Sign Language HPSG

Eva Safar, John Glauert
School of Computing Sciences
University of East Anglia
Norwich NR4 7TJ, UK
E-mail: E.Safar@uea.ac.uk, J.Glauert@uea.ac.uk

Abstract

We present an overview of some relevant aspects of sign language synthesis in the ViSiCAST project, which might serve as a possible basis for the Dicta-Sign project. Dicta-Sign is a 3-year EU-funded project, that undertakes parallel corpus collection in different Sign Languages (SLs) and fundamental research and development of sign recognition and generation techniques in order to open up new potential applications for sign language users. One of the aims in Dicta-Sign is to find a model that is suitable for both recognition and generation. In this paper we revisit the main aspects of the synthesis techniques implemented in ALE Prolog using a sign language specific HPSG with the view for future changes needed.

1. Introduction

We present an overview of some relevant aspects of sign language synthesis in the ViSiCAST project[1], which might serve as a possible basis for the Dicta-Sign project. Dicta-Sign[2] is a 3-year EU-funded project, that undertakes parallel corpus collection in different SLs and fundamental research and development of sign recognition and generation techniques in order to open up new potential applications for sign language users. Therefore the aim in Dicta-Sign is to find a model that is suitable for both recognition and generation.

In the ViSiCAST project we had sound reasons in favour of HPSG (Head-driven Phrase Structure Grammar) for sign language modelling. In sign languages variation in grammars is less than in lexicons, therefore a lexicalist approach is suitable for developing grammars for more than one target languages in parallel. Differences are encoded in the lexicon, while grammar rules are usually shared with occasional variation in semantic principles. A further consideration in favouring HPSG is that the feature structures can incorporate modality-specific aspects (non-manual features) of signs appropriately (Safar & Marshall, 2002).

2. Modifications to the ALE Implementation

Our HPSG is implemented in ALE Prolog (Shieber et al.,1989). The current ViSiCAST feature structure and grammar rules will have to be adapted in a way that they are suitable for analysis as well. Most changes can be expected in the phonetic and syntactic features. A list of thousand concepts has been collected for parallel corpora in German, French, British and Greek SLs, which initially serve the purpose of guiding the annotation of the collected corpus. This initial lexicon has to be refined to be used in SL grammars enhanced with the linguistic knowledge gained from the corpus for analysis and synthesis.

ALE has been modified to make it compatible with the more recent version of Prolog (SWI v.5.6) on PC and Mac. Picture 1 shows a typical lexical entry for a noun, which will be explained in more detail in the following sections. The left hand side (LHS) represents another modification to ALE. The LHS is a list of HamNoSys transcription symbols for manuals and non-manuals (Prillwitz et al., 1989) instead of a word. On the right hand side (RHS) the values of the phonetic (PHON) features are instantiated and propagated to the LHS (like accompanying 'Brow' in this example) via unification and principles. This way we created a dynamic lexicon without increasing compilation time.

3. Architecture in ViSiCAST

ViSiCAST produced a prototype English text to SL translation system. First the English written text was parsed. The output of the parser was then processed using λ-calculus, β-reduction and DRS merging (Blackburn & Bos, 1999). The result was a Discourse Representation Structure (DRS), which in a flattened form served as the input for the HPSG synthesis. In Dicta-Sign after reviving the old system (see Section 2) we can produce the HPSG output. The generated sequence is HamNoSys for manual features and codes for non-manual features. This linguistic analysis can be then linked with the animation technology by encoding the result in XML as SiGML which is then sent to the JASigning animation system (Elliott et al., 2010).

[1] It was an EU Framework V supported project, which developed virtual signing technology in order to provide information access and services to Deaf people.

[2] We acknowledge that this work is funded through the Dicta-Sign project under the European Union's 7th Framework Programme (grant 231135).

```
[[mug],[Brow],['mVg',hamfist,hamthumbacrossmod,hamextfingerol,hampalml,
                hamshoulders,hamclose,hamparbegin,hammoveu,hamarcu,hamsmallmod,hamreplace,
                hamextfingerul,hampalmdl,hamparend]] --->
    word,
           gloss:mug,
           phon:(face:brow:Brow,
                 mouth:pict:'mVg',
                 man:(ndh:hns_string,
                      hsh:[hamfist,hamthumbacrossmod],
                      ori:(efd:[hamextfingerol],
                           plm:[hampalml]),
                      mov:[(repeat:R,
                            mloc:Height,
                           m_hns:[hamparbegin,hammoveu,hamarcu,hamreplace,
                            hamextfingerul,hampalmdl,hamparend])],
                      const:[]),
                allow_weak_drop:minus),
           syn:(precomps:[],
               postcomps:[],
               head:(fix_or_noun, context:(context_in : Cin,
                                 context_out: Cin),
                    agr:(cl:(cl_manip:(@ upright_cylinder),
                                          cl_meas:(@ round_obj_meas),
                                          cl_subst:(@ upright_cylinder)),
                                      num:(number:Sg,collordist:Coll),
                                      per:per_three),
                                 form:normal),
               arg_st:[],
               allow_pl_sweep:no,
               allow_pl_repeat:no),
           sem:(index:(ref,Ind),
                count:(number:Sg,collordist:Coll),
                restr:[(sit:Ind3,
                        reln:mug,
                        sense:1,
                        args:[])]).
```

Picture 1: An example entry for a noun

4. HPSG Structure

The HPSG feature structure (see Picture 1) starts with the standard PHON (phonetic), SYN (syntactic) and SEM (semantic) components (Pollard & Sag, 1994).

The PHON component describes how the signs are formed by handshape, orientation, finger direction and movement. From the non-manuals the eye-brow movement and mouth-picture were implemented (PHON:FACE:BROW and PHON:MOUTH:PICT).

The SYN component determines the argument structure and conditions for unification. It contains information on what classifiers the word can take (the classifier features are associated with the complements (SYN:HEAD:AGR) and their values are propagated to the PHON structure of the verb in the unification process) or how pluralisation can be realised but also on mode, which is associated with sentence type and (pro)noun drop. The context feature is used to locate things in the three-dimensional signing space. The positions are used for referencing and for directional verbs, where such positions are obligatory morphemes. This feature is propagated through derivation. Movement of objects in signing space is achieved by associating an ADD_LIST and a DELETE_LIST with directional verbs (Safar & Marshall, 2002). Picture 2 shows an example of the HEAD feature of a verb.

The SEM structure includes semantic roles with WordNet definitions for sense to avoid eventual ambiguity in the English gloss.

5. Rules and Principles

The rules deal with sign order of (pre-/post-)modifiers (adjuncts) and (pre-/post-)complements. British Sign Language is a topic-comment language, where the complements can subcategorize for their own complements. Therefore we introduced a Last-Complement rule to finish the recursion of the pre- and postcomp rules. This means that we deviate from the standard Subject-Head rule or schema.

5.1 Mode

The principle of MODE propagates the eye-brow movement's value (neutral, furrowed, raised), which is associated with the sentence type in the input (declarative, yes-no question and wh-question) throughout.

5.2 Prodrop

The second type of principle deals with prodrop, which means the non-overt realization of the pronomina. We introduced an empty lexical entry. The principle checks the semantic head for the values of subject and object prodrop features. Picture 2 shows that the values can be *can* or *can't*, a third value is possible, which is *must*. We then extract the syntactic information for the empty lexical item, which has to be unified with the complement information of the verb. If the value is *can't* prodrop is not possible, in case of *can* we generate both solutions.

```
head:(dirmanipverb_lxm,
     agr:(num:(number:SSg,collordist:Coll),
          per:per,gref:Index2),
     aux:minus,
     context:(context_in : list_context,
          add_list   : [  (glossref:[(ref:Index2,glossr:Gloss)],
                                  locat:locatefd:Efd,
                                  distance:Distobj,
                                  heights:Heightobj),
                          (glossref:[(ref:Index1,glossr:Glosssubj)],
                                  locat:locatefd:Efdsubj,
                                  distance:Distsubj,
                                  heights:Heightsubj),
                            (glossref:[(ref:Index2,glossr:Gloss)],
                                  locat:locatefd:Efdsubj,
                                  distance:Distsubj,
                                  heights:Heightsubj)],
              context_out: list_context,
              delete_list:[(glossref:[(ref:Index2,glossr:Gloss)],
                                  locat:locatefd:Efd,
                                  distance:Distobj,
                                  heights:Heightobj)]),
     prodrp_subj:(first:can,
             second:cant,
             third:cant),
     prodrp_obj:(first:can,
             second:cant,
             third:cant)),   %no info on this ~put
arg_st:[(sem:(index:Index2,Precomp1)),(sem:(index:Index1,Precomp2))]),
```

Picture 2: The HEAD information of a verb

5.3 Plurals

The third type of principle controls the generation of plurals. We handle repeatable nouns and non-repeatable nouns with external quantifiers and plural verbs. The input contains the semantic information that is needed to generate plurals that is a result of the analysis of an English sentence. SLs sign distributive and collective meanings of plurals differently, so the semantic input has to carry that information. English in this respect is often underspecified, therefore in some cases we needed human intervention in the analysis stage. The lexical item determines whether it allows repetition or sweeping movement. Picture 1 shows the allow_pl_repeat and the allow_pl_sweep feature under SYN. (sweeping movement indicates the collective involvement of a whole group, while repetition has a distributive meaning) When the feature's value is *yes* in any case, then the MOV (movement) feature in PHON is instantiated to the appropriate HamNoSys symbol expressing repetition or sweep motion in agreement with the SEM:COUNT:COLLORDIST feature value. The verb pluralization is handled similarly. For more on plurals, its issues and relation to signing space we refer to (Marshall & Safar, 2005).

5.4 Signing Space

The fourth type of principle is the managing of signing space. In signing, being a visual language, we place objects in the 3D signing space. We not just place them in a certain position but we also move them around. These objects can be referred to via pointing or being signed in the same location (anaphoric relationships), but they can also be manipulated by directional verbs. Directional verbs need a starting and an end point for the movement, which can be obtained by propagating a map of sign space positions through derivation. The missing phonemes of those positions are available in the SYN:HEAD:CONTEXT feature. While generating the verbs arguments they are populated in different positions of the signing space. If the verb requires the movement of those objects, they will be deleted from the 'old' position and added to a new position. Picture 2 shows the CONTEXT feature with an add_list and a delete_list. These lists control the changes of the map. The CONTEXT_IN and CONTEXT_OUT features are the initial input and the changed output lists of the map. The map is threaded through the generation. The final CONTEXT_OUT will be the input for the next sentence.

6. How Parameterization works

We will show an example for a lexical entry that has uninstantiated values on the RHS in the PHON structure and therefore the LHS HamNoSys needs to be parameterized as well. (For more details see Marshall & Safar, 2004 and Marshall & Safar, 2005).

```
[[take],
 [Brow],
 ['teIk', Hsh, Efd, Plm, Heightobj, Distobj,
R1,  hamreplace,  Efd,  Plm,  Heightsubj,
Distsubj, R2]]        ---> RHS
```

This is a less frequent example entry when the LHS contains only the HamNoSys structure. The handshape (Hsh), the extended finger direction (Efd) and the palm orienteation (Plm) are resolved when the object complement is processed. As in Picture 1 the noun's SYN:HEAD:AGR:CL featrure contains information on the different classifier possibilities associated with that noun (@ stands for macro below). In the unification process this information is available for the verb and therefore its PHON features can be instantiated and propagated to the LHS:

```
syn:(precomps:
    [(@nmanip(Ph, Gloss,  Index2, Precomp1, Hsh,
    Efd, Plm, Sg)),
    (@np2(W,  Glosssubj,  Plm2,  EfdT,  Index1,
    Precomp2, Num, PLdistr))]
```

The complements are added to the allocation map (signing space). The allocation map is available for the verb as well which governs the allocation and deletion of places in the map (see SYN:HEAD:CONTEXT feature in Picture 2), therefore the locations for the start and end position can be instantiated in PHON and propagated to the LHS. `Heightobj` and `Distobj` stand for the location and the distance from that location for the starting point of the sign, which is the location for the object. `Heigthsubj` and `Distsubj` stand for the end point of the movement, which is the location of the subject in signing space.

The Brow value is associated with the sentence type in the input and is propagated throughout.

R1 is the placeholder for the sweeping motion of the plural collective reading. R2 stands for the repetition of the movement for a distribute meaning. The verb's SYN:HEAD:AGR:NUM:COLLORDIST feature is unified with the SEM:COUNT feature values. If the SYN:ALLOW_PL_SWEEP or the SYN:ALLOW_PL_REPEAT features permit R1 or R2 can be instantiated according to the semantics. If the semantic input contains singular, R1 and R2 remain uninstantiated and are ignored in the SiGML translation.

7. Conclusion

This approach, i.e. the synthesis within an HPSG framework in a style that allowed to appropriately parameterize the HamNoSys descriptions by inheriting information from other linguistic constructs, proved to be fruitful and could be further developed in the framework of Dicta-Sign.

The Dicta-Sign project undertakes parallel corpus collection and annotation in different SLs and fundamental research and development in a range of (sign recognition and generation) techniques. The lexicon and grammar design therefore have to provide formal representations for recognition, generation and annotation. A lexicon should code information dealing with phonology, semantics, grammar, usage, variation and translation equivalents (compare Johnston,1998). Our HPSG lexicon model in ViSiCAST described signs for intended production providing finer grained details of phonetics and grammar to be able to drive an avatar rather than details of semantics, variation or usage.

The aim in Dicta-Sign is to find a model that is suitable for both recognition and generation. Therefore we have to avoid any specification in the entries, which would restrict recognition, but be specific enough to guide the production. The ViSiCAST grammar was specifically constructed for sign synthesis, so ways to make this process reversible still have to be developed. Also the annotation or translation purposes in different SLs require more information on variations and exact semantic descriptions.

A list of thousand concepts has been collected for parallel corpora in German, French, British and Greek SLs, which initially serve the purpose of guiding the annotation of the collected corpus. This initial lexicon has to be refined to be used in SL grammars enhanced with the linguistic knowledge gained from the corpus analysis for the purposes explained above.

8. References

Blackburn, P., Bos, J. (1999). *Representation and Inference for Natural Language. A First Course in Computational Semantics.* Vol II. http://www.coli.uni-sb.de/~bos/comsem/book1.html

Elliott, R., Bueno, J., Kennaway, R., Glauert, J. (2010). Towards the Integration of Synthetic SL Animation with Avatars into Corpus Annotation Tools. In *Seventh International Conference on Language Resources and Evaluation, LREC 2010,* Malta (to be published)

Johnston, T. (1998). The lexical database of AUSLAN (Australian Sign Language). http://www.sign-lang.uni-hamburg.de/intersign/workshop1/johnston

Marshall, I., Safar, E. (2004). Sign Language Generation in an ALE HPSG. In Muller, S. (Ed.), *The Proceedings of the 11th International Conference on Head-Driven Phrase Structure Grammar.* Center for Computational Linguistics (HPSG-2004), Katholieke Universiteit Leuven, pp.189--201, ISSN 1535-1793.

Marshall, I., Safar, E. (2005). Grammar Development for Sign Language Avatar-Based Synthesis. In *3rd International Conference on UA in HCI,* vol. 8: Universal Access in HCI: Exploring New Dimensions of Diversity, Las Vegas, Nevada, USA, in HCII 2005 (CD-ROM).

Pollard, C., Sag, I. (1994). *Head-Driven Phrase Structure Grammar.* The University of Chicago Press, Chicago.

Prillwitz, S., Leven, R., Zienert, H., Hanke, T., Henning, J., et al.(1989). Hamburg Notation System for Sign Languages. An Introductory Guide. *International Studies on sign Language and the Communication of the Deaf.* Vol. 5., Institute of German Sign Language and Communication of the Deaf, University of Hamburg.

Safar, E., Marshall, I. (2002). Sign Language Translation via DRT and HPSG. In A. Gelbukh (Ed.), *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLing), Mexico.* Lecture Notes in Computer Science 2276, Springer Verlag, ISBN0302-9743, pp. 58—68.

Shieber, M., van Noord, G., Moore, C., Pereira, F.C.N.(1989). A Semantic-head-driven Generation Algorithm for Unification-based Formalisms. In *27th Annual Meeting of the Association for Computational Linguistics*. Vancouver, pp. 7—17.