Parse Tree Fragmentation of Ungrammatical Sentences

Homa B. Hashemi, Rebecca Hwa

Intelligent Systems Program
University of Pittsburgh

25th International Joint Conference on Artificial Intelligence (IJCAI)
July 2016
Parsing uncovers the hidden structure of a sentence:

“who did what to whom”

Parsing is useful for many NLP tasks, like:

- Machine Translation
- Information Extraction
- Summarization/Compression
- Text Simplification
- Web Search

If the parse is wrong, it would affect the downstream applications.
Parsing ungrammatical sentences

- Some example domains of ungrammatical sentences:
  - Writings of non-native speakers
  - Machine translation outputs
- Parsers produce full, syntactically well-formed trees that are not appropriate for ungrammatical sentences

Example: MT output

```
(Representing a sentence from a machine translation output)
```

```
S
  NP
    NP
      DT
      The
    NNS
      members
  PP
    IN
      of
  VP
    VBZ
      opposes
    S
      NP
        DT
        the
      NN
        vote
    NP
      DT
      any
    PRP
      him
```
Parsing ungrammatical sentences

- Some example domains of ungrammatical sentences:
  - Writings of non-native speakers
  - Machine translation outputs

- Parsers produce full, syntactically well-formed trees that are not appropriate for ungrammatical sentences

Example: MT output

How to parse ungrammatical sentences?

1. Keep the full tree over a problematic sentence
2. Fix the sentence and the tree together
3. Our proposed approach: re-interpret parse trees
Our proposed approach: Parse Tree Fragmentation

- Identify well-formed syntactic structures for the parts that make sense
- **Parse tree fragmentation** is the process of breaking up the tree
- **Fragments** are reasonable isolated parts of parse trees

Example

Stanford Parse Tree

Coherent fragments
Developing a Fragmentation Corpus

- Why not manually annotate a fragmentation corpus?
  - Annotation projects are expensive and time-consuming
  - Fragmentation may depend on the specific NLP application

- Instead we leverage the existing corpora
1) Pseudo Gold Fragmentation (PGold)

Given an ungrammatical sentence and its error corrections

- ESL sentence: *I am very good swimming.*
- Teacher corrections: *I am very good at swimming.*

1. Replacing error

2. Unnecessary error

3. Missing error
Developing a Fragmentation Corpus: (1) PGold example

Example

Parse tree of the grammatical sentence:

*I am very good at swimming.*
Developing a Fragmentation Corpus: (1) PGold example

Example

Parse tree of the grammatical sentence:

I am very good at swimming.

PGold fragments of the ungrammatical sentence:

I am very good swimming.
2) REFerence Fragmentation (REF)

Given an ungrammatical sentence and a grammatical version of the same sentence:

1. Find the alignments between two trees
2. Assign fragments to aligned nodes

Example

Parse tree of grammatical sentence

Parse tree of ungrammatical sentence
Developing a Fragmentation Corpus: (2) REF

2) REference Fragmentation (REF)
Given an ungrammatical sentence and a grammatical version of the same sentence:

1. Find the alignments between two trees
2. Assign fragments to aligned nodes

Example

Parse tree of grammatical sentence

Parse tree of ungrammatical sentence
2) REference Fragmentation (REF)

Given an ungrammatical sentence and a grammatical version of the same sentence:

1. Find the alignments between two trees
2. Assign fragments to aligned nodes

Example

 Parse tree of grammatical sentence

 Parse tree of ungrammatical sentence

Parse Tree Fragmentation of Ungrammatical Sentences
Our contributions

1. Development of a Fragmentation Corpus
   - Pseudo Gold Fragmentation (Gold)
   - REference Fragmentation (REF)

2. Fragmentation Methods
   - Classification-based Fragmentation (CLF)
   - TreeBank-based Fragmentation (TBF)
1) Classification-based Parse Tree Fragmentation (CLF)

- **Binary classification**: Each edge is kept or cut
- **Training data**: Parse trees fragments by REF
- **Features**:
  1. Labels of parent, child, grandparent
  2. Depth & height of parent, child
  3. Word bigrams and trigrams
  4. CFG rule frequencies in Treebank
1) Treebank-based Parse Tree Fragmentation (TBF)

- For domain that do not have parallel corpora, we back off to available resources
- Use context free grammar rule frequencies in treebank to keep or cut an edge

Example

```
S.
 .
NP VP.
 PRP VBP NP.
 I am ADJP NN.
 .
 RB JJ swimming.
 very good
```

Parse tree of ungrammatical sentence
1) Treebank-based Parse Tree Fragmentation (TBF)

- For domain that do not have parallel corpora, we back off to available resources
- Use context free grammar rule frequencies in treebank to **keep or cut** an edge

Example

Parse tree of **ungrammatical sentence**

- S
  - NP
    - PRP I
    - VBP am
    - ADJP
      - RB very
      - JJ good
      - NN swimming
  - VP
    - NP
      - VBP am
      - ADJP
        - RB very
        - JJ good
        - NN swimming

TBF fragments
**Data**

1. **English as a Second Language corpus (ESL)**
   - ESL sentence: *We live in changeable world.*
   - Teacher corrections: add(3, a), replace(4, changing)
   - 5000 sentences with 1+ errors
   - 7000 sentences with 0+ errors

2. **Machine Translation outputs (MT)**
   - MT output: *What can we now?*
   - Human post-edit: *What can we do now?*
   - Fluency score calculated by edit rates (HTER)
   - 4000 sentences with HTER score $> 0.1$
   - 6000 sentences with HTER scores $\geq 0$
Intrinsic Evaluation: Evaluation of Tree Fragmentation Methods

- Similarity of fragmentation methods with PGold fragments over ESL dataset

<table>
<thead>
<tr>
<th>method</th>
<th>avg. # of fragments</th>
<th>avg. size of fragments</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gold</td>
<td>6</td>
<td>10.9</td>
<td>-</td>
</tr>
<tr>
<td>Reference</td>
<td>5.7</td>
<td>13.2</td>
<td>0.86</td>
</tr>
<tr>
<td>Classification-based</td>
<td>7.1</td>
<td>9.3</td>
<td>0.74</td>
</tr>
<tr>
<td>Treebank-based</td>
<td>8.9</td>
<td>7.8</td>
<td>0.72</td>
</tr>
</tbody>
</table>

CLF using 10-fold cross validation with the standard Gradient Boosting Classifier [Friedman, 2001]
Extrinsic Evaluation: Fluency Judgment

**Binary classification**: a sentence has virtually no error or many errors

**Regression**: Predict number of errors in ESL dataset or HTER in MT dataset

**Our feature set**: number, avg. size, min size, max size of fragments

<table>
<thead>
<tr>
<th>feature set</th>
<th>ESL Classification</th>
<th>ESL Regression</th>
<th>MT Classification</th>
<th>MT Regression</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.(%)</td>
<td>AUC</td>
<td>Pearson's r</td>
<td>Acc.(%)</td>
</tr>
<tr>
<td>LM</td>
<td>76.7</td>
<td>0.73</td>
<td>0.279</td>
<td>74.4</td>
</tr>
<tr>
<td>C&amp;J</td>
<td>76.3</td>
<td>0.74</td>
<td>0.318</td>
<td>68.3</td>
</tr>
<tr>
<td>TSG</td>
<td>77.3</td>
<td>0.74</td>
<td>0.285</td>
<td>69.8</td>
</tr>
<tr>
<td>Gold</td>
<td>100</td>
<td>1</td>
<td>0.928</td>
<td>-</td>
</tr>
<tr>
<td>Reference</td>
<td>99.8</td>
<td>1</td>
<td>0.84</td>
<td>94.4</td>
</tr>
<tr>
<td>Classification (CLF)</td>
<td>79.9</td>
<td>0.81</td>
<td>0.377</td>
<td>73</td>
</tr>
<tr>
<td>Treebank-based</td>
<td>77.2</td>
<td>0.74</td>
<td>0.298</td>
<td>71.8</td>
</tr>
<tr>
<td>CLF + LM</td>
<td>82.2</td>
<td>0.86</td>
<td>0.462</td>
<td>74.7</td>
</tr>
</tbody>
</table>

Experiments using 10-fold cross validation with Gradient Boosting Classifier
Introducing the new task of **parse tree fragmentation**

Extracting gold standard fragments using existing corpora for other NLP applications

Proposing two practical fragmentation methods (CLF and TBF)

Thank You