Multimodal Grammar Implementation

Katya Alahverdzhieva
University of Edinburgh
10 Crichton Street
Edinburgh EH8 9AB
K.Alahverdzhieva@sms.ed.ac.uk

Dan Flickinger
Stanford University
Stanford, CA 94305-2150
danf@stanford.edu

Alex Lascarides
University of Edinburgh
10 Crichton Street
Edinburgh EH8 9AB
alex@inf.ed.ac.uk

Abstract

This paper reports on an implementation of a multimodal grammar of speech and co-speech gesture within the LKB/PET grammar engineering environment. The implementation extends the English Resource Grammar (ERG, Flickinger (2000)) with HPSG types and rules that capture the form of the linguistic signal, the form of the gestural signal and their relative timing to constrain the meaning of the multimodal action. The grammar yields a single parse tree that integrates the spoken and gestural modality thereby drawing on standard semantic composition techniques to derive the multimodal meaning representation. Using the current machinery, the main challenge for the grammar engineer is the non-linear input: the modalities can overlap temporally. We capture this by identical speech and gesture token edges. Further, the semantic contribution of gestures is encoded by lexical rules transforming a speech phrase into a multimodal entity of conjoined spoken and gestural semantics.

1 Introduction

Our aim is to regiment the form-meaning mapping of multimodal actions consisting of speech and co-speech gestures. The language of study is English, and the gestures of interest are *depicting*—the hand depicts the referent—and *deictic*—the hand points at the referent's spatial coordinates.

Motivation for encoding the form-meaning mapping in the *grammar* stems from the fact that *form* effects judgments of multimodal grammaticality: e.g., in (1)[1] the gesture performance along with the unaccented "called" in a single prosodic phrase seems ill-formed despite the gesture depicting an aspect of the referent—the act of calling.

(1) * Your MOTHER called ...
Hand lifts to the ear to imitate holding a receiver.

This intuitive judgment is in line with the empirical findings of Giorgolo and Verstraten (2008) who observed that prosody influences the perception of temporally misaligned speech-and-gesture signals as ill-formed. Further, Alahverdzhieva and Lascarides (2010) established empirically that the gesture performance can be predicted from the prosodic prominence in speech and that gestures not overlapping subject NPs cannot be semantically related with that subject NP. The fact that speech-and-gesture integration is informed by the form of the linguistic signal suggests formalising the integration within the grammar. Alternatively, integrating the gestural contribution by discourse update would involve pragmatic reasoning accessing information about linguistic form, disrupting the transition between syntax/semantics and pragmatics.

The work is set within HPSG — a constraint-based grammar framework with the different types and rules organised in a hierarchy. The semantic information, derived in parallel with syntax, is expressed in Minimal Recursion Semantics (MRS) which supports a high level of *underspecifiability* (Copestake et al., 2005). This is useful for computing gesture meaning since even through discourse processing not all semantic information resolves to a specific interpretation.

The rest of the paper is structured as follows: §2 provides theoretical background, §3 details the implementation and §4 discusses the evaluation.

[1] The speech item where the gesture is performed is marked by underlining, and the accented item is given in uppercase.

2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 582–586,
Montréal, Canada, June 3-8, 2012.

2 Background

2.1 Attachment Ambiguity

We view the integration of gesture and the synchronous, semantically related speech phrase as an attachment in a single parse tree constrained by the form of the speech signal—its prosodic prominence. With standard methods for semantic composition, we map this multimodal tree to an Underspecified Logical Form (ULF) which supports the possible interpretations of the speech and gesture in their context. The choices of attachment are not unique. Similarly to "John saw the man with the telescope", there is ambiguity as to which linguistic phrase a gesture is semantically related to, and hence likewise ambiguity as to which linguistic phrase it attaches to in syntax; e.g., in (2) the open vertical hand shape can denote a container containing books or a containee of books. This interpretation is supported by a gesture attachment to the N "books". A higher attachment to the root node of the tree supports another, metaphoric interpretation where the forward movement is the conduit metaphor of giving.

(2) I can give you other BOOKS ...
Hands are parallel with palms open vertical. They perform a short forward move to the frontal centre.

We address this ambiguity by grammar rules that allow for multiple attachments in the syntactic tree constrained by the prosodic prominence of the speech signal. The two basic rules are as follows:

1. **Prosodic Word Constraint.** Gesture can attach to a prosodically prominent spoken word if there is an overlap between the timing of the gesture and the timing of the speech word.
2. **Head-Argument Constraint**. Gesture can attach to a syntactic head partially or fully saturated with its arguments and/or modifiers if there is a temporal overlap between the syntactic constituent and the gesture.

Applied to (2), these rules would attach the gesture to "books" (a prosodically prominent item), also to "other books", "give you other books", "can give you other books" and even to "I can give you other books" (heads saturated with their arguments). However, nothing licenses attachments to "I" or "give". These distinct attachments would support the interpretations proposed above.

2.2 Representing Gesture Form and Meaning

It is now commonplace to represent gesture form with Typed Feature Structures (TFS) where each feature captures an aspect of the gesture's meaning; e.g., the gesture in (2) maps to the TFS in (3). Note that the TFS is typed as *depicting* so as to differentiate between, say, a hand shape of depicting gesture and a hand shape of deixis. This distinction effects the gestural interpretation: a depicting gesture provides non-spatial aspects of the referent's denotation, and so form bears resemblance to meaning. Conversely, deixis identifies the spatial coordinates of the referent in the physical space.

(3)
depicting	
HAND-SHAPE	open-flat
PALM-ORIENT	towards-centre
FINGER-ORIENT	away-body
HAND-LOCATION	centre-low
HAND-MOVEMENT	away-body-straight

Each feature introduces an underspecified elementary predication (EP) into LF; e.g., the hand shape introduces $l_1 : hand\_shape\_open\_flat(i_1)$ where l_1 is a unique label that underspecifies the scope of the EP relative to other EPs in the gesture's LF, i_1 is a unique metavariable that underspecifies the main argument' sort (e.g., in (2) it can resolve to an individual if the gesture denotes the books or an event if it denotes the giving act) and $hand\_shape\_open\_flat$ underspecifies reference to a property that the entity i_1 has and that can be depicted through the gesture's open flat hand shape.

In the grammar, we introduce underspecified semantic relations *vis_rel(s,g)* between speech *s* and depicting gesture *g*, and *deictic_rel(s,d)* between speech *s* and deixis *d*. The resolution of these underspecified predicates is a matter of commonsense reasoning (Lascarides and Stone, 2009) and it therefore lies outside the scope of the grammar.

3 Implementation

The grammar was implemented in the LKB grammar engineering platform (Copestake, 2002) which was designed for TFS grammars such as HPSG. Since the LKB parser accepts as input linearly ordered strings and we represent gesture form with TFSs, we used the PET engine (Callmeier, 2000) which allows for injecting an arbitrary XML-based FS into

the input tokens. The input to our grammar is a lattice of FSs where the spoken tokens are augmented with prosodic information and the gesture tokens are feature-value pairs such as (3).

The main challenge for the multimodal grammar implementation stems from the non-linear multimodal input. The HPSG-based parsing platforms—LKB, PET and TRALE—can parse linearly ordered strings, and so they do not handle multimodal signals whose input comes from separate channels connected through temporal relations. Also, these parsing platforms do not support quantitative comparison operations over the time stamps of the input tokens. This is essential for our grammar since the multimodal integration is constrained by temporal overlap between speech and gesture (recall §2.1).

To solve this, we pre-processed the XML-based FS input so that overlapping TIME_START and TIME_END values were "translated" into identical start and end edges of the speech token and the gesture token as follows:

```
<edge source="v0" target="v1">
        <fs type="speech_token">
<edge source="v0" target="v1">
        <fs type="gesture_token">
```

This robust pre-processing step is sufficient since the only temporal relation required by the grammar is *overlap*, an abstraction over more fined-grained relations between speech (S) and gesture (G) such as (*precedence(start(S), start(G)) ∧ identity (end(S), end(G))*).

The linking of gesture to its temporally overlapping speech segment happens prior to parsing via chart-mapping rules (Adolphs et al., 2008) which involve re-writing chart items into FSs. The `gesture-unary-rule` (see Fig.1) rewrites an input (I) speech token in the context (C) of a gesture token into a combined speech+gesture token where the +GEST and +PROS values of the speech and gesture tokens are copied onto the output (O).

```
gesture-unary-rule := cm_rule &
 [+CONTEXT <gesture_token & [+GEST #gest]>,
  +INPUT <speech_token & [+PROS #pros]>,
  +OUTPUT <speech+gesture_token &
        [+GEST #gest, +PROS #pros]>,
  +POSITION "O1@I1, I1@C1" ].
```

Figure 1: Definition of `gesture-unary-rule`

The +PROS attribute contains prosodic information and the +GEST attribute is a feature-structure representation as shown in (3). The +POSITION constraint restricts the position of the I, O and C items to an overlap (`@`), i.e., the edge markers of the gesture token should be identical to those of the speech token, and also identical to the speech+gesture token. This chart-mapping rule recognises the gesture token overlapping the speech token and it records this by "augmenting" the speech token with the gesture feature-values.

In the grammar, we extended the ERG word and phrase rules with prosodic and gestural information where the +PROS and +GEST features of the input token are identified with the PROS and GEST of the word and/or lexical phrase in the grammar. We then added a lexical rule (see Fig. 2) which projects a gesture daughter to a complex gesture-marked entity of a single argument for which both the PROS and GEST features are appropriate.

```
gesture_lexrule := phrase_or_lexrule &
 [ ORTH [ PROS #pros ],
   ARGS <[ ORTH [ GEST gesture-form,
                  PROS p-word & #pros ]]>].
```

Figure 2: Definition of `gesture_lexrule`

This rule constrains PROS to a prosodically prominent word of type *p-word* thereby preventing a gesture from plugging into a prosodically unmarked word. The *gesture-form* value is a supertype over the distinct gesture types—depicting and deictic. The `gesture_lexrule` is inherited by a lexical rule specific to depicting gestures, and by a lexical rule specific to deictic gestures. In this way, we can encode the semantic contribution of depicting gestures which is different from the semantic contribution of deixis. For the sake of space, Fig. 3 presents only the `depicting_lexrule`. The semantic information contributed by the rule is encoded within C-CONT.

Following §2.2, the rule introduces an underspecified *vis_rel* between the main label `#dltop` of the spoken sign (via the HCONS constraints) and the main label `#glbl` of the gesture semantics (via the HCONS constraints). Note that these two arguments are in a *geq* (greater or equal) constraint. This means that *vis_rel* can operate over any projection of the speech word; e.g., attaching the gesture to "book" in (2) means that the relation is not restricted to the EPs contributed by "books" but it can be also over the EPs of a higher projection. The gesture's semantics is a bag of EPs (see §2.2), all of which are outscoped

'gesture/12-04-02/pet' Coverage Profile							
Aggregate	**total items** ♯	**positive items** ♯	**word string** ϕ	**lexical items** ϕ	**distinct analyses** ϕ	**total results** ♯	**overall coverage** %
$90 \leq$ *i-length* < 95	126	92	93.00	26.46	1.67	92	100.0
$70 \leq$ *i-length* < 75	78	54	71.00	12.00	1.00	54	100.0
$60 \leq$ *i-length* < 65	249	179	60.00	9.42	1.00	179	100.0
$45 \leq$ *i-length* < 50	18	14	49.00	7.00	1.00	14	100.0
Total	**471**	**339**	**70.25**	**14.35**	**1.18**	**339**	**100.0**

Table 1: Coverage Profile of Test Items generated by [incr tsdb()]

```
depicting_lexrule := gesture_lexrule &
[ARGS <[ SYNSEM.LOCAL.CONT.HOOK.LTOP
                                #dltop,
          ORTH [ GEST depicting] >,
 C-CONT [ RELS <![ PRED vis_rel,
                   S-ARG #arg1,
                   G-ARG #arg2 ],
                 [ PRED G_mod,
                   LBL #glbl,
                   ARG1 #harg ],
                 [ LBL #larg1 ],...!>,
          HCONS <!geq&[ HARG #arg1,
                        LARG #dltop ],
                  qeq&[ HARG #arg2,
                        LARG #glbl ],
                  qeq&[ HARG #harg,
                        LARG #larg1 ],
                  ...!>]].
```

Figure 3: Definition of `depicting_lexrule`

by the gestural modality [$\mathcal{G}$]. The rule therefore introduces in RELS a label (here `#larg1`) for an EP which is in *qeq* constraints with [$\mathcal{G}$]. The instantiation of the particular EPs comes from the gestural lexical entry. In the real implementation, the number of these labels corresponds to the number of features. They are designed in the same way and we thus forego any details about the rest.

4 Evaluation

The evaluation was performed against a test suite designed in analogy to the traditional phenomenon-based test-suites (Lehmann et al., 1996): manually-crafted to ensure coverage of well-formed and ill-formed data, but inspired by an examination of natural data. We systematically tested syntactic phenomena (intransitivity, transitivity, complex NPs, coordination, negation and modification) over well-formed and ill-formed examples where the ill-formed items were derived by means of the following operations: prosodic permutation (varying the prosodic markedness, e.g., from (4a) we derive (4b) to reflect intuitions of native speakers); gesture variation (testing distinct gesture types) and temporal permutation (moving the gestural performance over the distinct speech items).

(4) a. ANNA ate ...
Depicting gesture along with "Anna".

b. *anna ATE ...
Depicting gesture along with "Anna".

The test set contained 471 multimodal items (72% well-formed) covering the full range of prosodic (prosodic markedness and unmarkedness) and gesture (the span of depicting/deictic gesture and its temporal relation to the prosodically marked elements) permutations. The gestural vocabulary was limited since a larger gesture lexicon has no effects on the performance. To test the grammar, we used the [incr tsdb()][2] competence and performance tool which enables batch processing of test items and which creates a coverage profile of the test set (see Table 1). The values are as follows: the left column separates the items per aggregation criterion (the length of test items); the next column shows the number of test items per aggregate; then we have the number of grammatical items; average length of test item; average number of lexical items; average number of distinct analyses and total coverage.

5 Conclusions and Future Work

This paper reported on an implementation of a multimodal grammar combining spoken and gestural input. The main challenge for the current parsing platforms was the non-linear input which we solved by extending the spoken sign with the synchronous gestural sign semantics where synchrony was established by means of identical token edges. In the future, we shall extend the lexical coverage so that the grammar can handle various gestures and we also intend to evaluate the grammar with naturally occurring examples in XML format.

[2] `http://www.delph-in.net/itsdb/`

References

Peter Adolphs, Stephan Oepen, Ulrich Callmeier, Berthold Crysmann, Daniel Flickinger, and Bernd Kiefer. 2008. Some fine points of hybrid natural language parsing. In *Proceedings of the Sixth International Language Resources and Evaluation*. ELRA.

Katya Alahverdzhieva and Alex Lascarides. 2010. Analysing speech and co-speech gesture in constraint-based grammars. In Stefan Müller, editor, *The Proceedings of the 17th International Conference on Head-Driven Phrase Structure Grammar*, pages 6–26, Stanford. CSLI Publications.

Ulrich Callmeier. 2000. PET — A platform for experimentation with efficient HPSG processing techniques. *Natural Language Engineering*, 6 (1) (Special Issue on Efficient Processing with HPSG):99 – 108.

Ann Copestake, Dan Flickinger, Ivan Sag, and Carl Pollard. 2005. Minimal recursion semantics: An introduction. *Journal of Research on Language and Computation*, 3(2–3):281–332.

Ann Copestake. 2002. *Implementing Typed Feature Structure Grammars*. CSLI Publications, Stanford, CA.

Dan Flickinger. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*.

Gianluca Giorgolo and Frans Verstraten. 2008. Perception of speech-and-gesture integration. In *Proceedings of the International Conference on Auditory-Visual Speech Processing 2008*, pages 31–36.

Alex Lascarides and Matthew Stone. 2009. A formal semantic analysis of gesture. *Journal of Semantics*.

Sabine Lehmann, Stephan Oepen, Sylvie Regnier-Prost, Klaus Netter, Veronika Lux, Judith Klein, Kirsten Falkedal, Frederik Fouvry, Dominique Estival, Eva Dauphin, Herve Compagnion, Judith Baur, Lorna Balkan, and Doug Arnold. 1996. Tsnlp - test suites for natural language processing. In *COLING*, pages 711–716.