Creating Disjunctive Logical Forms from Aligned Sentences for Grammar-Based Paraphrase Generation

Scott Martin and Michael White

Department of Linguistics
The Ohio State University
{scott,mwhite}@ling.ohio-state.edu

Workshop on Monolingual Text-to-text Translation
Portland, Oregon
June 24, 2011

Idea

- Use packed semantic dependency graphs — called **disjunctive logical forms (DLFs)** — based on aligned paraphrases to increase generation quality.
- We exploit **OpenCCG**'s ability to generate from DLFs in addition to its ability to parse and assign semantic representations.
- This approach merges two traditions in paraphrasing:

  Grammar-based surface realization in the NLG tradition;

  Data-driven aligning existing paraphrases in order to generate new ones.
- Eventual **goal**: **improve automatic evaluation** of machine translation outputs by generating additional high-quality reference sentences.

Motivation

- Using semantic representations is particularly appealing because they are more abstract and less tied to surface word order than syntactic parses, as in Pang et al. (2003).
- Purely data-driven approaches have some success at similar tasks (Barzilay & Lee, 2003; Kauchak & Barzilay, 2006; Zhao et al., 2009; Madnani, 2010).
- But a grammar-based approach can potentially go further:
  - **Word-order variation**, e.g. modifier placement;
  - **Function word use** based on grammatical principles.

Example Data-driven Approach: Zhao et al. (2009)

Source	Liu Lefei says that [in the long *term*] , in terms of *asset* allocation, overseas investment should occupy a certain *proportion* of [an insurance company's overall allocation] .
Reference	Liu Lefei said that in terms of *capital* allocation , outbound investment should make up a certain *ratio* of [overall allocations for insurance companies] [in the long *run*] .
Paraphrase	Liu Lefei says that [in the long *run*], in terms of *capital* allocation, overseas investment should occupy *the* certain *ratio* of an [insurance company's overall allocation]

Limits of Zhao et al.'s System

- This approach uses large paraphrase tables to successfully paraphrase the words in italics, making the source look more like the reference.
- But their system is unable to place *in the long run* at the end of the sentence or rephrase *an insurance company's overall allocations* as *overall allocations for insurance companies*.

Our Approach in a Nutshell

We start with a corpus of word-aligned paraphrases (Cohn et al. 2008).

1. The corpus sentence pairs are automatically parsed and assigned LFs using OpenCCG.
2. Word alignments (gold or automatic) are then projected onto the LFs.
3. Content from both LFs are merged to create a single disjunctive logical form.
4. The OpenCCG realizer is then used to generate new paraphrases that potentially incorporate content from both sentences in the paraphrase.

Parsing and Assigning LFs

- Corpus paraphrases are automatically parsed based on a broad-coverage CCG grammar extracted from the CCGbank and augmented with Propbank semantic roles.
- Parses are selected using Hockenmaier and Steedman's (2002) generative model.
- The derivation yields an LF that represents the semantic dependencies of a sentence as a graph, with lexical predicates as the nodes and relations between them as labeled edges.

Characterizing LF Differences

We start by characterizing the differences between the two LFs in a paraphrase as a set of edit operations.

Deletes occur when a subgraph of the first LF is not aligned.

Inserts are similar, involving unaligned subgraphs of the second LF.

Substitutions are subgraphs of the first LF that are aligned to subgraphs of the second LF.

Merging LFs

Our algorithm then creates a single disjunctive LF based on the graph edits (in both directions).

1. Each deleted or inserted subgraph is made optional from its parent (when aligned).
2. For (non-identical) substitutions, we force a choice at each parent node p between the substituted subgraph and its substitutions in the other LF that are subgraphs of a node aligned to p.

Optionality Example

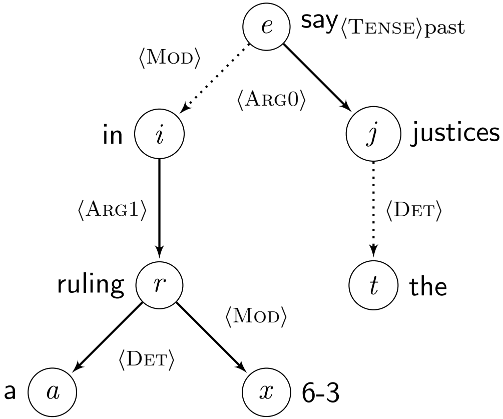
In (1b), the modifier phrase *In a 6-3 ruling* and the determiner *the* are inserted.

(1) a. Justices said that the constitution allows the government to administer drugs only in limited circumstances.

b. In a 6-3 ruling, the justices said such anti-psychotic drugs can be used only in limited circumstances.

Subgraph of the DLF for Example 1

The resulting subgraph in the merged LF contains optional nodes and edges (denoted here by dotted edge lines).



Choice Example

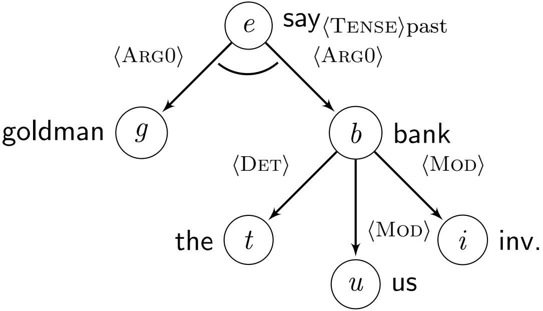
Example 2 shows a case where the subject of the matrix verb *said* can alternate between *The US investment bank* and *Goldman*:

(2) a. The US investment bank said: we believe the long-term prospects for the energy sector in the UK remain attractive.

b. We believe the long-term prospects for the energy sector in the UK remain attractive, Goldman said.

Subgraph of the DLF for Example 2

The choice between subjects is reflected in the DLF, which contains two possibilities for the ARG0 relation from *said* (shown here as an arc between ARG0 edges).



Lexical Diversity Pruning

- With the goal of creating more reference sentences, we need to counteract the OpenCCG realizer's tendency to generate candidates that re-use the same lexical choices.
- To achieve this, we use a **lexical diversity** pruning strategy:
  1. Group open class stems into equivalence classes, then
  2. Favor novel equivalence classes over those that have been previously used.
- The resulting *n*-best lists are more diverse with respect to lexical choices.

Incorporating a Self-Paraphrase Bias

- One immediately evident problem is due to a lack of named entity recognition in our system: the proper name *Charles O. Prince* is sometimes realized as *O. Charles Prince*.
- To counter this tendency, we use a version of Madnani's (2010) **self-paraphrase bias**.
- We implement this bias by adding a weighted n-gram precision score (approximating BLEU) to the realizer's perceptron ranking model score.

Example *n*-best List

Reference 1	lee said brianna had dragged food , toys and other things into the bedroom .
Reference 2	lee , 33 , said the girl had dragged the food , toys and other things into her mother 's bedroom .
Realizations	lee said *the girl* had dragged food , toys and other things into the bedroom . lee said brianna had dragged food , toys and other things into the bedroom . lee said , the girl had dragged [into the bedroom] food , toys and other things . lee said the girl has dragged into the bedroom food , toys and other things . . . . lee said the girl had dragged food , toys and other things into *her mother 's* bedroom . . . .

Preliminary Evaluation

	perfect	no dups	0-1 edits	N
rev. realization	1.07	0.60	4.67	360
gold alignments	1.87	1.61	4.78	276
auto alignments	**1.77**	**1.32**	4.91	264

- Intrinsic evaluation of 12-best lists from news sub-corpus (2nd author), emphasizing precision.
- Table shows average number of perfectly acceptable paraphrases per list, with and without ones duplicating the original reference, plus average number of paraphrases with up to one edit to be perfect.
- Number of perfect paraphrases generated with automatic alignments significantly higher than with direct reverse realization (χ^2 test, $p = 0.02$, $p = 0.005$).

Our Contributions

- We have shown how pairs of **word-aligned paraphrases** can be used **to create disjunctive logical forms**, from which OpenCCG can realize new paraphrases.
- A preliminary intrinsic evaluation indicates that the method yields significantly **more fully acceptable paraphrases** than direct reverse realization.
- The **lexical diversity pruning strategy** and **self-paraphrase bias** are essential; the latter helps avoid mangled named entities and ordering errors, as well as some parsing errors.

Loose Ends and New Directions

- Coverage remains an issue (under 50% currently; better cutoffs needed?), though this is less essential for the MT evaluation scenario.
- Investigating DLF rules for additional grammatical paraphrases, e.g. syntactic frame alternations.
- With high precision, automatically generated reference sentences can be expected to improve correlations of automatic MT evaluation metrics with human judgments; soon plan to verify this hypothesis.
- Synonym and paraphrasing matching in METEOR (Banerjie and Lavie, 2005) or TERp (Snover et al., 2010) should complement the new reference sentences generated with OpenCCG.