Recognizing Textual Entailment Using a Subsequence Kernel Method

Rui Wang & Günter Neumann

LT Lab at DFKI

Saarbrücken, Germany
Motivation: textual variability of semantic expression

Idea: given two text expressions T & H:
- Does text T justify an inference to hypothesis H?
- Is H semantically entailed in T?

PASCAL Recognising Textual Entailment Challenge
- 2007: 3rd RTE challenge, 25 research groups participated

A core technology for text understanding applications:
- Question Answering, Information Extraction, Semantic Search, Document Summarization, …
Processing of real text documents

☆ Semantic under-specification
  - Imprecise expressed semantic relationships
  - Vagueness, ambiguity

☆ Error tolerant methods needed
  - Noisy input data
  - Noisy intermediate component output

Different approaches consider/integrate features from different linguistics levels
Our goal: How far can we get with syntax only?

- **Subtree alignment on syntactic level**
  - Check similarity between tree of H and relevant subtree in T

- **Tree compression (redundancy reduction)**
  - Reduce noise from input/parsing
  - Yields compressed path-root-path sequences

- **Subsequence kernel**
  - Consider all possible subsequence of spine (path) difference pairs
  - SVM for classification
Sentence representation

☆ A sentence is represented as a set of triples of general form <head relation modifier>
  
  - Ex: Nicolas Cage’s son is called Kal’el

      <triple left="E0" right="6">fin:C i call:V </triple>
      <triple left="2" right="1">Nicolas_Cage:N lex-mod Nicolas:U </triple>
      <triple left="2" right="3">Nicolas_Cage:N poss 's:U </triple>
      <triple left="4" right="2">son:N gen Nicolas_Cage:N </triple>
      <triple left="6" right="4">call:V s son:N </triple>

☆ Dependency Structure
  
  - A DAG where nodes represent words and edges represent directed grammatical functions
  
  - We consider this as a “shallow semantic representation”
  
  - We use Minipar (Lin, 1998) and StanfordParser (Klein and Manning, 2003) as current parsing engines
System Overview: Feature Extraction

Backup Strategies

The Main Method
LT-Lab

System Workflow

Dependency Parser → T-H pairs

Apply Subsequence Kernel Method

Solved?

Backup Strategies Triple Matcher/BoW

No → Yes

Done
Basic idea, step 1: Dependency parsing

**Dependency Tree for T**

**Dependency Tree for H**
Basic idea, step 2: verb/noun subtree of H

Dependency Tree for T

Dependency Tree for H
Basic idea, step 3: Foot node alignment

Dependency Tree for T

Dependency Tree for H
LT-Lab Basic idea, step 4: Root node identification in T

Dependency Tree for T

Dependency Tree for H
LT-Lab Basic idea, step 5: Spine Difference

Dependency Tree for T

Dependency Tree for H
Basic idea, step 6: Root node alignment

Dependency Tree for T

Dependency Tree for H
Basic idea, step 7: Feature extraction

<table>
<thead>
<tr>
<th>Elementary Predicate</th>
<th>Left spine diff.</th>
<th>Right spine diff.</th>
<th>Verb cons.</th>
</tr>
</thead>
<tbody>
<tr>
<td>T:</td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>H:</td>
<td></td>
<td>ε</td>
<td></td>
</tr>
</tbody>
</table>
Pair: id=“61” entailment=“YES” task=“IE” source=“RTE”

– Text:

Although they were born on different planets, Oscar-winning actor Nicolas Cage's new son and Superman have something in common, both were named Kal-el.

– Hypothesis:

Nicolas Cage's son is called Kal-el.
Dependency Tree of T of pair (id=61):
Dependency Tree of H

of pair (id=61):

Nicolas Cage's son is called Kal-el.

- Observations
  - H is simpler than T
  - H can help us to identify the relevant parts in T
Dependency Tree of H of pair (id=61):

Nicolas Cage's son is called Kal-el.
Dependency Tree of T of pair (id=61):
☆ Left Spine #Root Node# Right Spine

– Text

Nicolas_Cage:N <PERSON> actor:N <GEN> son:N <SUBJ> have:V <I> fin:C <CN> fin:CN <OBJ1>

#Name:V#

<OBJ2> Kal-el:N

Nicolas_Cage:N & N <GEN> son:N <SUBJ> V <I> C <CN> CN <OBJ1> #Name:V# <OBJ2> Kal-el:N

Nicolas_Cage:N <GEN> son:N <SUBJ> V <SUBJ> #name:V# <OBJ> Kal-el:N

– Hypothesis

Nicolas_Cage:N <GEN> son:N <SUBJ> #call:V# <OBJ> Kal-el:N
Merging

- Left Spines: exclude Longest Common Prefixes
- Right Spines: exclude Longest Common Suffixes

RootNode Comparison

- Verb Consistence (VC)
- Verb Relation Consistence (VRC)

Nicolas_Cage:N <GEN> son:N <SUBJ> V <SUBJ> #name:V# <OBJ> Kal-el:N

Nicolas_Cage:N <GEN> son:N <SUBJ> #call:V# <OBJ> Kal-el:N

Left Spine Difference (LSD)
Pattern Format

- `<LSD, RSD, VC, VRC> → Predication`
- Example: `<“SUBJ V”, “”, 1, 1> → YES`

Closed-Class Symbol (CCS)

<table>
<thead>
<tr>
<th>Types</th>
<th>Symbols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dependency Relation Tags</td>
<td><code>SUBJ, OBJ, GEN, …</code></td>
</tr>
<tr>
<td>POS Tags</td>
<td><code>N, V, Prep, …</code></td>
</tr>
</tbody>
</table>

- LSD and RSD are either `NULL` or CCS sequences
Author Jim Moore was invited to argue his viewpoint that Oswald, acting alone, killed Kennedy.

Oswald killed Kennedy.
Text

......
<triple left="17" right="E0">kill:V mod-before vpsc:C</triple>
<triple left="17" right="16">kill:V punc , :U</triple>
<triple left="17" right="E8">kill:V subj Oswald:N</triple>
<triple left="17" right="18">kill:V obj Kennedy:N</triple>
......

Hypothesis
<triple left="E0" right="2">fin:C i kill:V</triple>
<triple left="2" right="1">kill:V s Oswald:N</triple>
<triple left="2" right="E2">kill:V subj Oswald:N</triple>
......

Oswald:N <SUBJ> V <SUBJ> #kill:V# <OBJ> Kennedy:N

<“SUBJ V”, “”, 1, 1> → YES
Entailment methods:

- Bag-of-Words (BoW)
- Triple Set Matcher (TSM)
- Minipar + Sequence Kernel + Backup Strategies (Mi+SK+BS)
- StanfordParser + Sequence Kernel + Backup Strategies (SP+SK+BS)

Classifier:

- SVM (SMO) classifier from the WEKA ML toolkit
From RTE challenges:
- RTE-2 Dev Set (800 T-H pairs) + Test Set (800 T-H pairs)
- RTE-3 Dev Set (800 T-H pairs) + Test Set (800 T-H pairs)

Additional data for IE and QA tasks:
- Automatically collected from MUC6, BinRel (Roth and Yih, 2004), TREC-2003
- Manually classified into yes/no concerning entailment relation
**Results on RTE-2 Data**

<table>
<thead>
<tr>
<th>Systems\Tasks</th>
<th>IE</th>
<th>IR</th>
<th>QA</th>
<th>SUM</th>
<th>ALL</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Exp A1: 10-Fold Cross-Validation on Dev+Test Set</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoW</td>
<td>50%*</td>
<td>58.8%</td>
<td>58.8%</td>
<td>74%</td>
<td>60.4%</td>
</tr>
<tr>
<td>TSM</td>
<td>50.8%</td>
<td>57%</td>
<td>62%</td>
<td>70.8%</td>
<td>60.2%</td>
</tr>
<tr>
<td>Mi+SK+BS</td>
<td>61.2%</td>
<td>58.8%</td>
<td>63.8%</td>
<td>74%</td>
<td>64.5%</td>
</tr>
<tr>
<td><strong>Exp A2: Train: Dev Set (50%); Test: Test Set (50%)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoW</td>
<td>50%</td>
<td>56%</td>
<td>60%</td>
<td>66.5%</td>
<td>58.1%</td>
</tr>
<tr>
<td>TSM</td>
<td>50%</td>
<td>53%</td>
<td>64.5%</td>
<td>65%</td>
<td>58.1%</td>
</tr>
<tr>
<td>Mi+SK+BS</td>
<td>62%</td>
<td>61.5%</td>
<td>64.5%</td>
<td>66.5%</td>
<td>63.6%</td>
</tr>
</tbody>
</table>

* The accuracy is actually 47.6%. Since random guess will achieve 50%, we take this for comparison.
## Results on RTE-3 Data

<table>
<thead>
<tr>
<th>Systems\Tasks</th>
<th>IE</th>
<th>IR</th>
<th>QA</th>
<th>SUM</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Exp B1: 10-fold Cross Validation on RTE-3 Dev Data</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BoW</td>
<td>54.5%</td>
<td>70%</td>
<td>76.5%</td>
<td>68.5%</td>
<td>67.4%</td>
</tr>
<tr>
<td>TSM</td>
<td>53.5%</td>
<td>60%</td>
<td>68%</td>
<td>62.5%</td>
<td>61.0%</td>
</tr>
<tr>
<td>Mi+SK+BS</td>
<td>63%</td>
<td>74%</td>
<td>79%</td>
<td>68.5%</td>
<td>71.1%</td>
</tr>
<tr>
<td>SP+SK+BS</td>
<td>60.5%</td>
<td>70%</td>
<td>81.5%</td>
<td>68.5%</td>
<td>70.1%</td>
</tr>
<tr>
<td><strong>Exp B2: Train: Dev Data; Test: Test Data</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mi+SP+SK+BS</td>
<td>58.5%</td>
<td>70.5%</td>
<td>79.5%</td>
<td>59%</td>
<td>66.9%*</td>
</tr>
</tbody>
</table>

* The 5th place of RTE-3 among 26 teams
## Components of the 5th best systems

<table>
<thead>
<tr>
<th>Systems</th>
<th>Acc. %</th>
<th>Lx*</th>
<th>Ng</th>
<th>Sy</th>
<th>Se</th>
<th>LI</th>
<th>C</th>
<th>ML</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hickl et al.</td>
<td>80.00</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Tatu et al.</td>
<td>72.25</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Iftene</td>
<td>69.13</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Adams</td>
<td>67.00</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>DFKI</td>
<td>66.87</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

* Following the notation in (Giampiccolo et al., 2007):
  
  Lx: Lexical Relation DB;
  Ng: N-Gram / Subsequence overlap;
  Sy: Syntactic Matching / Alignment;
  Se: Semantic Role Labeling;
  LI: Logical Inference;
  C: Corpus/Web;
  ML: ML Classification;
  B: Entailment corpora/Background Knowledge;
Puristic approach:
- We do not exploit any additional knowledge source beside the dependency trees nor have we extended the RTE training data

Relational method:
- For the IE task, SK method gives highest improvements
- Kernel method seem to be more appropriate if the underlying task reveals a more “relational nature”

Fallback strategies:
- The “shallow” methods realized through BoW and TSM seem to work better for IR and SUM.
☆ **IE:** MUC6, BinRel Corpus

- **T:** relevant sentence(s)

  - Dole had hoped to pull out a win in North Carolina, the home state of his wife, Elizabeth.

- **H:** NE + Relation + NE

  - Elizabeth is born in North Carolina.

☆ **QA:** TREC2003 QA

- **T:** (ir)relevant sentence(s)

  - Vice-President Albert Gore described the book "critically important" and compared it with "Silent Spring," Rachel Carson's 1962 book that set off a movement to ban DDT and other pesticides.

- **H:** question + answer

  - What book did Rachel Carson write in 1962?
  - Silent Spring
## Results for SK method

### Only SK method on Extra data (460 out of 750)

<table>
<thead>
<tr>
<th>Methods\tasks</th>
<th>IE (MUC,BinRel)</th>
<th>QA (TREC2003)</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>BoW</td>
<td>62.9%</td>
<td>61.4%</td>
<td>62.3%</td>
</tr>
<tr>
<td>TSM</td>
<td>64.9%</td>
<td>62.3%</td>
<td>63.8%</td>
</tr>
<tr>
<td>SK</td>
<td><strong>76.3%</strong></td>
<td><strong>65.7%</strong></td>
<td><strong>74.5%</strong></td>
</tr>
</tbody>
</table>

### Only SK method on RTE-2 data

<table>
<thead>
<tr>
<th>Exps\Tasks</th>
<th>IE</th>
<th>IR</th>
<th>QA</th>
<th>SUM</th>
<th>ALL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ExpA1: coverage</td>
<td>63.3%</td>
<td>18.3%</td>
<td><strong>36.3%</strong></td>
<td>16.3%</td>
<td>536</td>
</tr>
<tr>
<td>ExpA1: acc. of matches</td>
<td>64%</td>
<td>67.1%</td>
<td>66.2%</td>
<td>73.9%</td>
<td>66.2%</td>
</tr>
<tr>
<td>ExpA2: coverage</td>
<td>63.5%</td>
<td>23.5%</td>
<td><strong>44%</strong></td>
<td>17%</td>
<td>296</td>
</tr>
<tr>
<td>ExpA2: acc. of matches</td>
<td>66.9%</td>
<td>70.2%</td>
<td>58.0%</td>
<td>64.7%</td>
<td>64.5%</td>
</tr>
</tbody>
</table>
Coverage:

- For IE and QA pairs, SK+BS reveals a better coverage, more than a half
- For IR and SUM pairs, although it achieves good accuracies, the number of covered cases is low

Task-based strategy selection:

- IE and QA: SK+TSM
- IR: SK+BoW
- SUM: BoW
Future Work

☆ RTE core method

- Increase coverage of SK method
  - Integrate IE technology, especially NE recognition
  - Lexical semantics of function words
  - Extend to n-ary hypothesis texts
- Adapt to German language (e.g., rich morphology, noun compounds)

☆ Applications

- Entailment-based QA system on structured data (QALL-ME, project funded by European Commission)
- Unsupervised Relation extraction (IDEX, project funded by Investitionsbank Berlin)