Lexical Based Text Entailment System for Main Task of RTE6

Debarghya Majumdar

Computer Science and Engineering
IIT Bombay Powai, India
deb@cse.iitb.ac.in
majumdar.debarghya@gmail.com

Pushpak Bhattacharyya

Computer Science and Engineering
IIT Bombay Powai, India
pb@cse.iitb.ac.in

Abstract

Our system describes a simple lexical based system which detects entailment based on word overlap between the Text and Hypothesis. The system is mainly designed to incorporate various kind of co-reference that occurs within a document and how they take an active part in the event of Text Entailment.

1. Introduction

The basic definition of Text entailment says a piece of Text T entails another piece of Text H if a person reading T can infer whenever T is true H is also True. However since the person can use background knowledge while inferring so formally we can say that the Text T entails hypothesis H if T and background knowledge entails H but the background knowledge alone does not entail H.

Now the main task of RTE6 is following. Given a corpus and a set of "candidate" sentences retrieved by Lucene from that corpus, RTE systems are required to identify all the sentences from among the candidate sentences that entail a given Hypothesis. The RTE-6 Main Task is based on the TAC Update Summarization Task in which each topic contains two sets of documents ("A" and "B"), where all the "A" documents chronologically precede all the "B" documents.

Now in this scenario we have to find all the sentences in a document which entails a given hypothesis. So the challenge in the Text Entailment problem has changed now. Now a sentence may entail a hypothesis with the aid of other sentences in the document.

2. System Description

We use a basic lexical entailment model for detecting the event of Text Entailment. The T-H pair are fed into a Stanford named entity recognizer to detect the named-entities. Now the T-H pair is fed into a matching module.

Now we remove the stop words from both the T-H pair as they give a wrong impression to the matching between the T-H pair. The matching module performs various kinds of matching using different resources as we will describe later.

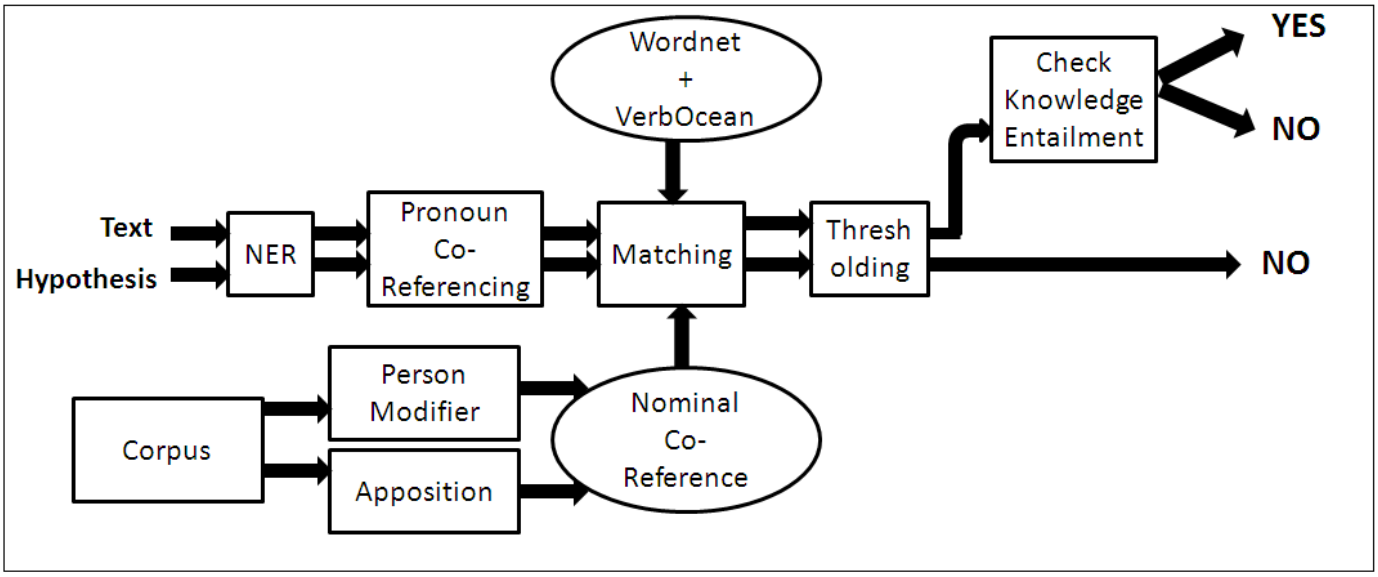

Figure 1 System Design

In this section we will discuss the different module of our system.

2.1 Acronym Finder

Our system did not use any external ACRONYM database but it uses the corpus to generate a small ACRONYM database.

Text: *Answering questions in parliament, Ahern told MPs that the meetings with Adams, whose party is the political wing of the* ***IRA****, Northern Ireland's main Catholic paramilitary group, had merely been to maintain dialogue.*

Hypothesis: *The* ***Irish Republican Army*** *is a Catholic paramilitary group.*

"The Irish Republican Army" in the hypothesis will match with the *"IRA"* in the Text. Since some document contained the information that *"IRA"* is the Acronym of *"Irish Republican Army"*so we were able to detect it by consulting the ACRONYM database prepared during the preprocessing stage.

2.2 Wordnet (Fellbaum, 1998)

We used path length between two words as distance measure between two words. A hypothesis word matches with a Text word if the Wordnet path length between the corresponding concept is lesser than two.

2.3 VerbOcean (Chklovski & Pantel, 2004)

We use VerbOcean to find whether entailment relation hold between two verb. VerbOcean contains relation like "*similar to*" and "*happens before*" which contains useful information about entailment. The relation "happens before" is directional while "*similar to*" is not. So if a verb X is in "*happens before*" relation to a verb Y then if Y is present in Text and X is present in Hypothesis we denote a match.

2.4 Named Entity Matching

We used Stanford named-entity recognizer to detect named-entity in both Text and hypothesis. A named entity in hypothesis can only match with a named entity in text and a single named entity mismatch generally leads to non-entailment.

If we do not find a match for a named entity of Text we search the ACRONYM database created by the Acronym finder. If the named entity has an ACRONYM then we also search for the corresponding ACRONYM in the Text.

However if a named entity is not matched we search the database created by Person Modifier and Apposition. This database contains phrases which describes the named entity. So if the Text contains such phrases we can find a match between the phrase in the Text and the named entity in the Hypothesis.

2.5 Number Matching

Since none of the resource described above contains information about number we developed a number matching module. We noticed in the RTE6 development set that lot of numbers contains numeric expression like *"at least", "near to"*. So our module tries to search whether any of the set of predefined pattern is present before the number in both the Text and the Hypothesis. If there is one, we normalize the number based on that.

Consider the example in RTE6.

***Text:* <u>*At least 35 people*</u> *were killed and 125 injured in three explosions targeting tourists in Egypt's Sinai desert region late Thursday, an Egyptian police source said.*

Hypothesis: <u>***At least 30 people***</u> *were killed in the blasts.*

So the system stores a flag which indicates that any number greater than 30 in the text will be matched with the numeric expression of the hypothesis. So the number 35 in Text matches with the expression in the Text. It is clear without normalization there would have been a number mismatch however after normalization the two numeric expressions will match.

2.6 Checking Knowledge Entailment

Since we have used different kind of co-reference and stored information about person in database during the preprocessing stage we have to ensure that the hypothesis has some part to play in the entailment decision i.e. n the knowledge alone does not entail the hypothesis.

Consider the example from RTE6.

Text: <u>***L. Dennis Kozlowski***</u> *wants to be clear: The $6,000 shower curtain wasn't*

Hypothesis: <u>***L. Dennis Kozlowski***</u> *is the* <u>***former chief executive of Tyco International Ltd.***</u>

Now our apposition module detected that*" L. Dennis Kozlowski"* and *"executive of Tyco International Ltd."* are in apposition relation. So while matching *" L. Dennis Kozlowski" of the Text will match both " L. Dennis Kozlowski"* and *"executive of Tyco International Ltd."* which may lead to wrong entailment decision. So we check whether most part of the hypothesis is matched from the Text or some other information obtained from the document (like co-reference). If most of the matching is not obtained from the Text then it is a case of non-entailment like in the case of the above example.

2.7 Co-reference

2.7.1 Pronominal Co-reference

For pronoun co-reference we used a tool called LingPipe. Generally pronoun co-reference is done by a noun which is in the same sentence o within the previous two three sentence. So while feeding the T-H pair we are also feeding the previous three sentences of the corpus to the tool. However this part was not incorporated when we tested the system as it was not complete.

2.7.2 Nominal Co-reference

Apposition

We developed a tool which will find two noun phrases which are in apposition relation in a given sentence. Now two noun phrases in the document are in apposition relation wecan substitute one of them by the other without changing the semantics of the sentence. So apposition can lead to nominal co-reference.

Consider the example in RTE6

Text: *The trial against* ***a millionaire*** *and a mill worker charged with multiple counts of conspiracy and first-degree murder hinges on the testimony of three star witnesses and contradicting bombing experts.*

Hypothesis: ***Ajaib Singh Bagri*** *had faced charges of murder and conspiracy.*

The document which contains the Text contains the line below

Ripudaman Singh Malik, ***a 57-year-old millionaire,*** *and Ajaib Singh Bagri, 55, are charged with planting bombs that exploded June 23, 1985, aboard Flight 182, killing 329 people, and 53 minutes earlier at Tokyo's Narita Airport, killing two baggage handlers.*

From the line above our module extracted that "*Ripudaman Singh Malik*" is a "*57-year old millionaire*" which leads to the entailment decision.

Time of Document

The year in which the document is written is specified in top of the document. Since all the sentences in the document are only true for the time at which the document was written so we have to normalize all temporal expression with respect to the time of the document. We studied the development set and found temporal expression like "*last year*", "*previous year*", "*25 year-old*" etc. We detect such pattern and thus generate the year which the Text specifies.

Consider the Text hypothesis pair in the development set of RTE6.

Text: *President Bush campaigned* ***last year*** *in favor of renewing the Patriot Act, and Attorney General Alberto Gonzales has indicated he doesn't favor any changes except, perhaps, to increase the government's powers in a few instances.*

Hypothesis*: The Patriot Act comes up for renewal in* ***2005****.*

The year in which the document was written was 2006. So in the Text the *"last year"* refers to *"2005"* which is indicated in the hypothesis.

Person Name Modifier

When a noun phrase is present before a name of the person then generally the noun phrase refers to the person.

So if this sentence is present in a document

"Al-Jazeera reiterates its rejection and condemnation of all forms of violence targeting journalists, and demands the release of the US journalist Jill Carroll," the station said.

From this we can tell that *"**Jill Carroll**"* is an "***US journalist***".

Text: *Arabic television Al-Jazeera said Tuesday the kidnappers of a **US woman journalist** abducted in Baghdad had threatened to kill her if female prisoners in Iraq were not freed within 72 hours.*

Hypothesis*:* ***Jill Carroll*** *was abducted in Iraq.*

So if we have tested for entailment without performing this co-reference it would have resulted in non-entailment but after co-reference clearly we can detect entailment.

3. Result

3.1 Performance on RTE6 dataset

We used three different thresholds for the three run. The first run used the threshold which was giving the best result for the development set. The second run has more lenient threshold and was aimed at higher recall value. In the third threshold we used a stricter threshold valued aimed for higher precision. The result for the three run is shown in the table below.

	Precision	**Recall**	**F-Score**
Run1	55.98	34.18	42.44
Run2	53.43	42.86	47.56
Run3	71.61	30.16	42.69

Table 1: Performance of Lexical Based System in RTE6

Clearly we can see run2 out performs the other two run in terms of F-Score value as both the precision and recall value of it are reasonable. It is also clear that for lexical based system if the precision value increases the recall value falls. This is because the acceptance of T-H pair after matching depends on threshold and if we increase the threshold very few T-H pair will be matched. So some of the entailment cases will be classified as non-entailment as the matching falls below the threshold and hence the recall value will decrease. In general the number of matched for entailment instance is higher than that of non-entailment instances. So the number of non-entailment instance that will have matching greater than the higher threshold will be very small. So the precision value will increase. So for Run3 even though we achieved a very high precision value we still have a low F-Score due to low recall value.

3.2 Ablation Test

We used two lexical resources the Wordnet and the VerbOcean. To check the role each resource plays an ablation test was performed on them. The result of the ablation test is given in table 2.

	Run1			Run2			Run3		
	Precision	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**	**Precision**	**Recall**	**F-Score**
Wordnet	-13.2	11.85	8.68	-7.85	13.55	7.90	-7.87	10.90	11.43
VerbOcean	0.14	2.33	1.87	-0.14	1.59	0.94	-0.49	2.54	2.50

Table 2 Ablation Test Result of Wordnet and VerbOcean

Clearly the values of the precision, recall and F-Score suggest Wordnet has a greater impact on the system compared to VerbOcean. The negative precision value for Wordnet indicates that Wordnet has helped in many matching even in case of non-entailment instance. So to counter this we will require a contradiction detection module in future. However VerbOcean does not decrease the precision value too much. The improvement in recall value is substantial compared to the loss of precision value. This is due to the fact that in a T-H pair more words are matched using Wordnet compared to VerbOcean. But still the overall impact of VerbOcean suggest it helps in matching.

4. Future Work and Conclusion

Our system only performs Text Entailment at the lexical level. So, we have to use other lexical resource like WIKEPEDIA and TEASE in future. We have incorporated only two type of nominal co-reference. The ablation test result shows many of the matches have led to the downfall of the precision value so a separate contradiction detection module is needed to enhance the precision.

While performing entailment we saw our Text Entailment System has low accuracy for longer hypothesis. The problem often is that for longer hypothesis all the words in the hypothesis might not possess important information. So even if such word is not matched entailment can still hold. So finding out the smallest hypothesis which posses the same information as other longer hypothesis is indeed a challenging problem and if solved can help different Entailment Systems.

5. Bibliography

Alias-i. 2008. LingPipe 4.0.1. http://alias-i.com/lingpipe (accessed October 1, 2008)

Bentivogli, L., Magnini, B., Dagan, I., Dang, H. T., & Giampiccolo, D. (2009). The Fifth PASCAL Recognizing Textual Entailment Challenge. *TAC.*

Chklovski, T., & Pantel, P. (2004). VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. *ACL*, (pp. 33-40).

Dagan, I., Roth, D., & Zanzotto, F. M. (n.d.). *Tutorial 3: Textual Entailment ACL Prague.* Retrieved from http://www.aclweb.org: http://www.aclweb.org/mirror/acl2007/tutorials/index.php/t3/index.html

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database.* Cambridge, MA: MIT Press.

Giampiccolo, D., Magnini, B., Dagan, I., & Dolan, B. (2007). The third PASCAL recognizing textual entailment challenge. *In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, (pp. 1-9). Prague.

Szpektor, I., & Dagan, I. (2008). Learning entailment rules for unary templates. *In Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, (pp. 849-856). Manchester.