1   Background

We present the current state of HEGRAM, a deep linguistic processing grammar of Modern Hebrew. HEGRAM is implemented in the Linguistic Knowledge Builder (LKB) system and grounded in the theoretical framework of Head-driven Phrase Structure Grammar (HPSG).

2   HPSG

Head-Driven Phrase Structure Grammar (HPSG; **?**) is a constraint-based grammatical theory. In HPSG all linguistic objects (i.e., words, phrases, and clauses) are represented as typed feature structures. The basic mechanism by which linguistic objects are related to each other is structure-sharing. Structure-sharing occurs when two paths in a feature structure lead to the very same (token-identical) node. As a result, the information content associated with that node is the unification of the information provided by the various shared paths. A linguistic expression is said to be grammatical when the information contributed by components of the linguistic object is compatible and can accumulate to form a complete description of the expression.

HPSG has logical and mathematical foundations which make it amenable to computational implementation. Moreover, the declarative nature of HPSG grammars makes them non-directional and thus suitable for both parsing and generation.

3   About HEGRAM

HEGRAM is derived from the LinGO Grammar Matrix, which is an open-source starter-kit for the development of broad-coverage, precision HPSG grammars for diverse languages (**?**). The Matrix provides a skeleton of a grammar, which covers basic lexical and phrasal types, semantic composition, and the infrastructure for unbounded dependencies and coordination. The current state of the grammar is significantly different from the Matrix-derived one.

We have refined and extended the core grammar with the goal of achieving a broad-coverage grammar. One notable modification is the adoption of the packed argument-frame approach proposed by **?** to account for multiple argument frames per verb. In addition we extended the coverage to account for the Hebrew copular construction, including the possibility of zero copula, as well as other language-specific features such as noun-adjective agreement, and accusative case marking. At its current state, HEGRAM contains a small lexicon of === something about the lexicon and how we intend to interface with a large-scale lexicon

At this point our grammar covers "canonical" clauses with SVO word order, different complement types, verbs with multiple argument frames, long distance dependencies (wh-questions and non-subject topicalization), non-verbal predicates (aka "nominal clauses") including zero copula constructions, and control verbs. In what follows we will focus on control verbs to illustrate key features of our analysis.

4   Control

Control verbs take infinitival VPs with unexpressed subjects as complements. In subject control (1a), the unexpressed subject of the VP complement is identified with the subject of the control verb. With object control verbs (1b), the unexpressed subject of the VP is the object of the control verb.

(1)   a. dni   hbvix   l-h-ildh   ltt   l-hklb awkl
          Danny promised to-the-girl to.give ACC   the-dog food
          'Danny promised the girl to give the dog food.'

      b. dni   hreh   l-h-ildh   ltt   l-hklb awkl
          Danny allowed   to-the-girl to.give ACC   the-dog food
          'Danny allowed the girl to give the dog food.'

Control is a phenomenon which poses challenges to a computational analysis for two reasons. First, one syntactic argument in a sentence assumes two semantic roles. For example, in (1a) the subject, Danny, is both the 'promiser' and the 'giver'. In (1b) the girl is both the 'allowed' and the 'giver'. In addition, the controller is not necessarily adjacent to the verb of which it is the semantic subject.

=== something about how only a deep analysis can identify and represent these non-trivial relationships between arguments ====

5   Proof of Concept

As proof of concept, in what follows we present the HEGRAM analysis of an example sentence (2), which involves object control and a wh-question. In addition to providing the correct semantic linking between the matrix object and the unexpressed VPinf subject, the parser is required to recognize the syntactic and semantic role of the 'displaced' wh-element.

(2)   mh   dni   hreh   l-h-ildh   ltt   l-hklb
      what   Danny allowed to-the-girl to.give ACC   the-dog
      'What did Danny allowed the girl to give the dog?'

The analysis produced by the LKB includes a syntactic phrase structure tree (3) and a semantic representation(4). The syntactic tree represents the constituent structure assumed for the sentence.

(3)

```
                              S
                    ┌─────────┴─────────┐
                   NP                   S'
                    │           ┌────────┴────────┐
                    N          NP                 VP
                    │           │          ┌──────┴──────┐
                   mh           N          V'            VP
                                │      ┌────┴───┐    ┌────┴────┐
                               dni     V       PP   V'        PP
                                       │     ┌──┴─┐  │      ┌──┴──┐
                                      hreh   P   NP  V      P    NP
                                             │    │  │      │     │
                                             l    N ltt     l     N
                                                  │               │
                                               hildh            hklb
```
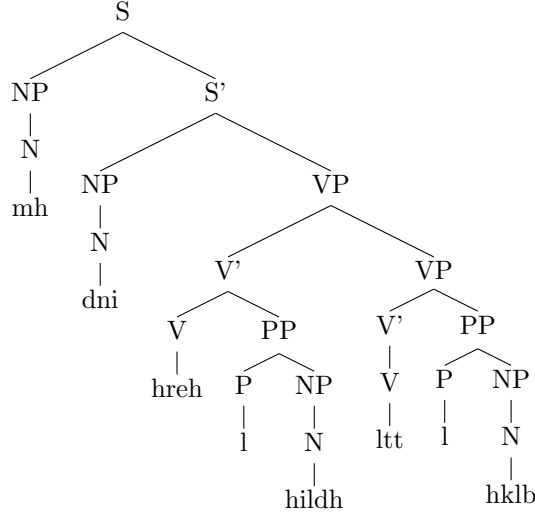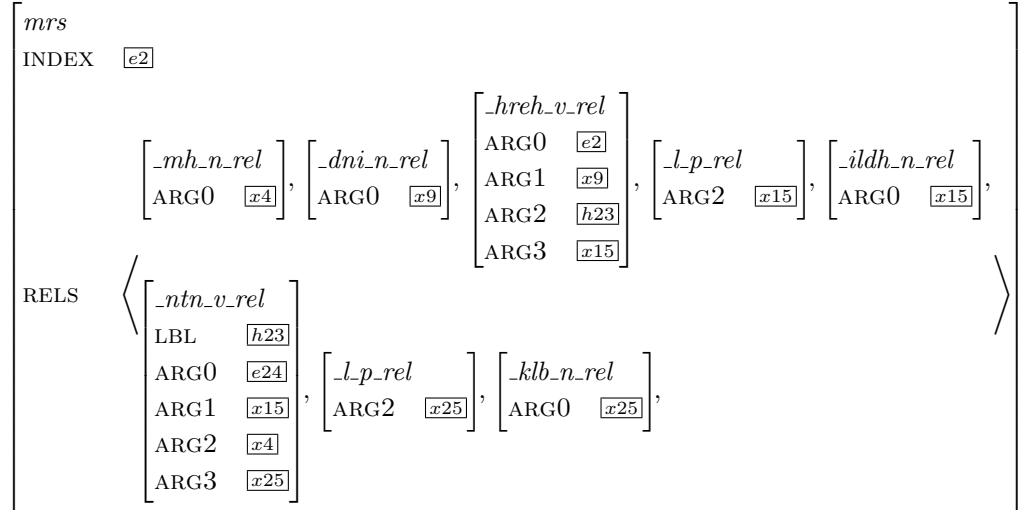
The semantic approach adopted by the LKB is Minimal Recursion Semantics (MRS), which assigns a syntactically flat semantic representation of linguistic expressions. The semantic representation mainly consists of a list of semantic relations and constraints on possible scope relations among them. Structure-sharing is expressed by way of co-indexation of arguments. Thus, for example, _hreh_v_rel, the semantic relation denoted by the matrix verb *hreh* ('allow') has 4 arguments: ARG0 is the event variable, which is co-indexed with the INDEX feature of the sentence, thus rendering it the main relation. ARG1, the 'agent' is co-indexed with the ARG0 of the semantic relation denoted by the subject, Danny. ARG2 is co-indexed with the label of the relation denoted by the VP complement, _ntn_v_rel, and ARG3 is co-indexed with ARG0 of the relation denoted by the object (_ildh_n_rel). Furthermore, the INDEX feature referring to 'Danny' is also the ARG1 of the relation denoted by the VP complement. === this is too long, and I'm not even done yet ===

$$(4) \begin{bmatrix} mrs \\ \text{INDEX} \quad \boxed{e2} \\ \\ \text{RELS} \left\langle \begin{array}{l} \begin{bmatrix} \_mh\_n\_rel \\ \text{ARG0} \quad \boxed{x4} \end{bmatrix}, \begin{bmatrix} \_dni\_n\_rel \\ \text{ARG0} \quad \boxed{x9} \end{bmatrix}, \begin{bmatrix} \_hreh\_v\_rel \\ \text{ARG0} \quad \boxed{e2} \\ \text{ARG1} \quad \boxed{x9} \\ \text{ARG2} \quad \boxed{h23} \\ \text{ARG3} \quad \boxed{x15} \end{bmatrix}, \begin{bmatrix} \_l\_p\_rel \\ \text{ARG2} \quad \boxed{x15} \end{bmatrix}, \begin{bmatrix} \_ildh\_n\_rel \\ \text{ARG0} \quad \boxed{x15} \end{bmatrix}, \\ \\ \begin{bmatrix} \_ntn\_v\_rel \\ \text{LBL} \quad \boxed{h23} \\ \text{ARG0} \quad \boxed{e24} \\ \text{ARG1} \quad \boxed{x15} \\ \text{ARG2} \quad \boxed{x4} \\ \text{ARG3} \quad \boxed{x25} \end{bmatrix}, \begin{bmatrix} \_l\_p\_rel \\ \text{ARG2} \quad \boxed{x25} \end{bmatrix}, \begin{bmatrix} \_klb\_n\_rel \\ \text{ARG0} \quad \boxed{x25} \end{bmatrix}, \end{array} \right\rangle \end{bmatrix}$$

References

Emily M. Bender, Daniel P. Flickinger, and Stephan Oepen. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In John Carroll, Nelleke Oostdijk, and Richard Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei, Taiwan, 2002.

Petter Haugereid. A grammar design accommodating packed argument frame information on verbs. In *PACLIC*, pages 31–40, 2011.

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