1 Introduction

In the problem of recognizing textual entailment, the goal is to decide, given a text $T$ and a hypothesis $H$ expressed in a natural language, whether a human reasoner would call $H$ a consequence of $T$. The following example is No. 115 in the collection of problems proposed as the Second PASCAL Recognizing Textual Entailment Challenge [Bar-Haim et al., 2006]:

$T$: The World Bank has also been criticized for its role in financing projects that have been detrimental to human rights and the natural environment.

$H$: The World Bank is criticized for its activities.

Expected answer: Yes.

Recognizing textual entailment is a special case of a more general and practically important problem, textual query answering.

To recognize the fact that $H$ is “entailed” by $T$, we often need to use some background commonsense knowledge. For instance, in the example above it is essential that financing is an activity.

The approach to recognizing textual entailment employed by Bos and Market in [Bos and Markert, 2005] and implemented in the system Nutcracker\footnote{http://svn.ask.it.usyd.edu.au/trac/candc/wiki/nutcracker} can be summarized as follows:

(i) $T$ and $H$ are represented by first-order formulas,

(ii) potentially relevant background knowledge is identified and expressed by a first-order formula $BK$,\footnote{http://svn.ask.it.usyd.edu.au/trac/candc/wiki/nutcracker}
an automated reasoning system is used to check whether the implication
\[ T \land BK \rightarrow H \]
is logically valid.

## 2 CSCI 4980/8986 Approach

In the class project we will build an RTE system that will follow the basic principles exemplified by the system Nutcracker. The architecture of the system follows:

- **NL input T and H**
- **Syntactic Parser / Semantic Shallow Analyzer**
- **Background Knowledge Collector**
- **ASP-based Automated Reasoner**

The Syntactic Parser/Semantic Shallow Analyzer component of the system is responsible for building *machine readable* representation of the sentence that contains information about some syntactic and semantic structure of the sentence. The task of Background Knowledge Collector component to extract background commonsense knowledge relevant to the input T and H pair using such lexical resources as WordNet, FrameNet, ConceptNet, . . . .

The decision on whether entailment takes place or not is carried out by by the Answer Set Programming (ASP) reasoner that relies on information from both the input pair and the relevant background knowledge. Answer Set Programming\(^2\) is a form of declarative programming oriented towards difficult (primarily NP-hard) search problems. It is based on the answer

\[^2\]http://en.wikipedia.org/wiki/Answer_set_programming
set (stable model) semantics of logic programming. ASP includes all applications of answer sets to knowledge representation [Gelfond, 2008] and the use of Prolog-style query evaluation for solving problems arising in these applications. In ASP, search problems are reduced to computing answer sets, and answer set solvers – programs for generating stable models are used to perform search.

3 Assignment 1: Basic RTE system

Organizational:
Two Teams:

1. Jonathan Guerrero (jguerrero@unomaha.edu),
   Songqing Liu (songqinliu@unomaha.edu),
   Aruna Weerakoon (rweerakoon@unomaha.edu)

2. April Corbet (acorbet@unomaha.edu),
   Mohammad DwaiKat (mdwaiKat@unomaha.edu),
   Nathan Schulte (nmschulte@unomaha.edu)

Due: September 12th.
All of the Assignments and Project development to be carried out at
loki.ist.unomaha.edu
server. Each one of you has an account on the system with the same credentials as your blackboard account. There is a directory
/team/nlu
that is meant as the space for the development. All of the files should have the group
nlu
You can change the group of files using the command
chgrp nlu <filename>

Web site http://loki.ist.unomaha.edu/ is meant to assist you with the questions on loki.
loki@pki.nebraska.edu is a mailing list for the administrators and engineers responsible for loki.
Assignment:

1. Develop a basic RTE system that
   (i) performs syntactic/shallow semantic analysis,
   (ii) carries out the decision on the entailment based on information gained from (i).

2. Apply the developed basic RTE system to the dataset available at the Second Recognizing Textual Entailment Challenge\(^3\) (RTE-2):
   http://nlp.stanford.edu/projects/infer/data/byformat/rte/RTE2_dev.xml

3. Analyze the results and write a report that provides the answers to the following:
   - Collect the data on how many T and H pairs from RTE-2 the basic RTE system establishes entailment.
   - Pick any 5 pairs for which it is known that the entailment is the case. Present the output of syntactic/shallow semantic analysis performed by your system.
   - For these 5 pairs perform a detailed analysis why the entailment is indeed the case. How one needs to augment the information given by the syntactic/shallow semantic analyzer in order for the system to carry out the right conclusion.

Input: to the basic RTE system: names of two files with plain text where the former contains T and the latter contains H.

Output: YES or NO.

Team 1: The backbone of syntactic parser/shallow semantic analyzer of this team will be Stanford dependencies parser available at
Consider T to contain the following text:

T: Bills on ports and immigration were submitted by Senator Brownback, Republican of Kansas

The Stanford dependency parser will produce the following analysis of the text:

\(^3\)http://pascallin.ecs.soton.ac.uk/Challenges/RTE2
We denote this analysis by \( \text{syn}(T) \).

We say that the text \( T \) entails text \( \mathcal{H} \), if

\[
\text{syn}(T) \supseteq \text{syn}(\mathcal{H}).
\]

(1)

As a result the basic RTE system outputs \( \text{YES} \) only in case if (2) holds for the given \( T \) and \( \mathcal{H} \).

For instance, for \( \mathcal{H} \)

\( \mathcal{H}: \) Bills were submitted by Brownback,

\( \text{syn}(\mathcal{H}) \) has the form

\[
\text{nsubjpass(submitted, Bills)}
\]
\[
\text{auxpass(submitted, were)}
\]
\[
\text{agent(submitted, Brownback)}
\]

and thus given \( T \) and \( \mathcal{H} \) the basic RTE system should output \( \text{YES} \).

**Team 2:** The backbone of syntactic parser/shallow semantic analyzer of this team will be *Mate-tools* available at

http://code.google.com/p/mate-tools/

The tool contains dependency parser and semantic role labeler. The idea is to use the semantic role labeler as it will provide more data. The entailment is understood in a similar manner as in the case of Team 1.

**References**
