Discriminative Reranking of Discourse Parses Using Tree Kernels

Shafiq Joty and **Alessandro Moschitti**
Qatar Computing Research Institute

معهد قطر لبحوث الحوسبة
Qatar Computing Research Institute
Member of Qatar Foundation عضو في مؤسسة قطر

Motivation

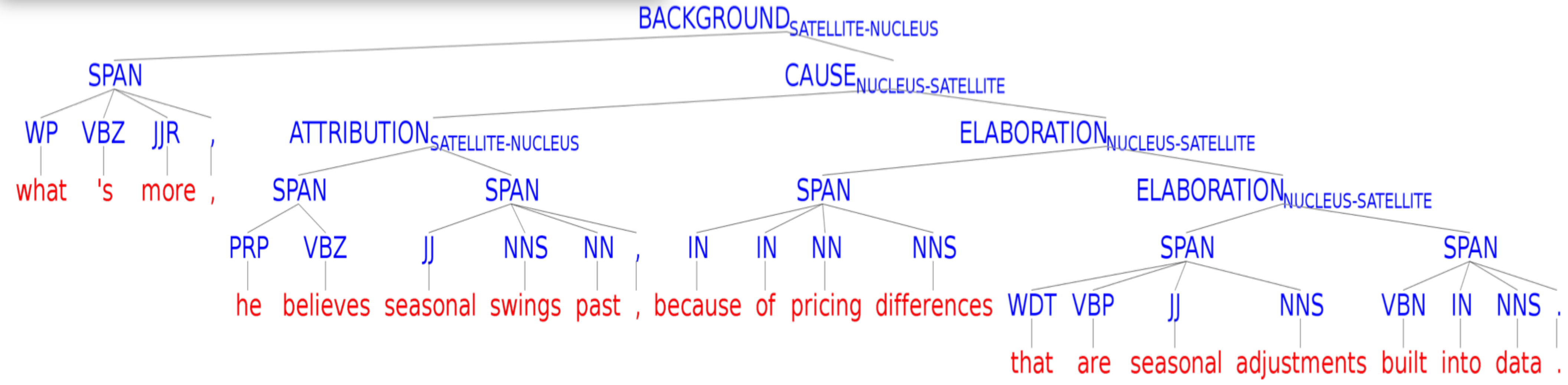
- Existing parsers fail to capture long range structural dependencies between constituents of a discourse tree.
- A reranker can exploit the global information as follows:
  ① A base parser produces *k* hypotheses.
  ② A classifier selects the best hypothesis by exploiting entire information in each hypothesis.
- Tree Kernels (TKs) allow kernel-based learning models like SVMs to learn from arbitrary tree fragments.

Our Method

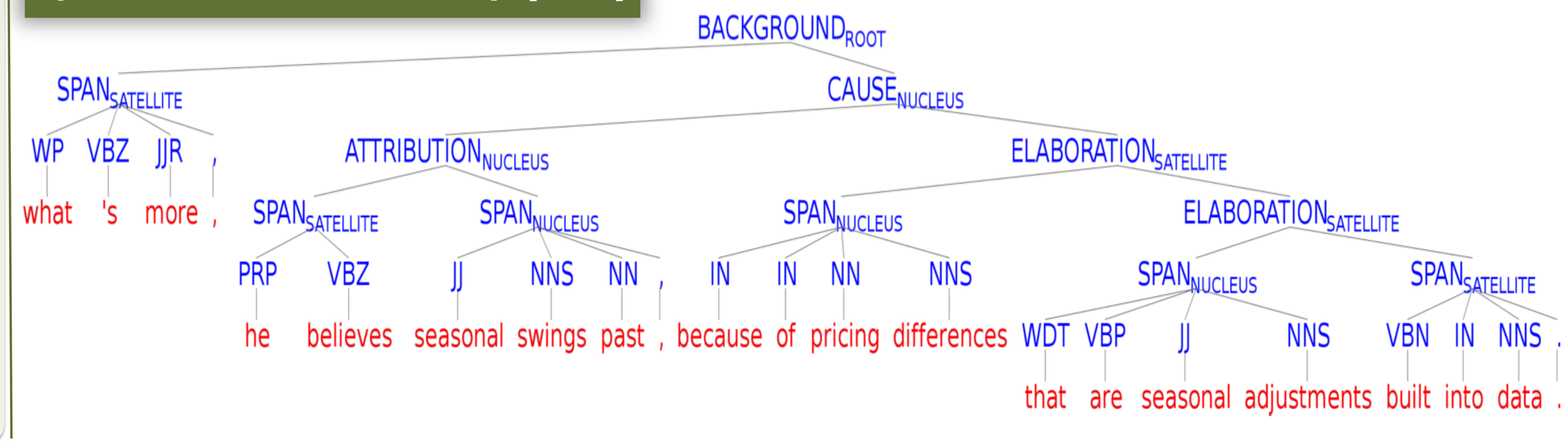
- Extend the parser of Joty et. al., (2013) to *k*-best parsing.
- Define novel kernels for discourse trees based on new representations.
- Use SVM preference reranking framework to rank *k* hypotheses and select the best tree.

Representations of Discourse Tree

Joint Relation-Nuclearity (JRN)



Split Relation-Nuclearity (SRN)



Preference Reranking

- Given a pair of hypotheses **<h$_i$, h$_j$>**, a classifier decides whether h$_i$ is a better tree than h$_j$.
- Training: **+ve <h$_1$, h$_j$>** -ve **<h$_j$, h$_1$>**
  h$_1$ has the highest f-score accuracy
- Testing: Apply the classifier to all possible pairs and take votes to rank the *k* candidates.

Preference Kernel

$$PK(\langle h_1, h_2\rangle, \langle h_1', h_2'\rangle) =$$

$$(\phi_K(h_1) - \phi_K(h_2)) \circ (\phi_K(h_1') - \phi_K(h_2')) =$$

$$K(h_1, h_1') + K(h_2, h_2') - K(h_1, h_2') - K(h_2, h_1')$$

Non sub-tree features

- Base parser rank & probability
- Structural properties of the DT
- Relation features

Experiments

Data (RST-DT)

- 385 news articles
  Train: 347 (7321 sent.)
  Test: 38 (951 sent.)
  Relation set: 18 coarser
- 5-fold CV was used to generate reranking data

Oracle scores

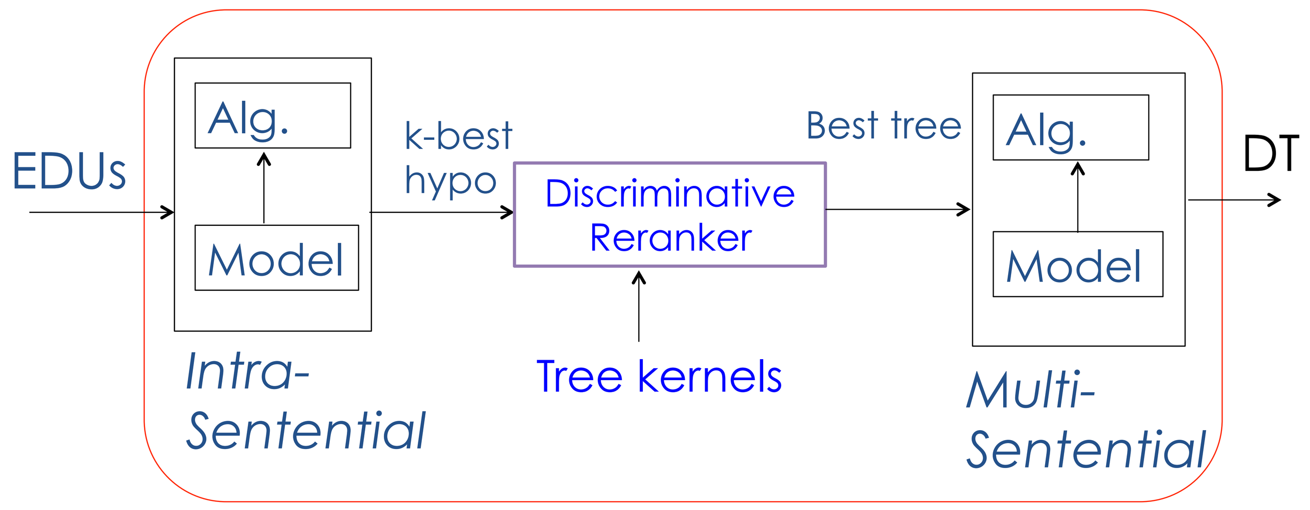
Sentence-level ☑

k	1	2	3	4	5	10	15	20	25	30
PAR-S	79.77	84.42	86.55	87.68	88.09	90.37	91.74	92.57	92.95	93.22

Document-level ☒

k	1	2	3	4	5	10	15	20	25	30
TSP 1S-1S	55.83	56.52	56.67	56.80	56.91	57.23	57.54	57.65	57.67	57.74

- Can a reranker improve at the sentence level?
- How much can the improvement push the document-level accuracy?



Results

- **Which TK works better on which representation?**

$\phi_{TK} \circ \phi_M$	JRN		SRN	
	Bigram	All	Bigram	All
STK	81.28	80.04	82.15	80.04
STK$_b$	81.35	80.28	82.18	80.25
PTK	81.63	78.50	81.42	78.25

Syntactic Tree Kernel (STK)
Partial Tree Kernel (PTK)

- **How does reranking performance vary for different values of k?**

	Standard test set						5-folds (average)					
	k=1	k=2	k=3	k=4	k=5	k=6	k=1	k=2	k=3	k=4	k=5	k=6
RR	79.77	81.08	81.56	**82.15**	**82.15**	82.11	78.57	79.76	80.28	80.68	80.80	**80.86**
ERR	-	6.48	8.85	**11.76**	**11.76**	11.57	-	5.88	8.45	10.43	11.02	**11.32**
OR	79.77	84.42	86.55	87.68	88.09	88.75	78.57	83.20	85.13	86.49	87.35	88.03

- **Which features are important?**

Baseline	Basic feat.	+ Rel. feat.	+ Tree
79.77	79.84	79.81	82.15

- **Overall document-level accuracy**

	(Joty et al., 2013)	With Reranker
PAR-D	55.8	**57.3**

Our Findings

- Bigram lexicalization is better then All.
- STK performs better than PTK on SRN.
- Best result is obtained for k=4,5 on std. testset.
- Improvement is consistent on whole corpus.
- Best result is obtained for k=6 on whole corpus.
- Non-subtree features doesn't help much.
- Subtree features learnt automatically are indeed crucial for the performance gain.
- Reranking at the sentence-level significantly pushes the state-of-the-art overall accuracy.

Reference: Shafiq Joty, Giuseppe Carenini and Raymond Ng. Combining Intra- and Multi-sentential Rhetorical Parsing for Document-level Discourse Analysis. In *ACL'13*.