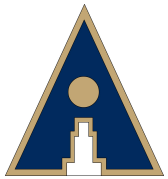
Parse Tree Fragmentation of Ungrammatical Sentences

Homa B. Hashemi, Rebecca Hwa



Intelligent Systems Program
University of Pittsburgh

25th International Joint Conference on Artificial Intelligence (IJCAI)
July 2016

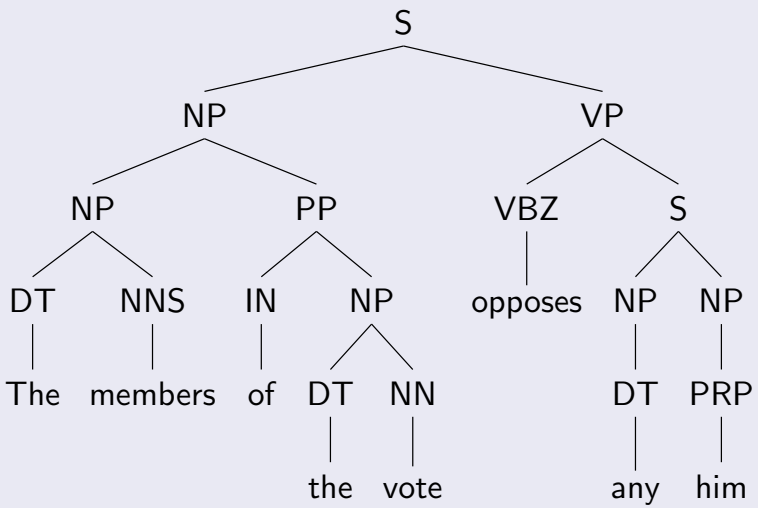
Parsing

- 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

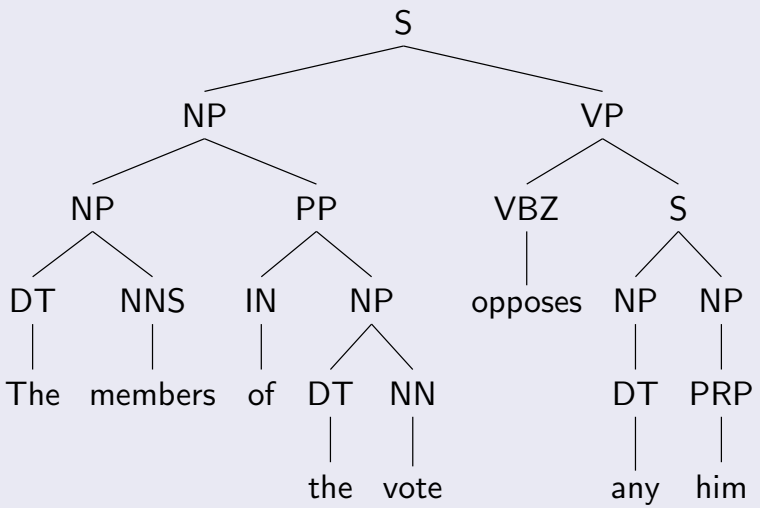
Example: MT output



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



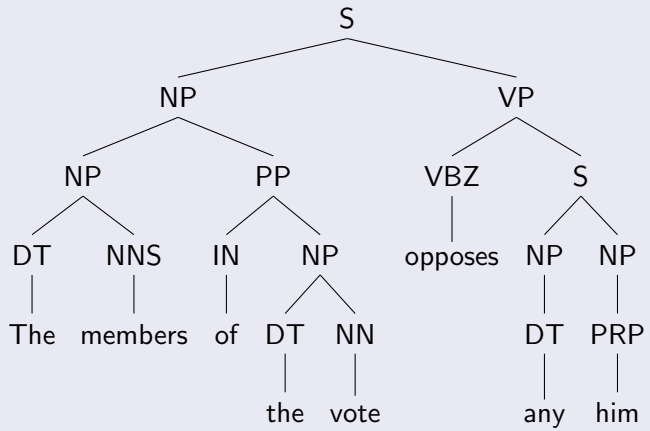
How to parse ungrammatical sentences?

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

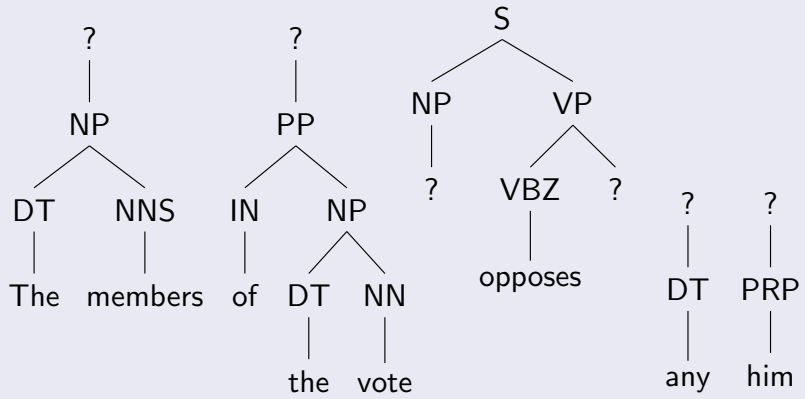
Our proposed approach: Parse Tree Fragmentation

- 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

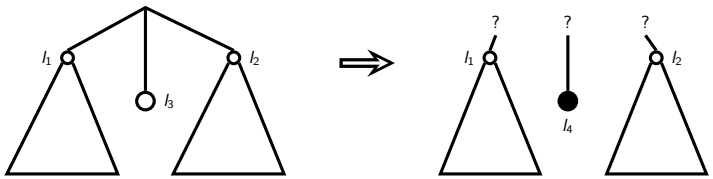
Developing a Fragmentation Corpus: (1) PGold

1) Pseudo Gold Fragmentation (PGold)

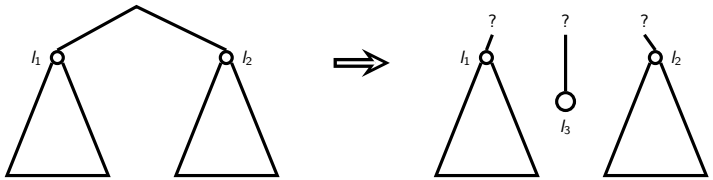
Given an ungrammatical sentence and its error corrections

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

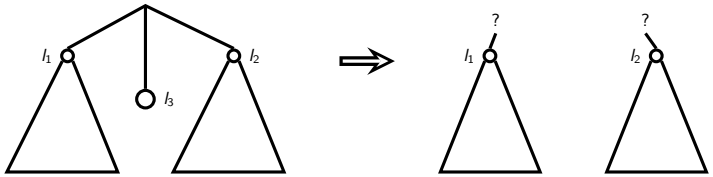
1. Replacing error



2. Unnecessary error

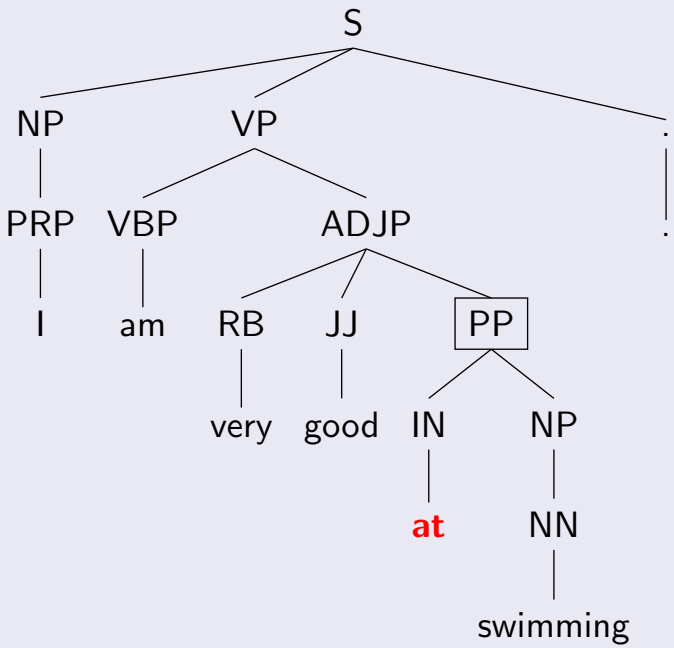


3. Missing error



Developing a Fragmentation Corpus: (1) PGold example

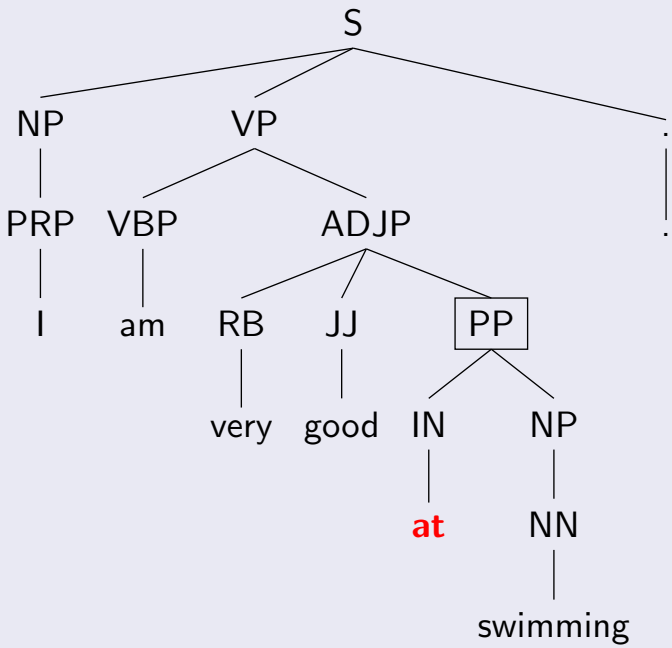
Example



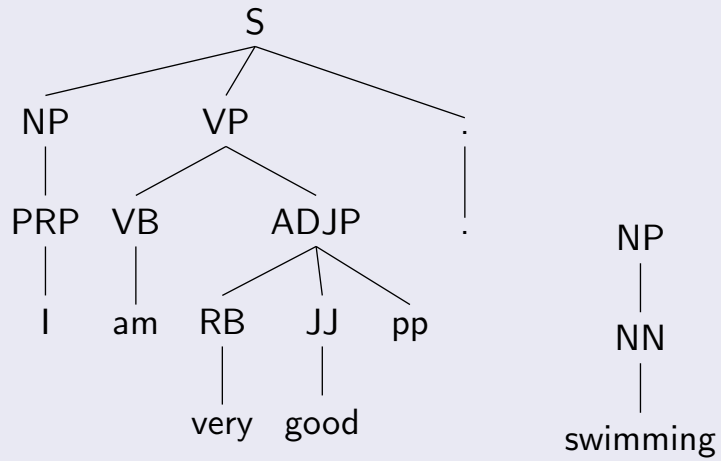
Parse tree of the grammatical sentence:
I am very good at swimming.

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.

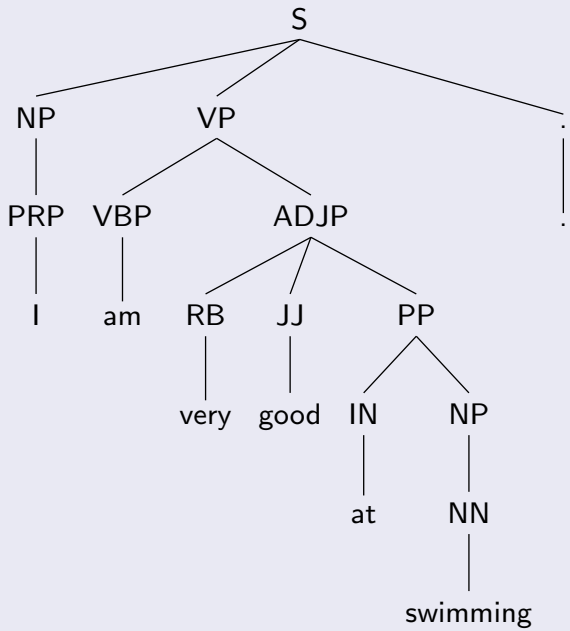
Developing a Fragmentation Corpus: (2) REF

2) REference Fragmentation (REF)

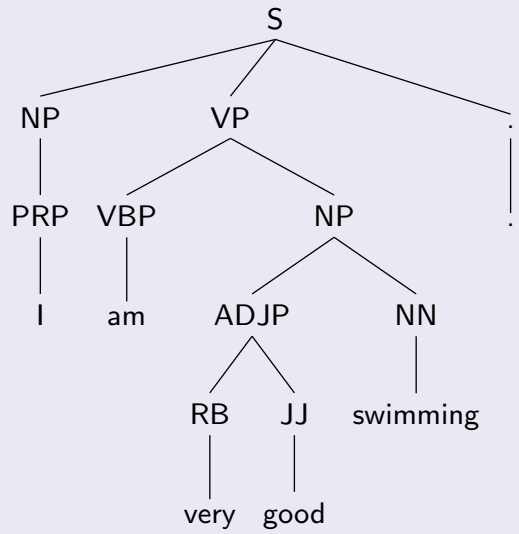
Given an ungrammatical sentence and a grammatical version of the same sentence:

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

Example



Parse tree of grammatical sentence



Parse tree of ungrammatical sentence

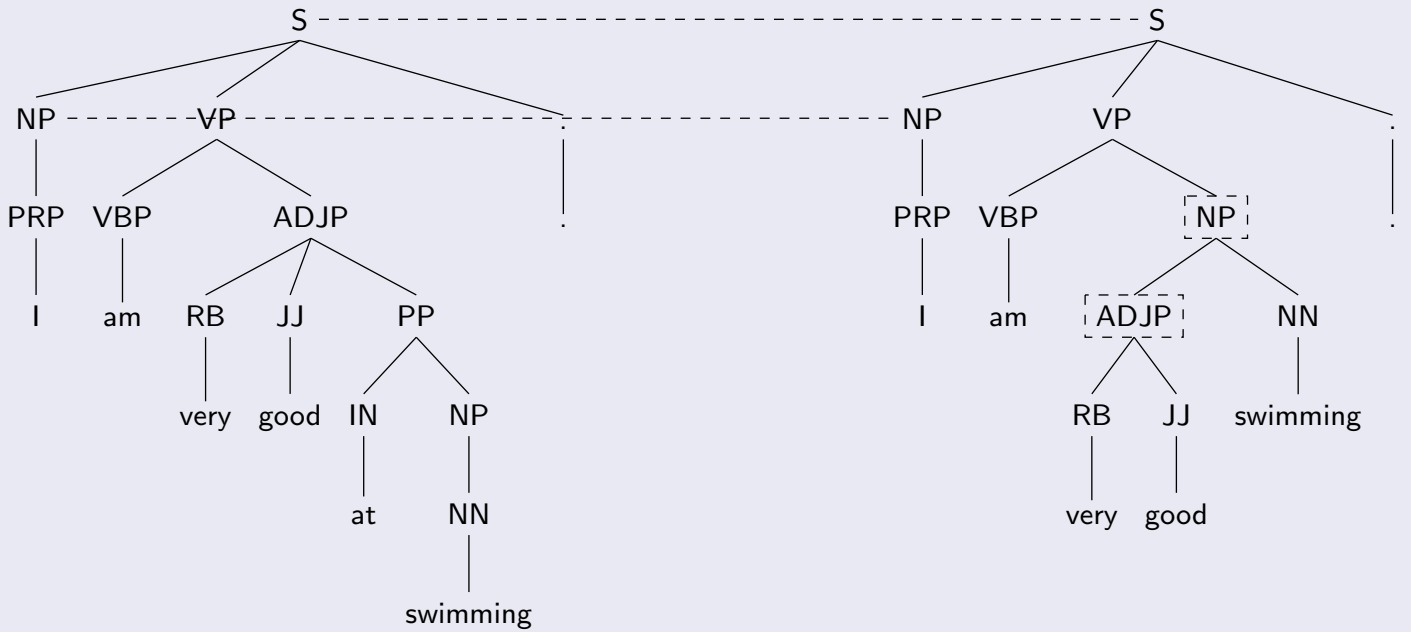
Developing a Fragmentation Corpus: (2) REF

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

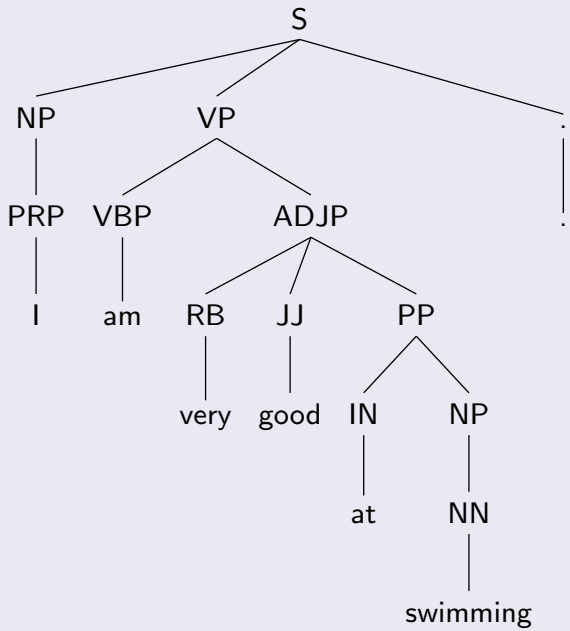
Developing a Fragmentation Corpus: (2) REF

2) REference Fragmentation (REF)

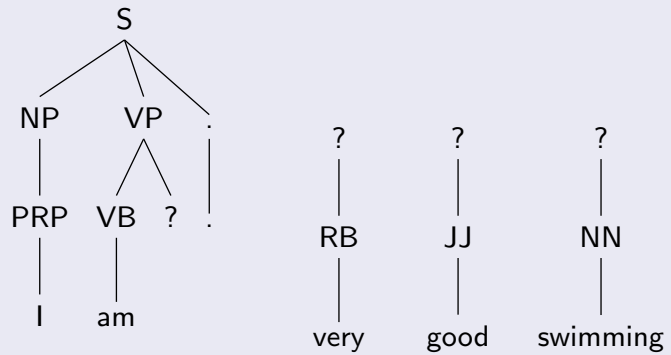
Given an ungrammatical sentence and a grammatical version of the same sentence:

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

Example



Parse tree of grammatical sentence



REF fragments of ungrammatical sentence

Our contributions

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

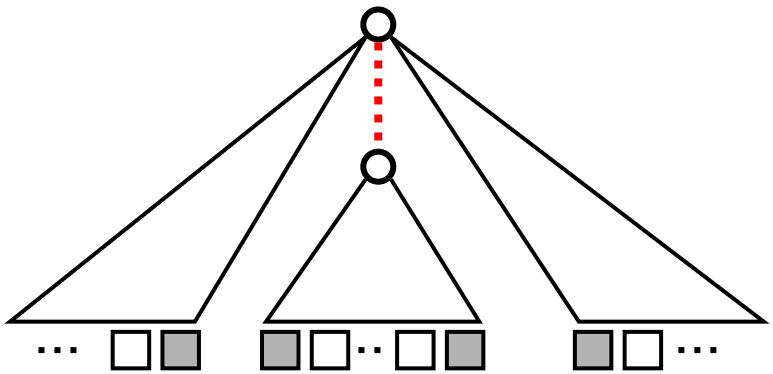
2. Fragmentation Methods
   - Classification-based Fragmentation (CLF)
   - TreeBank-based Fragmentation (TBF)

Fragmentation methods: (1) CLF

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

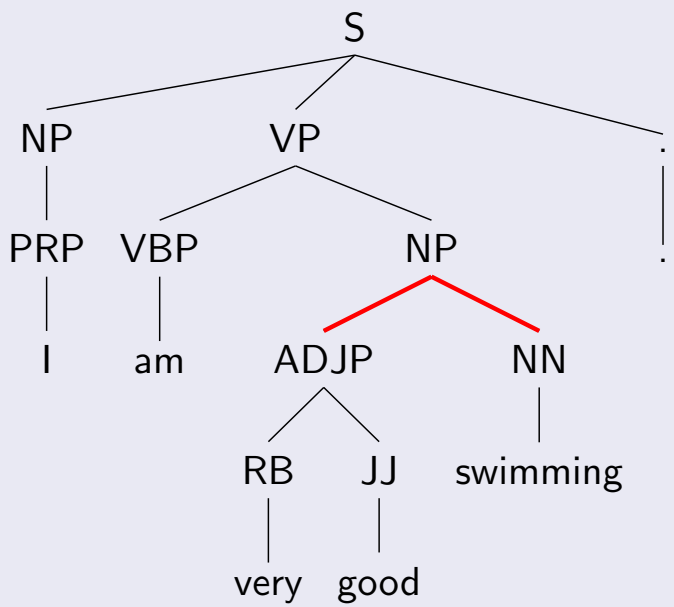


Fragmentation methods: (2) TBF

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



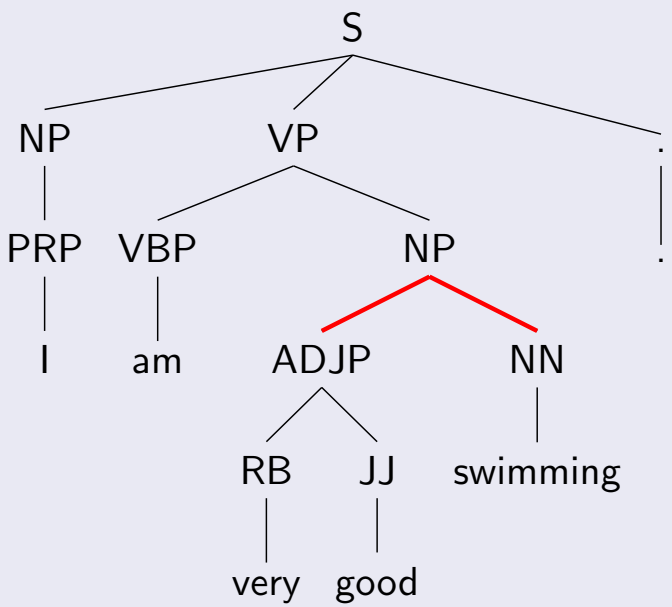
Parse tree of ungrammatical sentence

Fragmentation methods: (2) TBF

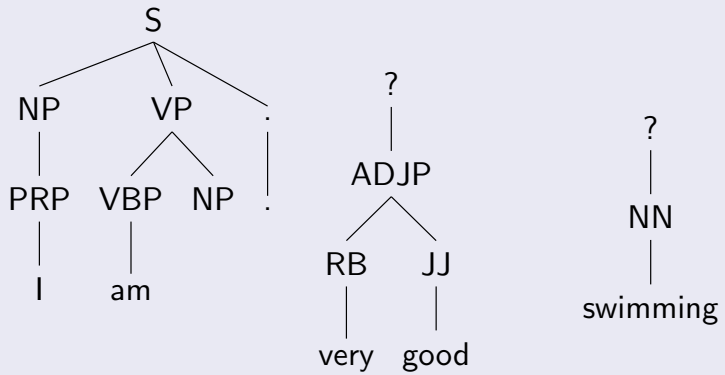
1) Treebank-based Parse Tree Fragmentation (TBF)

- 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



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 $\geqslant 0$

Intrinsic Evaluation: Evaluation of Tree Fragmentation Methods

- Similarity of fragmentation methods with PGold fragments over ESL dataset

method	avg. # of fragments	avg. size of fragments	F-score
Gold	6	10.9	-
Reference	5.7	13.2	**0.86**
Classification-based	7.1	9.3	0.74
Treebank-based	8.9	7.8	0.72

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

	ESL			**MT**		
	Classification		Regression	Classification		Regression
feature set	Acc.(%)	AUC	Pearson's r	Acc.(%)	AUC	Pearson's r
LM	76.7	0.73	0.279	74.4	0.71	0.307
C&J	76.3	0.74	0.318	68.3	0.6	0.136
TSG	77.3	0.74	0.285	69.8	0.59	0.105
Gold	100	1	0.928	-	-	-
Reference	99.8	1	0.84	94.4	0.99	0.782
Classification (CLF)	79.9	0.81	0.377	73	0.66	0.205
Treebank-based	77.2	0.74	0.298	71.8	0.51	0.04
CLF + LM	**82.2**	**0.86**	**0.462**	74.7	**0.73**	**0.324**

Experiments using 10-fold cross validation with Gradient Boosting Classifier
C&J: Charniak&Johnson, "Coarse-to-fine n-best parsing and MaxEnt discriminative reranking", ACL 2005.
TSG: Post, "Judging grammaticality with tree substitution grammar derivations", ACL 2011.

Conclusion

- 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