these systems). Some of these systems are rule-based, where they only translate based on pre-stored linguistic rules. Others are corpus-based, where they employ a corpus for translation. The translation accuracy achieved today is not considered high compared to what has been achieved between written languages. The main reason for this result with rule-based systems is that there are fewer linguistic studies of SLs compared to the number of studies on written languages. Also, regarding corpus-based systems, the main reason for this result is that the size of the used translation corpus is limited. Evaluating SLMT translation output is another challenge because SL has a multi-channel representation. The number of studies in this area is very limited and needs more attention.

This dissertation has tackled the major challenges of developing a translation system for SL. These challenges can be summed up as follows:

- The complexity of the procedure for building the SLMT corpora.
- The size limitation and coverage of SLMT corpora.
- How to evaluate the translated SL output?
- How to produce the translated SL output for the deaf?
- What is the suitable SL translation approach?

Therefore, with regard to building a bidirectional translation system from Arabic text into ArSL, we introduced new techniques that help to do the following:

- Build the corpus.
- Increase its coverage.
- Present the translated ArSL sentence in a way that ordinary deaf people can understand it.

- Evaluate the multi-channel output.

In addition, we did an investigation for selecting the suitable approach for our system as well as adapted and modified this selected approach method to come up with the best translation accuracy that can be obtained.

Therefore, with regard to building the ArSL corpora, we constructed the first two ArSL corpora to be used in the system. One has 710 signs and 203 signed sentences and its content is restricted to the domain of instructional language typically used in deaf education. The second corpus was constructed from a children's story and contains 2,478 signs and 813 signed sentences. Both ArSL corpora were heavily influenced by Saudi local SL because of the background of the signing team. These corpora were built under a serious quality-control responsibility. The team had to ensure that the translated sign sentences are fluent, complete, and clear for all ages of native signers and understandable by natives with different educational backgrounds and that the translated sign sentences are fully independent from the original Arabic sentences. After recording the sentence, we and the corpus team annotated the recorded videos using ELAN by adding a gloss transcript to all manual and non-manual features of all recorded signs. Then, using our developed compiling tool, we deliver our final corpus. This compiling tool should be largely adaptable to the task of translations from other written languages into other sign languages.

In addition to building the two ArSL corpora and developing the special-purpose compiling tool in order to construct the SLMT corpora, we employed Arabic morphological information to increase coverage and thus translation accuracy because all Arabic words that share the same root are related in meaning. In addition to a morphological analyser, we introduced a new technique to reduce ambiguity among a number of words so that the analyser will select the correct analysed word from among those that are produced and suggested by the morphological analyser. As a technical contribution in addition to the compiling tool, we rewrote the Al-Khalil morphological analyser to make it pluggable to any NLP system. The original source code was written in Java, and we rewrote it using Embarcadero Delphi XE, which uses Object Pascal code. We also added our new technique to reduce ambiguity and select the correct analysed words.

We decide to use the concatenated sign video clips as output rather than a signing avatar for two reasons. This format is simple, and it allows us to distinguish more easily between translation errors and sign synthesis errors. Therefore, we present a straightforward technique based on concatenated sign video clips that helps the evaluator to examine the translation output. At the outset, we were concerned that the transitions would be visually disruptive and could impair comprehension. To explore this possibility, the concatenated video output was tested by native signers in a team of three native signers plus one interpreter.

In addition to this sign synthesis, automatic evaluation metrics of MT systems have been designed to evaluate the linear sequence representations of languages. It fails when the researchers attempt to measure the multi-linear sequences of SL. Therefore, an evaluation metric for SL that agrees with human judgments while automatically generating scores is urgently needed. For that reason, we present a new technique called SiER that is an extension of the WER metric.

Further, we did some investigation into the existing translation approaches, and we came to the conclusion to choose an EBMT approach. This decision was made with many consideration factors, including the lack of linguistic studies of ArSL, the high cost of building a corpus, and the cost of extending the coverage of the translation approach. Therefore, we started evaluating four different EBMT methods. Then, we combined two of them to produce a more accurate translation output. We believe that we are the first to build a complete end-to-end ArSL corpus-based translation system, the first to solve the lexical and structural transfer from Arabic text to ArSL, and the first to use standard evaluation methods including both automatic evaluation metrics as well as human evaluation to examine the output of the system. The system displayed a high performance with the first corpus, with a WER of 36.2% and an average PER of 26.9%. With the second corpus, it produced translated sign sentence outputs with an average WER of 44% and an average PER rate of 28.1%.

The majority of the techniques that have been built are transferable to other natural language engineering tasks. The compiling tool that we developed to build the corpora can be used for any SL corpus-based translation system. In addition, the techniques we formulated to prevent word ambiguity can be used in other applications that require Arabic pre-processing, such as information retrieval.

While this dissertation has described a complete Arabic text into ArSL translation system and has dealt with many issues to able to successfully build this system, there are a number of potential future works that can be summed up as the following:

- Start building a large ArSL corpora in a long-term project like the BSL project that has been described in Section 3.4.1.3. This kind of corpus will be a great help in increasing SLMT translation accuracy. In addition, it will have a high impact for linguistic experts in term of getting a better understanding of the ArSL language. After we saw the phrase-based EBMT show a high accuracy among other EBMT methods, we started wondering if a phrase-based SMT would show a better result. Having a large corpus will allow us to try answering this question since SMT needs a large corpus.

- In addition, building or integrating our SLMT system with a signing avatar is another future work. One of candidate avatars is the UEA avatar described in Section 3.5.2.1. In order to integrate with this avatar, it is necessary to add a HamNoSys notation to each sign in the dictionary, which will first require training for using this notation by some of the signing team. Another option is by trying to build our own avatar and storing the motion captions of each sign in the dictionary by using some recent sophisticated motion capture devices. These details are sent to the avatar from the dictionary in order to generate the signing output. This option will require some expensive equipment that can capture all the details for the MFs and NMFs.

- By using a signing avatar, we can start dealing with spatial reference points.

- Integrating the SLMT system with Arabic voice recognition is also a goal. Therefore, we may start investigating this issue.

Finally, we believe these works need to attract more attention from the local research and disabilities center and it should open its doors for more collaboration and funding support, which would help us achieve our future goals. In addition, we will start building a wider network within the deaf community to support these future works.

Bibliography

Mahmoud Abdel-Fateh. Arabic Sign Language: A perspective. *Journal of Deaf Studies and Deaf Education*, 10(2):212–221, 2005. x, xi, 18, 20, 21, 25, 46

Ahmed Abdelali, Jim Cowie, and Hamady S. Soliman. Arabic Information Retrieval Perspectives. In *Proceedings of the 11th Conference on Natural Language Processing, JEP-TALN*, pages 19–22, Fez, Morocco, 2004. 14

Lahsen Abouenour, Said El Hassani, Tawfiq Yazidy, Karim Bouzoubaa, and Abdelfattah Hamdani. Building an Arabic Morphological Analyzer as part of an Open Arabic NLP Platform. In *Proceedings of the Workshop on HLT&NLP within the Arabic World, LREC*, Marrakech, Marocco, 2008. 35

Mohammed Al-Binali and Samir Samareen. *Grammar of the Unified Qatari Arabic Sign Language*. Dar Al-Sharq, Doha, Qatar, 2009. In Arabic. x, xi, 16, 17, 19, 21, 23, 24, 101

Hend Al-Khalifa. Introducing Arabic sign language for mobile phones. In *Proceedings of the 12th International Conference on Computers Helping People with Special Needs, ICCHP*, pages 213–220 in Springer Lecture Notes in Computer Science, Part II, vol. 6180, Linz, Austria, 2010. 47, 48

Tariq S. Al-Raiyis. For the First Time, Deaf Students Study at King Saud University. Newspaper article, 2011. URL `http://www.al-jazirah.com/20110913/fe16d.htm`. In Arabic. 3

Imad A. Al-Sughaiyer and Ibrahim A. Al-Kharashi. Rule parser for arabic stemmer. In *Proceedings of the 5th International Conference on Text, Speech and Dialogue, TSD*, pages 11–18, Brno, Czech Republic, 2002. 33

Imad A. Al-Sughaiyer and Ibrahim A. Al-Kharashi. Arabic Morphological Analysis Techniques: A Comprehensive Survey. *Journal of the American Society for Information Science and Technology*, 55(3):189–213, 2004. xi, 15, 33, 34, 35

Muhammad A. Alkhuli. *A Dictionary of Theoretical Linguistics: English-Arabic with an Arabic-English Glossary*. Library of Lebanon, Beirut, Lebanon, 1982. 15

Colin Allen. Global Survey Report by WFD Interim Regional Secretariat for Arab Region, WFD RSAR. Technical Report No. 7, Regional Survey Report, World Federation of the Deaf and Swedish National Association of the Deaf, Finland, 2008. 2, 4, 16

Abdulaziz Almohimeed, Mike Wald, and R. I. Damper. A new evaluation approach for sign language machine translation. In *Assistive Technology from Adapted Equipment to Inclusive Environments, AAATE 2009, Volume 25*, pages 498–502, Florence, Italy, 2009. 69

Abdulaziz Almohimeed, Mike Wald, and Robert Damper. An Arabic Sign Language corpus for instructional language in school. In *Proceedings of the 7th International Conference on Language Resources and Evaluation, LREC*, pages 81–91, Valetta, Malta, 2010. 74

Abeer Alnafjan. Tawasoul. Master's thesis, Department of Computer Science, King Saud University, Riyadh, Saudi Arabia, 2008. 48

Mohammad Altabbaa, Ammar Al-Zaraee, and Mohammad Arif Shukair. Qutuf: An Arabic Morphological Analyzer and Part-of-Speech Tagger. Master's thesis, Arab International University, Damascus, Syria, 2010. 37

Faisal Al Ameiri, Mohamed Jamal Zemerly, and Mohamed Al Marzouqi. Mobile Arabic Sign Language. In *Proceedings of the 6th International Conference on Internet Technology and Secured Translation*, pages 363–367, Abu Dhabi, United Arab Emierates, 2011. 47

Tantely Andriamanankasina, Kenji Araki, , and Koji Tochinai. Example-based machine translation of part-of-speech tagged sentences by recursive division. In

Proceedings of Machine Translation Summit VII, "MT in the Great Translation Era", pages 509–517, 1999. 109, 110

Doug Arnold, Lorna Balkan, Siety Meijer, R.Lee Humphreys, and Louisa Sadler. *Machine translation: an introductory guide.* NCC Blackwell, London, UK, 1994. 26

Mohammed A. Attia. *Handling Arabic Morphological and Syntactic Ambiguity withing the LFG Framework with a View to Machine Translation.* PhD thesis, University of Manchester, Manchester, UK, 2008. 35, 36

Stan Augarten. *Bit by Bit: An Illustrated History of Computers.* Tickner and Fields, New York, NY, 1984. 4

R. Harald Baayen. *Word Frequency Distributions.* Kluwer Academic Publishers, Dordrecht, Netherlands, 2001. 29

J. Andrew Bangham, Stephen Cox, Ralph Elliott, John Glauert, Ian Marshall, Sanja Rankov, and Mark Wells. Virtual Signing: Capture, Animation, Storage and Transmission - An Overview of the ViSiCAST Project. In *Proceedings of IEE Colloquium on Speech and Language Processing for Elderly and Disabled People*, pages 6/1–6/7, London, UK, 2000a. 63

J. Andrew Bangham, Stephen Cox, Michael Lincoln, Ian Marshall, Marcus Tutt, and Mark Wells. Signing for the Deaf using Virtual Humans. In *Proceedings of IEE Colloquium on Speech and Language Processing for Elderly and Disabled People*, pages 4/1–4/5, London, UK, 2000b. 38

Kenneth R. Beesley. Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans in 2001. In *Proceedings of the Workshop on Arabic Language Processing: Status and Prospects, ACL*, pages 31–34, Toulouse, France, 2001. 36

Kenneth R. Beesley and Lauri Karttunen. *Finite State Morphology.* CSLI Publications, Stanford, California, 2003. 35, 36

Brita Bergman and Johanna Mesch. ECHO data set for Swedish Sign Language (SSL). Technical report, Department of Linguistics, Stockholm University, Sweden, 2004. URL http://www.let.ru.nl/sign-lang/echo/. 52

Sami Boudelaa and William D Marslen-Wilson. Aralex: a lexical database for Modern Standard Arabic. *Behavior Research Methods*, 42(2):481–487, 2010. 13

Abdelhaleam Boudlal, Abdelhaq Lakhouaja, Azze-Eddine Mazroui, Abdelwafi Meziane, Mohammad Bebah, and Mostafa Shoul. Alkhalil Morpho. Sys: A Morphosyntactic analysis system for Arabic texts. In *Proceedings of the International Arab Conference on Information Technology, ACIT*, Al Kurah, Lebanon, 2012. 37

H. William Brelje. *Global Perspectives on the Education of the Deaf in Selected Countries*. Butte Publications, Hillsboro, Oregon, 1999. 2

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, , and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16(2): 80–85, 1990. 5, 29

Hennie Brugman and Albert Russel. Annotating multimedia/multi-modal resources with ELAN. In *Proceedings of the 4th International Conference on Language Resources and Evaluation, LREC*, pages 2065–2068, Lisbon, Portugal, 2004. 52, 71

Tim Buckwalter. *Buckwalter Arabic Morphological Analyzer Version 1.0*. Linguistic Data Consortium, LDC, Philadelphia, Pennsylvania, 2002. 34

Tim Buckwalter. Issues in Arabic Orthography and Morphology Analysis. In *Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages, CAASL*, pages 31–34, Geneva, Switzerland, 2004. 14, 15

Jan Bungeroth and Hermann Ney. Statistical Sign Language Translation. In *Proceedings of the 1st Workshop on Representation and Processing of Sign Languages: Corpora for Sign Language Technologies*, pages 105–108, Lisbon, Portugal, 2004. 75

Jan Bungeroth, Daniel Stein, Philippe Dreuw, Morteza Zahedi, and Hermann Ney. A German Sign Language Corpus of the Domain Weather Report. In *Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC*, pages 2000–2003, Genoa, Italy, 2006. 55

Jan Bungeroth, Daniel Stein, Philippe Dreuw, Hermann Ney, Sara Morrissey, Andy Way, and Lynette van Zijl. The ATIS Sign Language Corpus. In *Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC*, Marrakesh, Morocco, 2008. 56

Pavel Campr, Marek Hru'z, and Jana Trojanova'. Collection and Preprocessing of Czech Sign Language Corpus for Sign Language Recognition. In *Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC*, Marrakesh, Morocco, 2008. 56

Michael Carl and Silvia Hansen. Linking Translation Memories with Example-Based Machine Translation. In *Proceedings of Machine Translation Summit VII*, pages 617–624, Singapore, 1999. 74

Michael Carl and Andy Way. Introduction. In *Recent Advances in Example-based Machine Translation*, pages xvi–xxxi. Kluwer Academic Publishers, Dordrecht, Netherlands, 2003. 7, 98

Ze-Jing Chuang, Chung-Hsien Wu, and Wei-Sheng Chen. Movement Epenthesis Generation Using NURBS-Based Spatial Interpolation. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(11):1313–1323, 2006. 59, 62

Brona Collins. *Example-Based Machine Translation: An Adaptation-Guided Retrieval Approach*. PhD thesis, University of Dublin, Dublin, Ireland, 1998. 74

Stephen Cox, Michael Lincoln, Judy Tryggvason, Melanie Nakisa, Mark Wells, Marcus Tutt, and Sanja Abbott. TESSA, A System to Aid Communication with Deaf People. In *Proceedings of the 5th International ACM Conference on Assistive Technologies, ASSETS*, pages 205–212, Edinburgh, UK, 2002a. 47, 63

Stephen Cox, Ian Marshall, and Eva Sáfár. Approaches to English to Sign Translation. *The Linguist*, 41(1):6–10, 2002b. 40

Stephen Cox, Mike Lincoln, Judy Tryggvason, Melanie Nakisa, Mark Wells, Marcus Tutt, and Sanja Abbott. The Development and Evaluation of a Speech to Sign Translation System to Assist Transactions. *International Journal of Human-Computer Interaction*, 16(2):141–161, 2003. xi, 41, 42

Onno Crasborn and Inge Zwitserlood. The Corpus NGT: an online corpus for professionals and laymen. In *Proceedings of the 3rd Workshop on the Representation and Processing of Sign Languages, LREC*, pages 44–49, Marrakech, Marocco, 2008. 54

Onno Crasborn, Els van der Kooij, Annika Nonhebel, and Wim Emmerik. ECHO data set for Sign Language of the Netherlands (NGT). Technical report, Department of Linguistics, Radboud University Nijmegen, Nederlands, 2004. URL `http://www.let.ru.nl/sign-lang/echo/`. 52

Onno Crasborn, Inge Zwitserlood, and Johan Ros. The Corpus NGT: A digital open access corpus of movies and annotations of Sign Language of the Netherlands. Technical report, Centre for Language Studies, Radboud University Nijmegen, Nederlands, 2008. URL `http://www.ru.nl/corpusngtukgp/`. 54

Terry Crowley and Claire Bowern. *An introduction to historical linguistics*. Oxford University Press, Oxford, UK, 1977. 17

Walter Daelemans, Antal van den Bosch, and Jakub Zavrel. Forgetting exceptions is harmful in language learning. *Machine Learning*, 34(1–3):11–43, 1999. 29

Ralph Elliott, John Glauert, Richard Kennaway, Ian Marshall, and Eva Sáfár. Linguistic modelling and language-processing technologies for Avatar-based sign language presentation. *Universal Access in the Information Society*, 6 (4):375–391, 2008. 63, 65

Ralph Elliott, Javier Bueno, Richard Kennaway, and John Glauert. Towards the Integration of Synthetic SL Animation with Avatars into Corpus Annotation Tools. In *Proceedings of the 7th International Conference on Language Resources and Evaluation, LREC*, pages 84–87, Valetta, Malta, 2010. xii, 64

Moustafa Elshafei, Husni Al-Muhtaseb, and Mansour Alghamdi. Statistical Methods for Automatic diacritization of Arabic text. In *Proceedings 18th National computer Conference*, Riyadh, Saudi Arabia, 2006. 93

Victoria Fromkin, Robert Rodman, and Nina Hyams. *An Introduction to Language*. Wadsworth Cengage Learning, Boston, MA, 2009. 14

Osamu Furuse and Hitoshi Iida. An Example-Based Method for Transfer-Driven Machine Translation. In *Proceedings of the 3rd International Conference on Theoretical and Methodological Issues in Machine Translation, TMI*, pages 139–150, Baltimore, Maryland, 1992. 74

John Glauert, Ralph Elliott, Stephen Cox, Judy Tryggvason, and Mary Sheard. VANESSA - A System for Communication Between Deaf and Hearing People. *Technology and Disability*, 18(4):207–216, 2006. xii, 63, 65

Nano Gough and Andy Way. Robust large-scale EBMT with marker-based segmentation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation, TMI*, pages 95–104, Baltimore, Maryland, 2004. 29, 74

Thomas Green. The Necessity of Syntax Markers: Two Experiments with Artificial Languages. *Journal of Verbal Learning and Behavior*, 18:481–496, 1979. 45, 100

Nizar Y. Habash. *Introduction to Arabic Natural Language Processing*. Morgan & Claypool Publishers, San Rafael, California, 2010. 14

Jan Hajič, Otakar Smrž, Tim Buckwalter, and Hubert Jin. Feature-based Tagger of Approximations of Functional Arabic Morphology. In *Proceedings of the Workshop on Treebanks and Linguistic Theories, TLT*, pages 53–64, Barcelona, Spain, 2005. 35

Sami Halawani. Arabic Sign Language translation system on mobile devices. *IJCSNS International Journal of Computer Science and Network Security*, 8 (1):251–256, 2008. 47, 48

Ali Hanafe. A Survey of Deaf Teachers in Riyadh. In *Proceedings of the Workshop on Special Education in Saudi Arabia*, pages 1–36, Riyadh, Saudi Arabia, 2005. In Arabic. 17

Thomas Hanke. HamNoSys - representing sign language data in language resources and language processing contexts. In *Proceedings of the 1st Workshop on the Representation and Processing of Sign Languages, LREC*, pages 1–6, Paris, France, 2004. 49

Hieu Hoang and Philipp Koehn. Design of the Moses Decoder for Statistical Machine Translation. In *Proceedings of the Workshop on Software Engineering, Testing, and Quality Assurance for NLP, ACL*, pages 58–65, Columbus, Ohio, 2008. 45

Matt Huenerfauth. *Generating American Sign Language classifier predicates for English-to-ASL machine translation*. PhD thesis, University of Pennsylvania, Philadelphia, Pennsylvania, 2006. 38, 42

Matt Huenerfauth. Generating American Sign Language Animation: overcoming misconceptions and technical challenges. *Universal Access in the Information Society*, 6(4):419–434, 2008. 25

Sattar Izwaini. Problems of Arabic Machine Translation: Evaluation of three systems. In *Proceeding of the International Conference on the Challenge of Arabic for NLP/MT*, pages 118–148, London, UK, 2006. 31

Trevor Johnston and Adam Schembri. *Australian Sign Language: An Introduction to Sign Language Linguistics*. Cambridge University Press, Cambridge, UK, 2007. 21

Daniel Jones. *Analogical Natural Language Processing*. UCL Press, London, UK, 1996. 29

Jim Kakumasu. Urubu sign language. *the International Journal of American Linguistics, IJAL*, 34(4):275–281, 1968. 49

Hans Kamp. A theory of truth and semantic representation. In *Formal Methods in the Study of Language*, pages 277–322. Mathematisch Centrum, Amsterdam, Netherlands, 1981. 41

Mihoko Kato. A Study of Notation and Sign Writing Systems for the Deaf. *Journal of Intercultural Communication Studies, ICS*, 17(4):97–114, 2008. 48, 49

Shereen Khoja, Roger Garside, and Gerry Knowles. A tagset for the morphosyntactic tagging of Arabic. In *Proceedings of the Corpus Linguistics Conference, CL*, pages 341–350, Lancaster, UK, 2001. xvi, 101, 102

Michael Kipp. Anvil - A Generic Annotation Tool for Multimodal Dialogue. In *Proceedings of the 7th European Conference on Speech Communication and Technology, Eurospeech*, pages 1367–1370, Aalborg, Denmark, 2001. 71

Chunyu Kit, Haihua Pan, and Jonathan J. Webster. Example-Based Machine Translation: A New Paradigm. In *Translation and information technology*, pages 57–78. The Chinese University Press, Hong Kong, China, 2002. 87

Reinhard Klette and Piero Zamperoni. *Handbook of Image Processing Operators*. Wiley-Blackwell, Hoboken, New Jersey, 1996. 60

LDC. Linguistic data annotation specification: Assessment of fluency and adequacy in translations. Technical Report Revision No. 1.5, Linguistic Data Consortium, USA, 2005. 116

Lorraine Leeson, John Saeed, Deirdre Byrne-Dunne, Alison Macduff, and Cormac Leonard. Moving Heads and Moving Hands: Developing a Digital Corpus of Irish Sign Language. In *Proceedings of the 6th Annual Information Technology and Telecommunications Conference , IT&T*, Carlow, Ireland, 2006. 52

Vladimir I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966. 29, 95, 99, 107, 123, 124

Vincenzo Lombardo, Cristina Battaglino, Rossana Damiano, and Fabrizio Nunnari. An avatarbased interface for the Italian Sign Language. In *Proceedings of the 5th International Conference on Complex, Intelligent, and Software Intensive Systems, CISIS*, pages 589–594, Seoul, Korea, 2011. xii, 67, 68

Andrew J. Lotto and Lori L. Holt. The illusion of the phoneme. *Chicago Linguistic Society*, 35(1):191–204, 2000. 15

Xiaoyi Ma and Christopher Cieri. Corpus Support for Machine Translation at LDC. In *Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC*, pages 859–864, Genoa, Italy, 2006. 116

A. Dave Marshall and Ralph R. Martin. *Computer Vision, Models and Inspection*. World Scientific Publishing Co., Singapore, 1992. 60

Ian Marshall and Eva Sáfár. Grammar Development for Sign Language Avatar-Based Synthesis. In *Proceedings of the 11th International Conference on Human Computer Interaction, HCII*, Las Vegas, Nevada, 2005. Published in CD-ROM. xi, 40, 42

Elizabeth A. Martin. *Oxford Concise Medical Dictionary.* Oxford University Press, Oxford, UK, 2007.

Abdulrahman Mirza. Developing University E-Content Through Incentives Almost Overnight, the Case of King Saud University. In *Proceedings of the World Conference on Educational Multimedia, Hypermedia & Telecommunications, ED-MEDIA*, pages 414–426, Honolulu, Hawaii, 2009. 2

Kgatlhego Aretha Moemedi. Rendering an avatar from signwriting notation for sign language animation. Master's thesis, Department of Computer Science, University of the Western Cape, Cape Town, South Africa, 2010. xii, 67, 68

Mohamed Mohandes. Automatic translation of Arabic text to Arabic Sign Language. *International Journal on Artificial Intelligence and Machine Learning, ICGST*, 6(4):15–19, 2006. 47

Sara Morrissey. *Data-Driven Machine Translation for Sign Languages.* PhD thesis, Dublin City University, Dublin, Ireland, 2008. xvi, 43, 44, 47, 57, 69, 75, 100, 101, 125, 131

Sara Morrissey. Body at Work: Using Corpora in Sign Language Machine Translation. In *Proceedings of the 1st International Workshop on Sign Language Translation and Avatar Technology, SLTAT*, Berlin, Germany, 2011. 75

Sara Morrissey and Andy Way. Joining Hands: Developing A Sign Language Machine Translation System with and for the Deaf Community. In *Proceedings of the Conference and Workshop on Assistive Technologies for People with Vision and Hearing Impairments: Assistive Technology for All Ages, CVHI*, Granada, Spain, 2007. xvi, 44, 46, 75

Sara Morrissey, Harold Somers, Robert Smith, Shane Gilchrist, and Sandipan Dandapat. Building a sign language corpus for use in machine translation. In

Proceedings of the 4th Workshop on Representation and Processing of Sign Languages: Corpora for Sign Language Technologies, LREC, pages 17–23, Valletta, Malta, 2010. 72

Makoto Nagao. A framework of a mechanical translation between Japanese and English by analogy principle. In *Artificial and Human Intelligence*, pages 173–180. North Holland, Amsterdam, Netherlands, 1984. 5, 29

Saduf Naqvi. End-User Involvement in Assistive Technology Design for the Deaf Are Artificial Forms of Sign Language Meeting the Needs of the Target Audience? In *Proceedings of the Conference and Workshop on Assistive Technologies for People with Vision & Hearing Impairments, CVHI*, pages 227–232, Granada, Spain, 2007. 58

Terrance M. Nearey. The segment as a unit of speech perception. *Journal of Phonetics*, 18(1–3):347–373, 1990. 14

Carol Neidle. SignStream Annotation: Conventions used for the American Sign Language Linguistic Research Project. Technical Report No. 11, Report, Boston University, Boston, Massachusetts, 2002. 56, 71

Carol Neidle. SignStream Annotation: Addendum to Conventions used for the American Sign Language Linguistic Research Project. Technical Report No. 13, Report, Boston University, Boston, Massachusetts, 2007. 56

Rani Nelken and Stuart M. Shieber. Arabic Diacritization Using Weighted Finite-State Transducers. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages, ACL*, pages 79–86, Ann Arbor, Michigan, 2005. 93

Franz Josef Och and Hermann Ney. Improved Statistical Alignment Models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 440–447, Hong Kong, China, 2000. 87, 111

Franz Josef Och and Hermann Ney. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, 2003. 45, 99

Julian Odell, Dan Kershaw, Dave Ollason, Valtcho Valtchev, and David Whitehouse. *The HAPI Book: A description of the HTK Application Programming Interface*. Entropic Cambridge Research Laboratory, Cambridge, UK, 1997. 39

Sinja Ojala. Studies on Individuality in Speech and Sign. Technical Report No. 135, TUCS Dissertations, Turku Centre for Computer Science, University of Turku, Finland, 2011. 19

Achraf Othman and Mohamed Jemni. Statistical Sign Language Machine Translation: from English written text to American Sign Language Gloss. *International Journal of Computer Science Issues, IJCSI*, 8(3):65–73, 2011. 57, 69

Carol A. Padden. Sign language geography. In *Deaf Around the World: The Impact of Language*, pages 19–37. Oxford University Press, Oxford, UK, 2010. x, 17, 18

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 311–318, Stroudsburg, PA, 2002. 123

Les Piegl and Wayne Tiller. *The NURBS Book*. Springer, Berlin, Germany, 1997. 62

Carl Pollard and Ivan A. Sag. *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago, Illinois, 1994. 41

Siegmund Prillwitz, Regina Leven, Heiko Zienert, Thomas Hanke, and Jan Henning. HamNoSys Version 2.0: Hamburg Notation System for Sign Languages: An Introductory Guide. *International Studies on Sign Language and Communication of the Deaf*, 5(1), 1989. xi, 50

William A. Rugh. Education in Saudi Arabia: Choices and Constraints. *Middle East Policy*, 9(2):40–55, 2002. 3

Biswajit Sarkar, Kaushik Datta, C. D. Datta, Debranjan Sarkar, Shashanka J. Dutta, Indranil Das Roy, Amalesh Paul, Joshim Uddin Molla, and Anirban Paul. A Translator for Bangla Text to Sign Language. In *Proceedings of the*

IEEE Annual India Conference, INDCON, pages 1–4, Gandhinagar, India, 2009. 60

Majdi S. Sawalha. *Open-source Resources and Standards for Arabic Word Structure Analysis: Fine Grained Morphological Analysis of Arabic Text Corpora*. PhD thesis, The University of Leeds, Leeds, UK, 2011. 37

Adam Schembri. British Sign Language Corpus Project: Open access archives and the Observers Paradox. In *Proceedings of the 6th International Conference on Language Resources and Evaluation, LREC*, Marrakesh, Morocco, 2008. 53

Rubén San Segundo, Alicia Pérez, Daniel Ortiz, Luis Fernando D'Haro, M. Inés Torres, and Francisco Casacuberta. Evaluation of Alternatives on Speech to Sign Language Translation. In *Proceedings of the 8th Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 2529–2532, Antwerp, Belgium, 2007. 125

Nasredine Semmar, Faiza Elkateb-Gara, and Christian Fluhr. Using A Stemmer In A Natural Language Processing system To Treat Arabic For Cross-language Information Retrieval. In *Proceedings of the 5th Conference on Language Engineering, CLE*, pages 1–10, Cairo, Egypt, 2005. 33

Daniel Sleator and Davy Temperley. Parsing English with a Link Grammar. Technical report, Department of Computer Science, Carnegie Mellon University, 1991. URL `http://www.cs.cmu.edu/afs/cs.cmu.edu/project/link/pub/www/papers/ps/tr91-196.pdf`. 41

Yasuhiro Sobashima, Osamu Furuse, Susumu Akamine, Jun Kawai, and Hitoshi Iida. A bidirectional, Transfer-Driven Machine Translation system for spoken dialogues. In *Proceedings of the 15th Conference on Computational linguistics, COLING*, pages 64–68, Stroudsburg, Pennsylvania, 1994. 74

Franc Solina, Slavko Krapež, Aleš JakliČ, and Vito Komac. Multimedia Dictionary and Synthesis of Sign Language. In *Design and Management of Multimedia Information Systems*, pages 268–281. Idea Group Publishing, Hershey, Pennsylvania, 2001. xii, 59, 61

Daniel Stein, Philippe Dreuw, Hermann Ney, Sara Morrissey, and Andy Way. Hand in Hand: Automatic Sign Language to English Translation. In *Proceedings of The 11th Conference on Theoretical and Methodological Issues in Machine Translation, TMI*, Skövde, Sweden, 2007. 56, 69

William C. Stokoe. Sign Language Structure: An Outline of the Visual Communication Systems of the American Deaf. *Studies in linguistics: Occasional papers*, 1960. 19, 48

Nicolas Stroppa and Andy Way. MaTrEx: the DCU Machine Translation System for IWSLT 2006. In *Proceedings of the International Workshop on Spoken Language Translation, IWSLT*, pages 31–36, Kyoto, Japan, 2006. xi, 44, 45

Michael Studdert-Kennedy. Contemporary issues in experimental phonetics. In *Speech perception*, pages 243–293. Elsevier, Amsterdam, Netherlands, 1976. 14

K.-Y. Su and J.-S. Chang. Why corpus-based statistics-oriented machine translation. In *Proceedings of TMI-92, 4th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 249–262, Montreal, Canada, 1992. 5, 28

Eiichiro Sumita and Hitoshi Iida. Experiments and prospects of Example-Based Machine Translation. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, ACL*, pages 185–192, Berkeley, California, 1991. 28, 73

Valerie Sutton. *Lessons in SignWriting*. The Deaf Action Committee For SignWriting, DAC, La Jolla, California, 2002. xi, 49, 50, 67

Christoph Tillmann, Stephan Vogel, Hermann Ney, A. Zubiaga, and Hassan Sawaf. Accelerated DP-based Search for Statistical Translation. In *Proceedings of the 5th European Conference on Speech Communication and Technology, EUROSPEECH*, pages 2667–2670, Rhodes, Greece, 1997. 124

Ho TuBao. *Introduction to Knowledge Discovery and Data Mining*. National Center for Natural Science and Technology, Hanoi, Vietnam, 2000. 132

Lynette van Zijl. South African Sign Language Machine Translation System. In *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*, pages 233–234, Portland, Oregon, 2006. 42

Lynette van Zijl and Dean Barker. South African Sign Language Machine Translation System. In *Proceedings of the 2nd international conference on Computer graphics, virtual Reality, visualisation and interaction in Africa*, pages 49–52, Cape Town, South Africa, 2003. 43

Tony Veale and Andy Way. Gaijin: A bootstrapping approach to example-based machine translation. In *Proceedings of the Second Conference on Recent Advances in Natural Language Processing, RANLP*, pages 27–34, Tzigov Chark, Bulgaria, 1997. 43

Tony Veale, Alan Conway, and Bróna Collins. The Challenges of Cross-Modal Translation: English-to-Sign-Language Translation in the Zardoz System. *Machine Translation*, 13(1):81–106, 1998. 38

Kees Versteegh. *The Arabic Language*. Edinburgh University Press, Edinburgh, UK, 1997. 14

Andy Way and Nano Gough. wEBMT: Developing and validating an example-based machine translation system using the World Wide Web. *Computational Linguistics*, 29(3):421–457, 2003. 43

Andy Way and Nano Gough. Comparing example-based and statistical machine translation. *Natural Language Engineering*, 11(3):295–309, 2005. 43, 100

Bencie Woll, Rachel Sutton-Spence, and Dafydd Waters. ECHO data set for British Sign Language (BSL). Technical report, Department of Language and Communication Science, University College London, UK, 2004. URL http://www.let.ru.nl/sign-lang/echo/. 52

Alison Wray, Stephen Cox, Mike Lincoln, and Judy Tryggvason. A formulaic approach to translation at the post office: Reading the signs. *Language and Communication*, 24(1):59–75, 2004. 38

Steve Young, Gunnar Evermann, Thomas Hain, Dan Kershaw, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, Valtcho Valtchev, and Phil Woodland. *The HTK Book*. Entropic Cambridge Research Laboratory, Cambridge, UK, 1995. 39

Krňoul Zdeněk, Kanis Jakub, Campr Pavel, Železný Miloš, and Müller Luděk. Sign Speech Synthesis System. In *Proceedings of the 1st International Workshop on Sign Language Translation and Avatar Technology, SLTAT*, Berlin, Germany, 2011. xii, 65, 66

Liwei Zhao, Karin Kipper, William Schuler, Christian Vogler, Norman I. Badler, and Martha Stone Palmer. A Machine Translation System from English to American Sign Language. In *Proceedings of the 4th Conference of the Association for Machine Translation in the Americas, AMTA*, pages 54–67, Cuernavaca, Mexico, 2000. 41