

# COSI 137 Information Extraction - Final Project

## Event Relation Extraction in Syntax Constructions using Tree Kernel

Chuan Wang<sup>1</sup>

<sup>1</sup>Department of Computer Science  
Brandeis University

May 7, 2013

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results

# Event Relation Extraction

- Event relation extraction is to find the temporal relation between events in natural language processing.
- TimeML, which contains parts of the temporal structure of the 186 document Timebank corpus, aims to provide researchers the structured data to train models for extracting events and temporal structure.

# Event Relation Extraction

## TimeML Example

### Example

*Pacific First Financial Corp.* < EVENT eid = "e1" class = "REPORTING" > said < /EVENT > shareholders < EVENT eid = "e2" class = "OCCURRENCE" > approved < /EVENT > its < EVENT eid = "e8" class = "OCCURRENCE" > acquisition < /EVENT > by Royal Trustco Ltd. of Toronto.  
< TLINK lid = "l1" relType = "BEFORE" eventInstanceID = "ei80" relatedToTime = "t10" signalID = "s12" / >  
< SLINK lid = "l7" relType = "EVIDENTIAL" eventInstanceID = "ei73" subordinatedEventInstance = "ei74" / >

Compared to event detection, event relation i.e. temporal ordering could be more challenging because of the data sparsity and annotation inconsistency

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

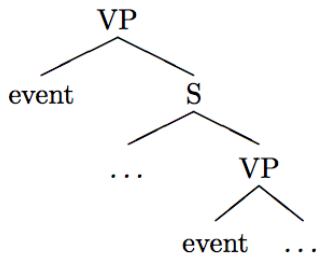
- Data Preparation
- Experiment Set and Results

# Verb-clause Syntactic Construction

- In the verb-clause construction, the first event is a verb and the second event is the head of a clausal argument to that verb. While this syntactic pattern is fairly specific, it occurs quite frequently in the TimeBank, almost 50% of adjacent pairs of verbal events participate in exactly a verb-clause construction.

# Verb-clause Syntactic Construction

- In the verb-clause construction, the first event is a verb and the second event is the head of a clausal argument to that verb. While this syntactic pattern is fairly specific, it occurs quite frequently in the TimeBank, almost 50% of adjacent pairs of verbal events participate in exactly a verb-clause construction.





## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

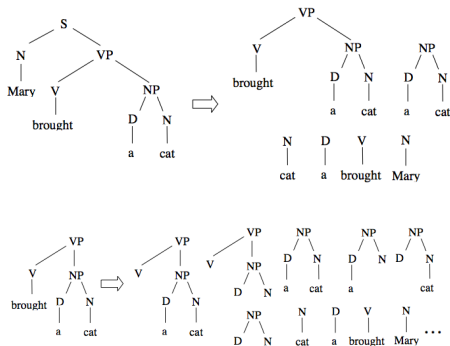
## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results

# Subtrees(STs) and Subset trees (SSTs)



Both the subtrees (STs) and subset trees (SSTs) can be regarded as two different ways of representing the features extracted from syntax tree.

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results

# Tree Kernel Functions

the definition of tree kernel is given as,

$$K(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \phi(n_1, n_2) \quad (1)$$

# Tree Kernel Functions

the definition of tree kernel is given as,

$$K(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \phi(n_1, n_2) \quad (1)$$

where  $\phi(n_1, n_2)$  can be computed in polynomial time using recursive definition

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results

# Data Preparation

- The 132 newswire documents in Wall Street Journal section of TimeBank has been selected and also the associated gold-standard syntactic trees from the TreeBank
- I only select the event pairs in the verb-clause construction

<b>Class</b>	<b>Train</b>	<b>Test</b>
AFTER/BEFORE	99	46
SIMUL	65	50
MODAL	225	91
EVIDENTIAL	300	141
<b>Total</b>	<b>689</b>	<b>424</b>

Merge the IS\_INCLUDED, INCLUDES, DURING, SIMULTANEOUS label because of sparsity and they are difficult to classify (even for human).

## 1 Introduction

- Event Relation Extraction
- Verb-clause Syntactic Construction

## 2 Tree Kernel method

- Subtrees(STs) and Subset trees (SSTs)
- Tree Kernel Functions

## 3 Experiments

- Data Preparation
- Experiment Set and Results



# Experiment Set

- Using SVM-light tree kernel
- Since SVM-light is binary classifier, use one-vs-all method to handle the multi-class
- 3 Experiments:
  - Tree Kernel only (using the minimum subtree which dominates the two events )
  - Tree Kernel + manual features extracted from TimeML corpus like tense, aspect, stem and POS, etc.
  - Manual features with SVM polynomial kernel

# Results

Class	Precision	Recall	F1
AFTER/BEFORE	0.2745	0.3043	0.288
SIMUL	0.3888	0.28	0.33
MODAL	0.7954	0.7692	0.78
EVIDENTIAL	0.8039	0.8723	0.84

Table: Tree Kernel only classification results

Class	Precision	Recall	F1
AFTER/BEFORE	0.4464	0.5435	0.49
SIMUL	0.5	0.24	0.32
MODAL	0.9277	0.8462	0.885
EVIDENTIAL	0.83	0.97	0.894

Table: Tree Kernel + manual features classification results

# Results

Class	Precision	Recall	F1
AFTER/BEFORE	0.4375	0.6086	0.509
SIMUL	0.411	0.14	0.2089
MODAL	0.939	0.8461	0.89
EVIDENTIAL	0.83	0.97	0.8954

Table: Manual features only classification results

# Summary

- As shown in many others' work, tree kernel only are not comparable to traditional manual selected features.
- However, tree kernel can give more useful information which can improve the performance of traditional manual feature engineering.