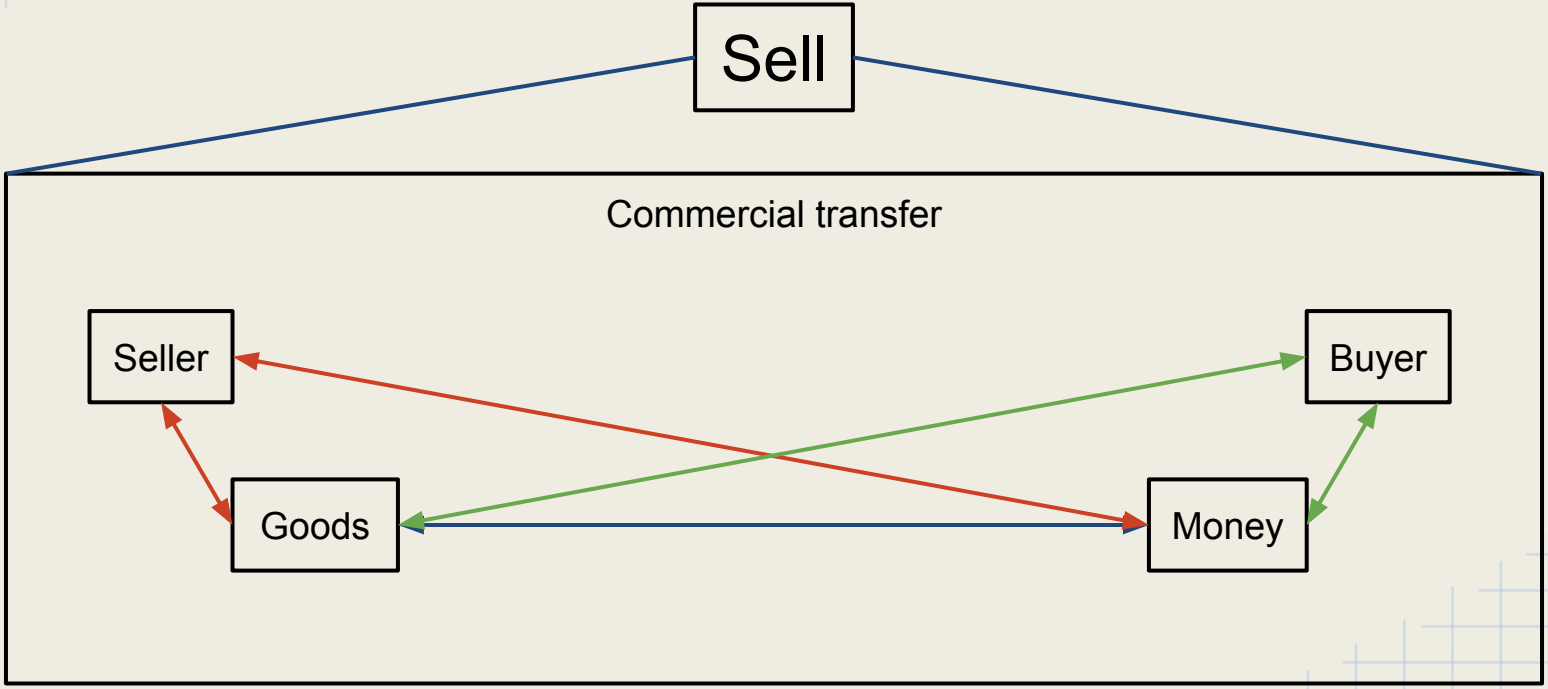
Automatic SRL in Hebrew and English

Part I

Introduction

- SRL / Semantic Parsing:
  - Detect the predicates in a given sentence
  - For each one
    - Detect its semantic arguments
    - Classify each argument into its specific role

Frame semantics (example)



Step I - Reproduce results

- The initial FrameNet article
  - Automatic labeling of semantic roles
    by: Daniel Gildea, Daniel Jurafsky
    Computational Linguistics, Vol. 28, No. 3. (2002), pp. 245-288

  - Task 1: Given a sentence, the predicate (lexical unit), a semantic frame and the boundaries of a semantic argument (frame element), find its role.
  - Task 2: Given a sentence, the predicate (lexical unit), and a semantic frame, find the semantic arguments

Step I - Reproduce results

- Collect dataset
  - FrameNet v1.5
  - Use sentences hand-annotated for a single target word
  - Only use target words with more than 10 annotated sentences

  - 137,903 sentences
  - 296,080 annotations of frame elements

  - 80%-10%-10% split for training-development-testing (per LU)

Features

- Most of the features are extracted from the constituency tree
- Use State-of-the-art CoreNLP constituent parser (v3.3.0)
- Parser to FE matching:
  - FEs contained in constituents: 23,111 (**7.81%**)
  - FEs spanning multiple neighbouring constituents: 15,582 (**5.26%**)
  - Perfect match between constituent and FE: 257,357 (**86.92**%)
  - No match (possible mutual overlap): 30 (**0.01%**)

Features

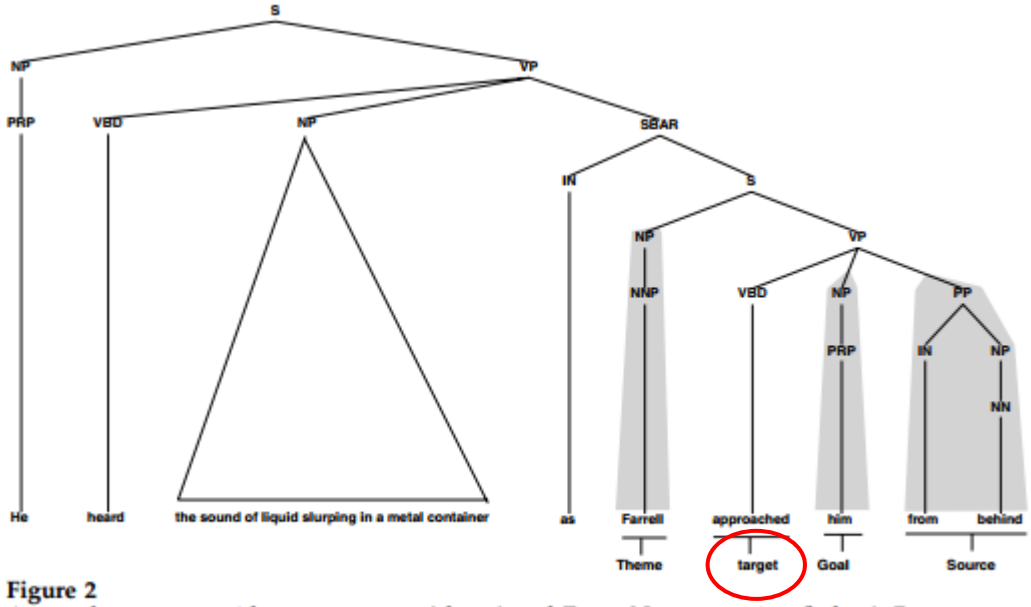


Figure 2
A sample sentence with parser output (above) and FrameNet annotation (below). Parse constituents corresponding to frame elements are highlighted.

Features

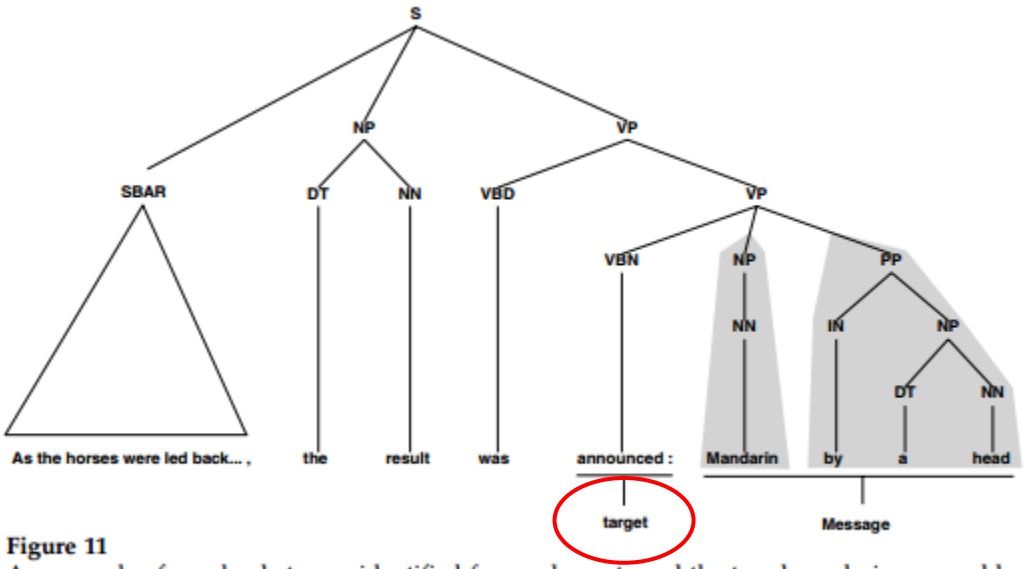


Figure 11
An example of overlap between identified frame elements and the true boundaries, caused by parser error. In this case two frame elements identified by the classifier (shaded subtrees) are entirely within the human annotation (indicated below the sentence), contributing two instances to row 2 of Table 9.

Features - task 1

- **Phrase type** - indicates the syntactic category of the phrase expressing the semantic roles
  - Uses the set of syntactic categories of the Penn Treebank project.

- **Governing category** - the type of the closest ancestor of the constituent corresponding to a frame element, which is either S or VP (can only have these 2 values). S->...->NP usually indicates NP is subject, VP->...->NP usually indicates objects.
  - This feature applies only to NPs, as it found to have little effect on the results when applied to other phrase types

Features - task 1

- **Parse tree path** - the directed path from the predicate to the constituent in question through the parse tree.

- **Position** - indicates whether the constituent to be labeled occurs before or after the predicate defining the semantic frame.

- **Voice** - label verbs as either passive or active.
  - Done by creating a set of 10 passive-identifying patterns, each requiring both a passive auxiliary (**to be**, **to get**, etc.) and a past participle.

Features - task 1

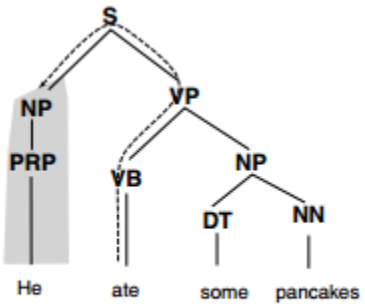


Figure 3
In this example, the path from the target word *ate* to the frame element *He* can be represented as VB↑VP↑S↓NP, with ↑ indicating upward movement in the parse tree and ↓ downward movement. The NP corresponding to *He* is found as described in Section 4.1.1.

Features - task 1

- **Head word** - the head word of the constituent to be labeled. generated as part of the standard analysis of the parser. no attempt to resolve pronoun references.

- **Target identity** - the predicate itself

Features - task 2

- **Parse tree path**
- **Head word**
- **Target identity**

Next steps

- Select a few learning methods (e.g., SVM, CRF, ILP, etc.)
- Train using the features and evaluate performance
- Compare different methods
  - Also compare to basic statistical method described in the article as a baseline