# Recognising Textual Entailment Focusing on Non-Entailing Text and Hypothesis

**Rongzhou Shen, Thade Nahnsen, Claire Grover, Ewan Klein,**
University of Edinburgh
{rshen,t.nahnsen,grover,ewan}@inf.ed.ac.uk

## Abstract

This paper describes a predominantly shallow approach to the RTE-4 Challenge. We focus our attention on the non-entailing Text and Hypothesis pairs in the dataset. The system uses a Maximum Entropy framework to classify each pair of Text and Hypothesis as either YES or NO, using a range of different feature sets based on an analysis of the existing non-entailing pairs in RTE training data.

## 1 Introduction

In this paper, we address the problem of Recognizing Textual Entailment (RTE) by treating it as a classification task. The main features we used involved the overlap of Named Entities (NEs) and relations between them. We focused our attention on the two-way (YES/NO) decision as opposed to the three-way (YES/CONTRADICTION/UNKNOWN) decision introduced in RTE-4.

Up to now, three consecutive RTE Challenges were held. Approaches to the problem ranged from as simple as measuring the degree of word overlap, to as complex as using all sorts of deep language processing methods such as parsing, semantic analysis, coreference and so on. Although in both of Hickl et al (Hickl et al., 2006; Hickl and Bensley, 2007) submissions, a combination of deep methods has proven to perform very well, we decided to test the accuracy of using only shallow language processing methods. Furthermore, we incorporated a complex version of Relation Matching to test whether relations in the Text and Hypothesis can help in identifying entailments.

Prior to implementing a system, we carried out an analysis of the RTE data set that focused on the non-entailing pairs (i.e., the pairs that were labelled NO). Starting from the point of view that an entailment holds if all the information contained in the Hypothesis can be embedded into the information content of the Text, we grouped the non-entailing pairs in the following manner.

**Extraneous Information:** The Hypothesis contains information that is absent from the Text. Many of these cases involve extraneous NEs, as illustrated in pair ID 4 in Table 1. However, other cases involve nominal terms which are not NEs, such as that illustrated in pair ID 18 in Table 1.

**Incompatible Information:** The Text contains everything that is in the Hypothesis, but some of the information given in the Hypothesis is incompatible with what is given the Text. There are a number of reasons why this occurs, but we focussed on two subcases:

*Incompatible relations:* Although there is a mapping from nominal information in the Hypothesis to the Text, the relevant entities are related in incompatible ways, as illustrated by ID 14 in Table 1.

*Negation:* The polarity of the relation in Hypothesis is reversed in the Text, as illustrated by ID 48 in Table 1.

Given the analysis above, we discovered features, especially shallow features, that would help in determining the non-entailing Text and Hypothesis pairs. The system is built on top of preprocessed data and uses a Maximum Entropy classifier. [1]

## 2 Preprocessing

Before the classification stage, the RTE pairs were passed through a processing pipeline implemented using the LT-TTT2 tools (`http://www.ltg.ed.ac.uk/software/lt-ttt2`). Early stages of this process involved tokenisation and sentence splitting followed by part-of-speech tagging using Curran and Clark's 2003 Maximum Entropy Markov model tagger trained on the Penn Treebank. The last three stages were lemmatisation using Minnen and Carroll's 2000 *morpha* system, chunking using

[1] We used a Java based MAXENT package in OpenNLP: http://maxent.sourceforge.net

| Id | Text | Hypothesis | Entailment |
|---|---|---|---|
| 4 | Save the Children demanded action after its research found that starving and desperate youngsters as young as six were being coerced to sell sex for food, money, soap and even mobile phones in war zones and disaster areas. | **UN** peacekeepers abuse children | NO |
| 14 | Set in the New York City borough of The Bronx, the show starred Ted Danson as the title character, Dr. John Becker, a doctor who operates a small practice and is constantly annoyed by his patients, coworkers, friends, and practically everything and everybody else in his world. Becker has **never played tennis** in his life. | Becker was a **tennis champion** | NO |
| 18 | The victims' families, as well as women who survived Michel Fourniret's alleged attacks, sat opposite the accused and his wife Monique Olivier on the first day of the trial for the kidnap, rape and murder of seven young women and girls. | Michel Fourniret was sentenced to **life imprisonment**. | NO |
| 48 | **Indonesia** has **revisit**ed its **OPEC** membership, but decided to stay on to maintain high-level relations with bigtime oil powers like Saudi Arabia. After all, Indonesia has the world's highest Muslim population, giving it another major tie to Gulf exporters. | **Indonesia leave**s **OPEC**. | NO |

Table 1: Examples from RTE-4

the LT-TTT2 rule-based chunker (Grover and Tobin, 2006) and rule-based Named Entity Recognition which is part of the recent release of LT-TTT2. In addition to the NEs identified by LT-TTT2, we merged in the NEs produced by the Stanford NER Tagger Finkel et al. (2005). As an example, the XML in Figure 1 combined with the XML in Figure 2 would produce the results shown in Figure 3.

```
<entailment>
    <pair id='1'>
        <t>
            <wi num="0" entity="O">The</wi>
            <wi num="1" entity="O">sale</wi>
            <wi num="2" entity="O">was</wi>
            <wi num="3" entity="O">made</wi>
            <wi num="4" entity="O">to</wi>
            <wi num="5" entity="O">pay</wi>
            <wi num="6" entity="PERSON">Yukos</wi>
            ...
        </t>
        ...
    </pair>
    ...
</entailment>
```

Figure 1: XML produced by Stanford NER Tagger

```
<?xml version="1.0" encoding="UTF-8"?>
<entailment-corpus challenge="3">
  <pair id="1" entailment="YES" task="IE" length="short">
    <t type="t">
      <s id="s1">
        <ng>
          <w pws="yes" id="w5" p="DT" ent="O">The</w>
          <w headn="yes" vstem="sell" l="sale" pws="yes" id="w9" p="NN" ent="O">sale</w>
        </ng>
        <vg>
          <vg modal="no" asp="simple" voice="pass" tense="past">
            <rel arg2="w9" arg1="w18" type="ncsubj"/>
            <w l="be" pws="yes" id="w14" p="VBD" ent="O">was</w>
            <w headv="yes" l="make" pws="yes" id="w18" p="VBN" ent="O">made</w>
          </vg>
          <vg modal="no" asp="simple" voice="act" tense="inf">
            <rel arg2="w38" arg1="w26" type="dobj"/>
            <w pws="yes" id="w23" p="TO" ent="O">to</w>
            <w headv="yes" l="pay" pws="yes" id="w26" p="VB" ent="O">pay</w>
          </vg>
        </vg>
        <ng>
          <ng>
            <w headn="yes" l="yuko" pws="yes" id="w30" p="NNP" ent="O">Yukos</w>
          </ng>
        </ng>
      </s>
    </t>
  </pair>
</entailment-corpus>
```

Figure 2: XML produced by LT-TTT2

## 3 Shallow Features

Given the preprocessed data just described, we attempted to detect the kinds of non-entailing pairs described in Section 1 with corresponding match-

```
<entailment>
    <pair entailment="YES" id="0" task="IE">
        <t>
            <w chunk="ng0" headn="NA" id="w5" lemma="NA" ner="NA" pos="DT" si="31" tense="NA">The</w>
            <w chunk="ng0" headn="yes" id="w9" lemma="sale" ner="NA" pos="NN" si="31" tense="NA">sale</w>
            <w chunk="vg2" headn="NA" id="w14" lemma="be" ner="NA" pos="VBD" si="31" tense="past">was</w>
            <w chunk="vg2" headn="yes" id="w18" lemma="make" ner="NA" pos="VBN" si="31" tense="past">made</w>
            <w chunk="vg3" headn="NA" id="w23" lemma="NA" ner="NA" pos="TO" si="31" tense="inf">to</w>
            <w chunk="vg3" headn="yes" id="w26" lemma="pay" ner="NA" pos="VB" si="31" tense="inf">pay</w>
          <w chunk="ng3" headn="yes" id="w30" lemma="yuko" ner="PERSON" pos="NNP" si="31" tense="NA">Yukos</w>
        ...
        </t>
    ...
    </pair>
...
</entailment>
```

Figure 3: Preprocessed XML format

ing operations over Text / Hypothesis pairs:

- Identifying and matching NEs and other noun group chunks between Text and Hypothesis.
- Identifying and matching relational paths between the entities between Text and Hypothesis.

The degree of matching was then encoded as a feature value [NB was this high/medium/low/].

## 3.1 Named Entity Matching

As shown in Figure 4, NEs are initially identified in the Hypothesis and then searched for in the Text. We considered both exact and approximate matches using three distance measures:

**Levenshtein distance:** This is based on the minimum number of character-level operations (*delete*, *replace*, *insert*) needed to transform one string (in our case, the NE from the Hypothesis) into the other (the NE from the Text).

**Soundex:** This is an algorithm for detecting similarity in pronunciations, so that those names whose pronunciations are the same are mapped to the same alphanumeric code, despite the spelling differences.

**Substring:** The simplest of all, this finds out whether one NE is a substring of the other. If it is, then they are considered to be similar and not otherwise.

## 3.2 Noun Group Matching

As with NE Matching, Noun Group Matching serves the purpose of detecting whether there is extraneous information in the Hypothesis. Figure 4 illustrates a case where a match of the Noun Group in the Hypothesis (*ten members*) can be found in the Text (*10 members*). Noun Groups are identified in the Hypothesis with the LT-TTT2 chunker, and then matches are searched for in the Text using Levenshtein distance.

## 3.3 More Complex NE and Noun Group Matching

In contrast to the simple NE and noun matching, we also explored more complex versions for the matching, using linguistic resources such as WordNet (Fellbaum, 1998) and DIRT (Lin and Pantel, 2001), and linguistic information such as head constituents. For this purpose, the following variables are considered:

- The ordering of the words in the two groups compared.
- Whether only the heads or the whole groups are compared.
- Whether WordNet synonym matches are considered. (Early experiments showed that relaxing constraints further in the taxonomy led to severe problems of overgeneralization, resulting in numerous highly problematic matches.)
- Whether DIRT paraphrases help in determining matching. (For paraphrase matches to be successful, it is necessary to find all words from the DIRT paraphrase in the text; ordering, however, is not considered.)

## 3.4 Simple Relation Matching

Given our emphasis on using shallow features, the approach to relation matching at this stage is very simple. Features are encoded according to the following properties:

- Recording the presence of negation words in the Text / Hypothesis.
- For each sentence in the Text / Hypothesis, extracting the head nouns and verbs from the sentence and joining them as a single string to represent the relation; then determining whether these two relation representation forms are similar to each other using string comparison measures.

# 4 Relation Matching

As shown in Figure 5, relations are extracted from both the Text and Hypothesis and compared against each other for similarity. Relation Extraction relied on dependency paths generated by the RASP parser (Briscoe et al., 2006). The following steps are used to obtain dependency relations from the Text and Hypothesis:

1. Feed preprocessed XML-format text into the RASP system. [2] Example output is shown in Figure 6.

2. Use a SED script to extract all the relations from the RASP output and store them in XML format. For example, the RASP output given in Figure 6 would produce the XML in Figure 7.

3. Combine the relations-based XML with the preprocessed data XML by inserting each `rel` element into its corresponding RTE pair.

With the extracted relations, we then perform the following steps to compare the Text and Hypothesis for similarity:

1. Find the NEs, head nouns and head verbs in the Hypothesis

2. Use NE and Noun Group matching, find the entities in the Text that match the ones in the Hypothesis

3. For each pair of entities found in the Text, if the entities are within the same noun group, then ignore that pair, otherwise find a path between these two entities using the dependency relations we obtained.

4. If a path is found, extract all head verbs from these paths.

5. If head verbs are found, compare these head verbs with the ones in the Hypothesis using string comparison.

When performing the above comparison, features are encoded on the fly for feeding into the MAXENT classifier. These features are listed in Table 2.

# 5 Experiments and Results

In this section, we briefly present the performance of three versions of our system against the RTE-4 gold standard test data. Furthermore, we present the experiments done with the shallow features.

[2] We modified the RASP script to ensure that RASP tokenisation was consistent with LT-TTT2.

## 5.1 Experiments with Shallow Features

In order to evaluate the shallow features we have generated several results using different approaches. Three standard measures (precision, recall and F1 score) were used to evaluate the system in addition to the two (accuracy and average precision) defined by the RTE task. To start with, we implemented a simple word match system that bases the classification merely on setting a threshold on the number of exact word matches between the Text and Hypothesis (the number is normalised by the length of the Hypothesis). The results on the RTE-3 development dataset are shown in Table 3.

We can see that with only as simple a method as exact word matching, prediction on the QA category can achieve an accuracy of 71.50%. A recall for YES of as high as 93.46% but only having a precision of 66.67% indicates that the system predicted most of the data pairs as YES, meaning the contents of the Text and Hypothesis in QA data pairs are very much alike. Similarly for SUM and IR, although accuracies for both are lower than that of QA, their precision and recall values resemble those of QA, thus indicating that similarity can be a key issue in the classification. However, although measuring similarity can predict almost all the positive data pairs in IE, it cannot do the same for negative pairs. Also notice that the precision for the negative pairs was low too. This means that although the Text and Hypothesis in the IE category are similar in contents, their similarity *does not* contribute much to the prediction of negative pairs.

Taking the above observations into consideration, we focused on predicting non-entailing pairs and tested whether improving the matching algorithms helped in increasing the performances of the system. After training a Maximum Entropy classifier with the shallow features given in Section 3 on the RTE-1 and RTE-2 corpus, we tested it on the RTE-3 development dataset. Table 4 shows the results.

As can be seen, the accuracies in Table 4 are a large improvement over the word match version. Comparing the precision and recall rates of Table 3 and Table 4, both a higher negative recall (recall rate for NO) and a higher positive precision (precision rate for YES) were achieved, while the other two values were lower.

As described in Section 3.3, several variables were considered in order to investigate the use of more complex matching methods. The results of these considerations are summarised in Tables 5 and 6. They show that marked differences exist both in terms of the selection criterion

| | Feature (1 if the following is true, 0 otherwise) |
|---|---|
| 1 | There are paths between the entities in the Text |
| 2 | Head verbs are found in the paths |
| 3 | Exact matches of head verbs in the Hypothesis can be found in the head verbs obtained from the paths |
| 4 | Inexact matches of head verbs in the Hypothesis can be found in the head verbs obtained from the paths |
| 5 | The number of paths found between entities in the Text is larger than 3 |
| 6 | The maximum length of the paths is smaller than 3 or the minimum length of the paths is larger than 6 |
| 7 | Two entities making up a path are found in different sentences |

Table 2: Features from the relation comparison

| | | QA | SUM | IR | IE |
|---|---|---|---|---|---|
| Accuracy | | 71.50% | 65.00% | 64.50% | 55.00% |
| Avg. Precision | | 82.67% | 67.76% | 74.50% | 52.44% |
| Precision | YES | 66.67% | 62.76% | 57.63% | 55.08% |
| | NO | 86.00% | 70.91% | 74.39% | 53.85% |
| Recall | YES | 93.46% | 85.05% | 76.40% | 94.50% |
| | NO | 46.24% | 41.94% | 54.95% | 7.69% |
| F1 | YES | 77.82% | 72.22% | 65.70% | 69.59% |
| | NO | 60.14% | 52.70% | 63.21% | 13.46% |

Table 3: Results of the word match version

| | | QA | SUM | IR | IE |
|---|---|---|---|---|---|
| Accuracy | | 83.50% | 66.50% | 70.50% | 53.00% |
| Avg. Precision | | 93.92% | 66.53% | 81.75% | 48.22% |
| Precision | YES | 94.05% | 70.41% | 75.00% | 54.24% |
| | NO | 75.86% | 62.75% | 68.57% | 43.48% |
| Recall | YES | 73.83% | 64.49% | 50.56% | 88.07% |
| | NO | 94.62% | 68.82% | 86.49% | 10.99% |
| F1 | YES | 82.72% | 67.32% | 60.40% | 67.13% |
| | NO | 84.21% | 65.64% | 76.49% | 17.54% |

Table 4: Results of the Maximum Entropy classifier with the shallow features

as well as in terms of the category from which the hypothesis and text are selected. In terms of verb group matching, it appears that the use of paraphrases has a positive impact in all categories apart from the summarisation documents, in which an extreme decrease in performance can be observed. In contrast, the noun group matching seems to have decreased performance when using paraphrases apart from the question answering category, which shows a modest improvement. For subsequent experiments we used the bold matching criterion that is highlighted in the tables.

After changing the simple Relation Matching with the more complex one, we were not able to increase the accuracy of the system as a whole.

| Subtask | Accuracy (%) |
|---|---|
| IR | 64.67 |
| QA | 67.67 |
| SUM | 76.33 |
| IE | 52.00 |
| Average | 58.20 |

Table 7: Official results for Run 1 of our system

## 5.2 Results and Analysis

We submitted three runs to RTE-4, corresponding to the following versions of the system:

**Run 1** Train the Maximum Entropy classifier with only shallow features provided in Sec-

| Criterion | IE | IR | QA | SUM |
|---|---|---|---|---|
| DIRT paraphrases | 0.511 | 0.536 | **0.528** | 0.495 |
| WordNet; ordering irrelevant | 0.513 | **0.565** | 0.505 | **0.616** |
| WordNet; ordering relevant | 0.513 | 0.565 | 0.505 | 0.616 |
| heads only; ordering irrelevant; identical lemmas | 0.513 | 0.565 | 0.509 | 0.603 |
| heads only; ordering relevant; identical lemmas | 0.517 | 0.556 | 0.509 | 0.586 |
| ordering irrelevant; identical lemmas | **0.522** | 0.534 | 0.495 | 0.560 |
| ordering relevant; identical lemmas | 0.522 | 0.534 | 0.493 | 0.560 |

Table 5: Accuracy for different configurations of noun group matching

| Criterion | IE | IR | QA | SUM |
|---|---|---|---|---|
| DIRT paraphrases | **0.501** | **0.505** | **0.514** | 0.111 |
| WordNet; ordering irrelevant | 0.441 | 0.488 | 0.486 | **0.301** |
| WordNet; ordering relevant | 0.441 | 0.492 | 0.486 | 0.301 |
| heads only; ordering irrelevant; identical lemmas | 0.427 | 0.484 | 0.486 | 0.303 |
| heads only; ordering relevant; identical lemmas | 0.427 | 0.484 | 0.486 | 0.318 |
| ordering irrelevant; identical lemmas | 0.433 | 0.449 | 0.484 | 0.384 |
| ordering relevant; identical lemmas | 0.433 | 0.446 | 0.484 | 0.385 |

Table 6: Accuracy for different configurations of verb group matching

| Subtask | Accuracy (%) |
|---|---|
| IR | 64.67 |
| QA | 67.67 |
| SUM | 76.33 |
| IE | 48.00 |
| Average | 57.00 |

Table 8: Official results for Run 2 of our system

| Subtask | Accuracy (%) |
|---|---|
| IR | 57.00 |
| QA | 65.00 |
| SUM | 71.33 |
| IE | 48.00 |
| Average | 52.40 |

Table 9: Official results for Run 3 of our system

tion 3.

**Run 2** Change the shallow Relation Matching feature used in the previous version with the more complex one described in Section 4.

**Run 3** Use only the more complex Relation Matching feature for training and prediction.

Table 7 to Table 9 summarise the results we obtained for the RTE-4 test data. As we can see, the first run achieved the best results out of the three, meaning that a deeper method such as the more complex Relation Matching did not help in our case. However, from Table 10, we may discover that deeper methods helped in improving the prediction on non-entailing pairs.

| Run | Accuracy (%) |
|---|---|
| 1 | 37.80 |
| 2 | 46.00 |
| 3 | 58.60 |

Table 10: Results for the non-entailing pairs in each of the three runs

A drop in overall accuracy when using the Relation Matching feature can be attributed to several reasons:

- **Incorrect NE or noun matches are found:** *Asia* and *Triceratops fossils*.
- **Incorrect paths found between entities:** For a pair in the training data, the RASP tool was unable to find a path between *G8 Summit* and *Sea Island bay*, where the relation between the two should be *took place*.
- **Incorrect matches between the head verbs are found:** When using a paraphrase matching, some words having different meanings may still co-occur very often. For example: *failure* and *success*.

# 6 Conclusion and Future Work

In this project, we analysed the previous three RTE corpus in order to find characteristics of the

non-entailing Text/Hypothesis pairs. As a consequence of the analysis, shallow features have been chosen to detect such non-entailing pairs. In addition, we tested the performance of a more complex version of Relation Matching.

The results obtained reveal that, the complex Relation Matching did not help in improving the system's overall accuracy, which was contrary to what we expected. However, incorporating such a feature into the system increased the accuracy of predicting non-entailing pairs by 20.80%. Out of the three runs, the Shallow Features one performed best, with an accuracy that is 5.80% higher than the Relation Matching run. This indicates that our system can be a fall-back when deep methods fail or are unavailable.

Future work can be directed to further improvement of both Noun/Verb Matching and Relation Matching. Our current Complex Noun/Verb Matching algorithm can be mislead mainly because it identifies a match between two words that do not have the same meaning (e.g. Asia and Triceratops fossils). Although further improvement on Noun/Verb Matching can increase the accuracy of Relation Matching, improving the performance of the dependency relation tool can have a larger positive effect. Current problems in Relation Matching lies mainly in finding paths between entities that have been identified in the Text. If we are able to find more correct paths, especially when two entities are not within the same sentence, then a more accurate comparison between Text and Hypothesis can be carried out.

## References


E. Briscoe, J. Carroll, and R. Watson. The Second Release of the RASP System. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 2006.

James R. Curran and Steven Clark. Investigating GIS and Smoothing for Maximum Entropy Taggers. In *The 11th Meeting of the European Chapter of the Association for Compuational Linguistics (EACL-03)*, pages 91–98, 2003.

Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the 43rd Annual Meeting of the Association of Compuational Linguistics (ACL 2005)*, pages 363–370, 2005.

Claire Grover and Richard Tobin. Rule-based Chunking and Reusability. In *LREC 2006*, pages 873–878, 2006.

Andrew Hickl and Jeremy Bensley. A Discourse Commitment-Based Framework for Recognizing Textual Entailment. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 171–176, 2007.

Andrew Hickl, Jeremy Bensley, John Williams, Kirk Roberts, Bryan Rink, and Ying Shi. Recognising Textual Entailment with LCC's GROUNDHOG System. In *Proceedings of the Second Recognising Textual Entailment Challenge*, pages 80–85, 2006.

Dekang Lin and Patrick Pantel. DIRT - Discovery of Inference Rules from Text. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining (KDD-01)*, pages 323–328, 2001.

Guido Minnen, John Carroll, and Darren Pearce. Robust, Applied Morphological Generation. In *INLG 2000*, pages 201–208, 2000.

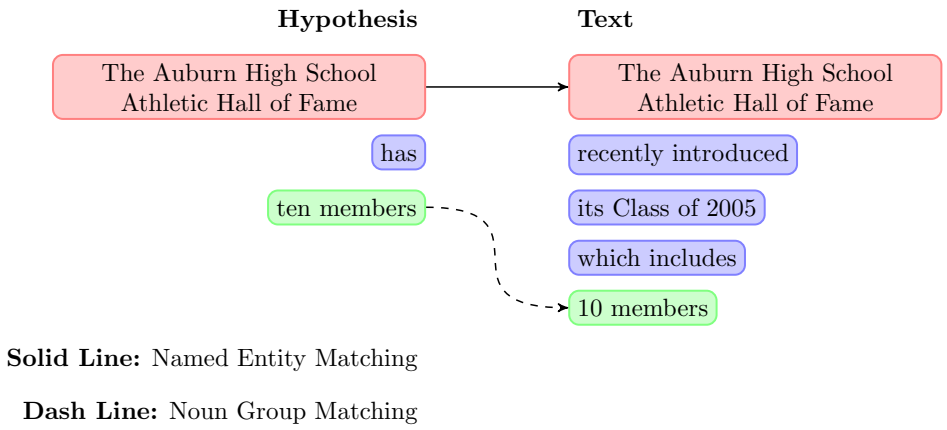


Figure 4: Named Entity and Noun Matching

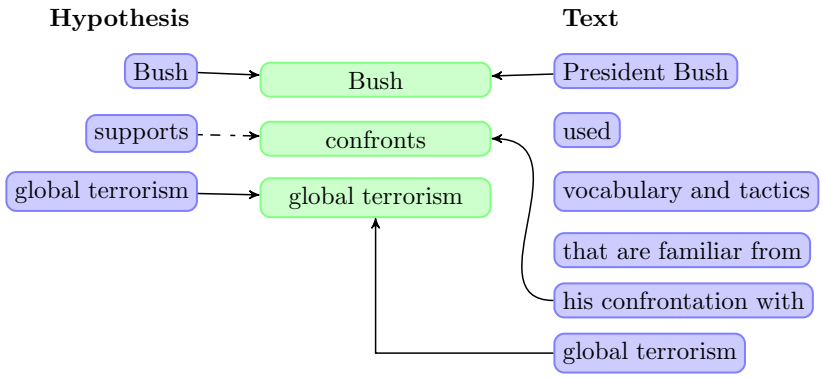


Figure 5: Relation Matching

```
(|ta| |end| |<w id=‘‘w18’’>made:4_VVN</w>| |<w id=‘‘w95’’>sold:16_VVN</w>|)
(|ncsubj| |<w id=‘‘w95’’>sold:16_VVN</w>| |<w id=‘‘w64’’>Yuganskneftegaz:13_NP1</w>| _)
(|ncmod| _ |<w id=‘‘w95’’>sold:16_VVN</w>| |<w id=‘‘w84’’>originally:15_RR</w>|)
(|aux| |<w id=‘‘w95’’>sold:16_VVN</w>| |<w id=‘‘w80’’>was:14_VBDZ</w>|)
(|passive| |<w id=‘‘w95’’>sold:16_VVN</w>|)
(|iobj| |<w id=‘‘w95’’>sold:16_VVN</w>| |<w id=‘‘w100’’>for:17_IF</w>|)
(|dobj| |<w id=‘‘w100’’>for:17_IF</w>| |<w id=‘‘e2’’>US$_9.4_billion:18_NN1</w>|)
```

Figure 6: Output from RASP

```
<rel type=‘‘ta’’ arg1=‘‘w18’’ arg2=‘‘w95’’/>
<rel type=‘‘ncsubj’’ arg1=‘‘w95’’ arg2=‘‘w64’’/>
<rel type=‘‘ncmod’’ arg1=‘‘w95’’ arg2=‘‘w84’’/>
<rel type=‘‘aux’’ arg1=‘‘w95’’ arg2=‘‘w80’’/>
<rel type=‘‘iobj’’ arg1=‘‘w95’’ arg2=‘‘w100’’/>
<rel type=‘‘dobj’’ arg1=‘‘w100’’ arg2=‘‘e2’’/>
```

Figure 7: Relations in XML format extracted from RASP output