The South African Sign Language Machine Translation Project

Lynette van Zijl
Department of Computer Science
Stellenbosch University
P/Bag X1, Matieland 7602, South Africa
lynette@cs.sun.ac.za

ABSTRACT
We describe the South African Sign Language Machine Translation project, and point out the role that the project is playing in the larger context of South African Sign Language and accessibility for the South African Deaf Community.

Categories and Subject Descriptors
I.2.7 [Artificial Intelligence]: Natural Language Processing—Machine Translation

Keywords
Machine Translation, Sign Language

1. INTRODUCTION
The South African Deaf community is marginalized in its access to information. The reasons for this marginalization include the scarcity of and excessive cost of hiring interpreters, the lack of published information about South African Sign Language (SASL), and the low literacy rate (especially in disadvantaged communities). The aim of the South African Sign Language Machine Translation (SASL-MT) project is to increase the access of the Deaf community to information by (a) developing tools to assist hearing students to acquire SASL, (b) providing free access to SASL linguistic data and (c) developing a machine translation system from English text to SASL for specific domains where the need is greatest, such as clinics, hospitals and police stations.

The SASL-MT project has three major components:

- linguistic component and support toolsets – no supporting electronic data on SASL was available when we started the project, and a large amount of effort was spent on data gathering and the building of rudimentary SASL word lists and phrase lists;
- machine translation component – we implemented a tree adjoining parser for English, with translation rules to SASL; and
- graphical output component – we implemented a re-usable graphical signing avatar.

In this paper, we give a broad overview of the design and current status of each of these three components.

Sign languages are natural languages with their own syntax and grammar, but are visual-spatial languages instead of spoken languages. This modality of sign language causes unique difficulties in the machine translation process. For example, the space where signs are executed relative to the body of the signer, is known as the signing space – in the sentence generation phase of the translation, this signing space has to be constructed, with objects correctly placed in the signing space. Note that in sign languages, when a person or object is initially mentioned, a position in the signing space is indicated, and for subsequent references to it, the position is then indicated. For example, in the sentences *Harry eats a chocolate. He likes chocolates.*, the person *Harry* will be assigned a position in the signing space, and the pronoun *He* has to be translated as the signer pointing to that position.

Another essential aspect of sign language is the so-called non-manual signs, which comprise facial expression and body posture. The non-manual signs form an intrinsic part of the grammar of sign languages. For example, a question which is not accompanied by a questioning facial expression, is simply meaningless. For the generation of non-manual signs we use a solution based on text-to-speech algorithms. This is, to our knowledge, a novel approach.

1.1 Linguistics and Toolsets
There is almost no published information available on SASL. In particular, when the SASL-MT project was initiated, there was no freely available electronic data that could be utilized in the project. The first stage of the project therefore involved data gathering. A complication to the data gathering phase was caused by the current and active debate on whether South African Sign Language is one language with various dialects, or whether the regional differences indicate different languages [1]. We settled our data gathering exercise in the Western Cape region of the country, in the
city of Cape Town, since the grammar variations among the large metropoles are small.

Our data gathering followed the guidelines by Neidle [10], and we specifically set up our data gathering environments to minimize code switching. Our video data was analyzed and transcribed by SASL interpreters, mostly by hand. We currently have an annotated word list of approximately 800 words. We also compiled a bilingual English-SASL phrase book with commonly used South African phrases.

In addition, we developed a set of web-based tools to allow for popular access to the data that we have gathered. These include words with associated videos, as well as phrases and associated videos. These proved to be in high demand, especially for (hearing) people who are learning SASL. To our knowledge, this is the only electronic source of freely available SASL data in the country.

1.2 Machine Translation Component

In the SASL-MT project, we wanted to minimize the development time spent on the analysis of English text. We therefore decided to re-use existing freely available data and linguistic tools. In addition, we wanted to get a parser implemented as quickly as possible. We hence followed the design of the TEAM project [12], where a synchronous tree adjoining grammar (STAG) parser was used to translate English into American Sign Language. For the SASL-MT project, we wrote an Early-type tree adjoining grammar (TAG) parser, and re-used the English grammar definitions of the TEAM project1.

Given the parse trees for the English text, we constructed SASL trees and rule-based transfer rules to SASL trees. Due to the lack of linguistic information for SASL, we constructed these trees and rules by hand from a prototype set of sentences. As the linguistic knowledge about SASL expands, more trees can be incorporated into the system. When the system is due for user testing by linguists and the Deaf community, we will also have an opportunity to refine and improve our initial set of rules. This will provide a unique linguistic opportunity for a detailed investigation of SASL, and will thus be an important spin-off of the project.

Some of the problems in the original TEAM project [8], include the placement of objects in the signing space and the generation of non-manual signs. We address these problems by flagging the output parse tree of the original English text. In particular, we run a standard pronoun resolution algorithm on the English text. Any pronouns in the current sentence is then associated with the appropriate object in the signing space data structure. Initial results were good, but highlighted the fact that the identification of co-referential noun phrases [11] would be essential for the correct placement of objects in the signing space. For example, in the sentences *Harry eats a chocolate. That boy likes his chocolates*, the person *Harry* must be assigned a position in the signing space, and the system must recognize that *That boy refers to Harry* and hence must refer to the same position in the signing space. We are currently considering practical ways to incorporate noun phrase co-resolution in the SASL-MT system, but this is by no means an easy extension.

For non-manual signs, we note that these may be considered analogous to intonation and stress in speech [3]. For example, a frowning expression occurs for a question in sign language, while a change in pitch occurs for questions in spoken languages. Prosody (that is, intonation, pitch and accent) analysis can be undertaken based either on syntax or on semantics. The systems based on semantical analysis are known as concept-to-speech systems [6], and it is this approach that we follow in the SASL-MT project. Again, we run our prosody analysis algorithms on the original input text, and flag the English parse tree with metadata. This metadata flagging takes place on two levels – the first is on individual words, and the second on phrases (that is, subtrees).

In order to flag the tree for expressiveness, we first determine the type of sentence (that is, statement, exclamation or question). This provides an indicator for the speed and range in the movement of the sign. Secondly, the emotional quality of the utterance can influence the expressiveness of the sign. Based on classes of standard human emotions (such as love, happiness, surprise, anxiety, or anger), we analyze adverbial information in the text to flag phrases with a given emotion.

Lastly, we analyze the input sentence for accent on specific words. The assumption is that new information receives more stress than old information [3, 6]. We keep history lists and equivalence lists of previous sentences, and words not found in these lists are flagged as accented words with stress. We also identify contrasts, and flag these.

Our experiments showed that a STAG-based parser can be implemented for a text to sign language machine translation system, and that additional processing of the parse tree eliminates problems with placement in the signing space and with non-manual sign generation. In addition, the flagging of the parse tree improved the quality of the data that is sent to the signing avatar.

1.3 The Signing Avatar Component

The signing avatar component of the SASL-MT project concerns three separate issues: the input notation to the avatar, the avatar itself, and the animations performed on the avatar.

To make the avatar as general as possible, we based our avatar on the H-Anim standard [5]. However, the H-Anim standard lacks enough features to model facial expressions in sign language, and we had to extend the avatar to include these features.

One of our major goals with the signing avatar was to develop a pluggable and re-usable signing avatar. Currently, most signing avatars are either commercial and not freely available, or are purpose-designed and fully integrated into other systems [4]. Such integrated avatars do not lend themselves to re-use, and we wanted a stand-alone generic signing avatar that can be used freely in other projects. To that end,
we developed a Java-based avatar, with animation capabilities defined relative to the size of the avatar. For example, in physiological terms, a shoulder joint should be able to move maximally one-third of the length of the upper arm. This means that any reasonable avatar can be coupled to our animation code.

Some sign language descriptive notation must activate the animations in the signing avatar, so that it can execute a given sign. We use the well-known XSTEP notation [7] for this purpose. As long as the output from any other system produces an XSTEP file, our animation code can let an H-Anim avatar execute the sign.

1.4 Future Work
The SASL-MT project is still under development, with specific aspects implemented as separate modules. We have a working: TAG parser for English, metadata generator for pronoun resolution and generation of emotional flags, and a signing avatar. Based on prototype linguistic SASL data, a given English sentence can be analysed, and a given parse tree translated to a SASL tree. Given an XSTEP description of a sign, the avatar can sign it correctly.

We are currently working on combining the separate modules into a single integrated project. This entails: automatic selection of the best parse tree from the selection generated by the TAG parser, sentence generation from the SASL parse tree, and an interface between the machine translation component and the signing avatar.

2. ACKNOWLEDGEMENTS
This research was supported by NRF grant #2053436.

3. REFERENCES


