**Available Online at www.ijcsmc.com**

**International Journal of Computer Science and Mobile Computing**

**A Monthly Journal of Computer Science and Information Technology**

**ISSN 2320–088X**

***IJCSMC, Vol. 3, Issue. 6, June 2014, pg.528 – 534***

**RESEARCH ARTICLE**

# Text Grouping using Textual Entailment

**Partha Pakray**

Department of Computer & Information Science (IDI)
Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
www.parthapakray.com
partha.pakray@idi.ntnu.no

***Abstract*— *Textual Entailment is an important field in Natural Language Processing domain. Given two texts called T (Text) and H (Hypothesis), the textual entailment recognition is the task of deciding whether the meaning of H can be logically inferred from that of T. A Textual Entailment (TE) system has developed and this system has tested on various entailment standard datasets. This TE will apply to different texts then the TE system will group them into different single group. A corpus has created for this experiment that has total 10 groups which contains 3540 sentences. F-score of the textual entailment system is 61% and will detect 8 groups correctly from 10 groups.***

***Keywords*— *Natural Language Processing, Textual Entailment, reverb, Support Vector Machine***

## I. INTRODUCTION

Many efforts have devoted by the Natural Language Processing (NLP) community to develop advanced methodologies in Textual Entailment (TE), which is considered as a core NLP task. Various international conferences and several evaluation track competitions on Textual Entailment have been held, notably at *PASCAL-Pattern Analysis, Statistical Modelling and Computational Learning*[1], *Text Analysis Conferences (TAC)*[2] organized by the United States *National Institute of Standards and Technology* (NIST), *Evaluation Exercises on Semantic Evaluation* (SemEval)[3], *National Institute of Informatics Test Collection for Information Retrieval System* (NTCIR)[4] since 2005. Textual entailment can be more formally defined [1] as

- ✓ *A text T entails a hypothesis H, if H is true in every circumstance in which T is true.*
- ✓ *A text T entails a hypothesis H if, typically, a human reading T would infer that H is most likely true.*

For example, the text T = "*John's assassin is in jail*" entails the hypothesis H = "*John is dead*"; indeed, if there exists one's assassin, then this person is dead. Similarly, T = "*Mary lives in France*" entails H = "*Mary lives in Europe*". On the other hand, T = "*Mary lives in Europe*" does not entail H = "*Mary lives in US*".

Main focus of this experiment is that Text Grouping (i.e. clustering) can do by Textual Entailment. For this experiment own developed TE system used that already developed previously and participated various Recognising Textual Entailment (RTE) Challenges and tested on RTE datasets. This TE system has successfully applied to Question Answering (QA) domain and participated QA track (QA4MRE) at Conference and Labs of

[1] http://pascallin.ecs.soton.ac.uk/Challenges/
[2] http://www.nist.gov/tac/tracks/index.html
[3] http://semeval2.fbk.eu/semeval2.php
[4] http://research.nii.ac.jp/ntcir/ntcir-9/

the Evaluation Forum (CLEF) and got best results. And this TE system also applied to Summarization domain to evaluate summary. Now in this experiment TE system has applied for text grouping. In future this TE system will use for event based clustering. Related Works has described in Section II. The TE system has described in Section III. Own developed corpus has reported in Section IV. In Section V system performance and evaluation result set has described.

## II. Related Works

In the various Textual Entailment Challenges, several methods are applied to tackle textual entailment problems. Most of these systems use some sort of lexical matching, e.g., n-gram, word similarity etc. and even simple word overlap. A number of systems represent the texts as parse trees (e.g., syntactic or dependency trees) before the actual task. Some of the systems use semantic relation (e.g., logical inference, Semantic Role Labelling) for solving the text and hypothesis entailment problem. The system [2] proposed a new architecture for textual inference in which finding a good alignment is separated from evaluating entailment. The Emory system [3] used a supervised machine learning approach to train a classifier over a variety of lexical, syntactic, and semantic metrics. The system [4] used string similarity measures applied to shallow abstractions of the input sentences, and a Maximum Entropy classifier to learn how to combine the resulting features. NutCracker[5] [5] system is based on logical representation and theorem proving. EDITS[6] [6] is based on a distance-based system, which used only lexical knowledge resources. BIUTEE[7] [7] is an open-source system, which recognizes textual entailment. It has used various types of knowledge resources. This system decides whether the text entails the hypothesis by observing the quality of this sequence.

## III. System Description: Textual Entailment

A two-way automatic textual entailment (TE) recognition system that uses lexical, syntactic, and semantic features has been described in this section. The system architecture has been shown in Figure 1. The TE system has used the Support Vector Machine technique that uses thirty-one features for training purpose. In lexical module there are eighteen features and eleven features from syntactic module, one feature by using reVerb and one feature from semantic module.

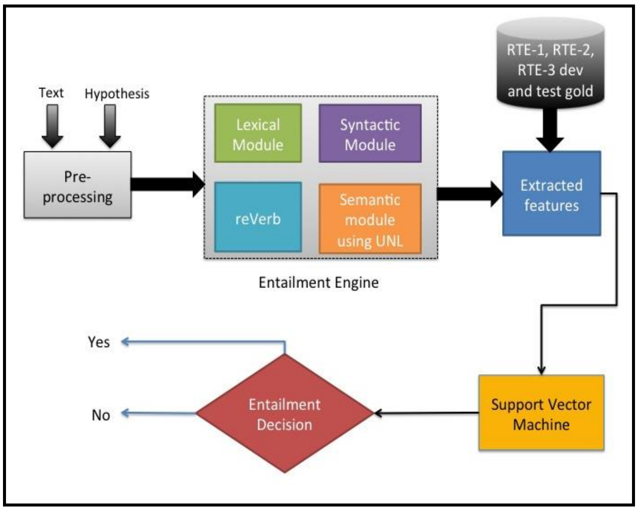

Fig. 1 Textual Entailment System Architecture

### A. *Lexical Module*

In this module six lexical comparisons and twelve lexical distance comparisons between text and hypothesis has used. Six lexical comparisons are WordNet [8] based unigram match, bigram match, longest common sub-sequence, skip-gram, stemming and named entity matching. It has calculated weight from each of these six comparisons in equation (1).

[5] http://svn.ask.it.usyd.edu.au/trac/candc/wiki/nutcracker

[6] http://edits.fbk.eu/

[7] http://u.cs.biu.ac.il/~nlp/downloads/biutee/protected-biutee.html

$$weight = \frac{\sum number - of - common - tokens - between - text - and - hypothesis}{\sum number - of - tokens - in - hypothesis} \qquad (1)$$

WordNet [8] is one of most important resource for lexical analysis. WordNet provides different kinds of relations, such as for **Nouns** (hypernyms, hyponyms, coordinate terms, holonym, meronym), **Verbs** (hypernym, troponym, entailment, coordinate terms), **Adjectives** (related nouns similar to participle of verb) and **Adverbs** (root adjectives). The WordNet 2.1[8] has been used for WordNet based unigram match (synset match) and stemming. The API for WordNet Searching (JAWS)[9] provides Java applications with the ability to retrieve data from the WordNet 2.1 database.

For Named entity detection it has used Text Tokenisation Toolkit (LT-TTT2) [9]. The LT-TTT2 named entity component has been used. The named entity component which recognizes and marks up the following kinds of named entities: numex (e.g., sums of money and percentages), timex (e.g., dates and times), enamex (e.g., persons, organizations and locations) and role (e.g., president).

For lexical distance measure, it has used features of Vector Space Measures (Euclidean distance, Block distance, Minkowsky distance, Cosine similarity, Matching Coefficient), Set-based Similarities (Dice, Jaccard, Overlap, Harmonic), Edit Distance Measures (Levenshtein distance, Smith-Waterman distance, Jaro Distance). Lexical distance measurement has used the libraries SimMetrics[10], SimPack[11] and SecondString[12]. SimMetrics is a Similarity Metric Library, e.g., from edit distance (Levenshtein, Gotoh, Jaro etc) to other metrics, (e.g Soundex, Chapman). SimPack is a similarity between concepts (complex objects) in ontologies. All measures of SimPack are implemented in Java-based generic similarity framework. SecondString is an open-source Java-based package of approximate string-matching techniques.

### B. *Syntactic Module*

The syntactic module compares the dependency relations in both hypothesis and text. The system extracts syntactic structures from the text-hypothesis pairs using Combinatory Categorial Grammar (C&C CCG) Parser and Stanford Parser and compares the corresponding structures to determine if the entailment relation is established. Two different systems have been implemented one system used Stanford Parser output and another system used C&C CCG Parser. The system accepts pairs of text snippets (text and hypothesis) at the input and gives score for each comparison. Some of the important comparisons on the dependency structures of the text and the hypothesis are Subject-subject comparison, WordNet Based Subject-Verb Comparison, Subject-Subject Comparison, Object-Verb Comparison, WordNet Based Object-Verb Comparison, Cross Subject-Object Comparison, Number Comparison, Noun Comparison, Prepositional Phrase Comparison, Determiner Comparison and other relation Comparison.

### C. *reVerb Module*

ReVerb[13] is a tool, which extracts binary relationships from English sentences. The extraction format has shown in Table I.

TABLE I
AN EXAMPLE BY REVERB TOOL

| | |
|---|---|
| **Extraction Format** | argument1 relation argument2 |
| **Example** | A person is playing a guitar |
| **reVerb Extracts** | arg1= {A person} rel = {is playing} arg2 = {a guitar} |

The system parsed the text and the hypothesis by reverb tool. Each of the relations compares between text and hypothesis and calculates a score for each pair.

### D. *Semantic Module*

The semantic module based on the Universal Networking Language (UNL) [10]. UNL is an artificial language that can be used as a pivot language in Machine Translation systems or as a knowledge representation language

[8] http://wordnetcode.princeton.edu/2.1/

[9] http://lyle.smu.edu/~tspell/jaws/index.html?utm_source=twitterfeed&utm_medium=twitter

[10] http://sourceforge.net/projects/simmetrics/

[11] https://files.ifi.uzh.ch/ddis/oldweb/ddis/research/simpack/index.html

[12] http://sourceforge.net/projects/secondstring/

[13] http://reverb.cs.washington.edu/

in Information Retrieval applications. The UNL can express information or knowledge in semantic network form with hyper-nodes. The UNL is like a natural language for computers to represent and process human knowledge. UNL system has En-converter and De-converter module. The process of representing natural language sentences in UNL graphs is called En-converting and the process of generating natural language sentences out of UNL graphs is called De-converting. An En-Converter is a language independent parser, which provides a framework for morphological, syntactic, and semantic analysis synchronously. The En-Converter is based on a word dictionary and a set of en-conversion grammar rules. It analyses sentences according to the en-conversion rules. A De-Converter is a language independent generator, which provides a framework for syntactic and morphological generation synchronously.
An example UNL relation for a sentence "*Pfizer is accused of murdering 11 children*" is shown in Figure 2.

```
[S:00]
{org:en} Pfizer is accused of murdering 11 children {/org}
{unl}
obj(accuse(icl>do,equ>charge,cob>abstract_thing,agt>person,obj>person).@entry
.@present,pfizer.@topic)
qua:01(child(icl>juvenile>thing).@pl,11)
obj:01(murder(icl>kill>do,agt>thing,obj>living_thing).@entry,child(icl>juvenile
>thing).@pl)
cob(accuse(icl>do,equ>charge,cob>abstract_thing,agt>person,obj>person).@entr
y.@present,:01)
{/unl}
[/S]
```

Fig. 2 UNL Example

The system converts the text and the hypothesis into UNL relations by En-Converter. Then it compares the UNL relations in both the text and the hypothesis and gives a score for each comparison.

### *E. Extracted Features*

The features for Machine Learning are listed in Table II:

TABLE II
FEATURES FOR MACHINE LEARNING

| Name of Features | No of features |
|---|---|
| Lexical Module | 18 |
| Syntactic Module | 11 |
| reVerb Module | 1 |
| Semantic Module | 1 |

### *F. Support Vector Machine*

Recognizing Textual Entailment task is a text classification problem, i.e., two-way or three-way classification problem. Machine learning methods can be used to solve the textual entailment problem. The main advantage of machine learning based approaches to textual entailment is that multiple entailment features can be easily combined to learn an entailment classifier. Several Machine Learning methods have been widely used in RTE, such as Support Vector Machines (SVMs), Decision Trees (DTs), Maximum Entropy (ME), Naive Bayes classifier etc. In machine learning, support vector machines (SVMs)[14] are supervised learning models used for classification and regression analysis. Associated learning algorithms analyse data and recognize patterns. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes form the output, making it a non-probabilistic binary linear classifier.
The SVM based our Textual Entailment system has used the following data sets: RTE-1 development and RTE-1 annotated test set, RTE-2 development set and RTE-2 annotated test set, RTE-3 development set and RTE-3 annotated test set to deal with the two-way classification task. The system has used the LIBSVM -- A Library for Support Vector Machines[15] for the classifier to learn from this data set.

[14] http://en.wikipedia.org/wiki/Support_vector_machine
[15] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

## IV. CORPUS CREATION

For this experiment total 3540 sentences are created from various news websites. Then it has grouped into ten groups by manual annotators. The groups have shown in Table III.

TABLE III
GROUP NAMES

| Group Name |
|---|
| Political |
| Visa Fraud |
| Price Hike |
| Flood |
| Earth quake |
| Cricket |
| Football |
| Bollywood |
| Music |
| Winter Strom |

## V. COMPARISON AND EVALUATION BETWEEN GOLD STANDARD (MANUAL ANNOTATION) AND TEXTUAL ENTAILMENT SYSTEM

The dataset contains total 3540 sentences. The Textual Entailment system runs on this dataset. This TE system compares each sentence (i.e. called “Text”) with other sentence (i.e. “Hypothesis”) and system gives each pair a score with entailment decision. System has detected 8 different classes out of 10, has shown in Table IV.

TABLE IV
RESULT FOR GROUP DETECTION

| Description | Total numbers |
|---|---|
| No of groups by Manual Annotation | 10 |
| No of groups detected by Textual entailment System | 8 |

So, it means no political related sentence can entail with flood related sentence. The sentences from flood groups shown in Table V. Manual graph has shown in Figure 3 and our TE system output graph has shown in Figure 4.

TABLE V

Example

| Example | Name |
|---|---|
| The Kedarnath valley, along with and other parts of the state of Uttarakhand, was hit with unprecedented flash floods on 16 and 17 June 2013 almost after 80 years. | **A** |
| It is almost after 80 years, such a severe flash flood hits the Kedarnath valley and many other parts of Uttarakhand. | **B** |
| The Kedarnath valley, along with other parts of the state of Uttarakhand, was hit with unprecedented flash floods on 16 and 17 June 2013. | **C** |
| Hundreds died and thousands were left homeless in the June 16-17, 2013 flash floods in the Kedarnath Valley of Uttarakhand, the worst disaster in the state in decades. | **D** |

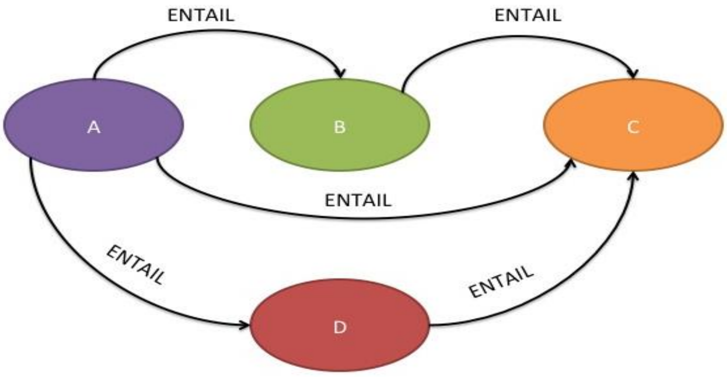

**Fig. 3.** Entailment by Manual Annotation

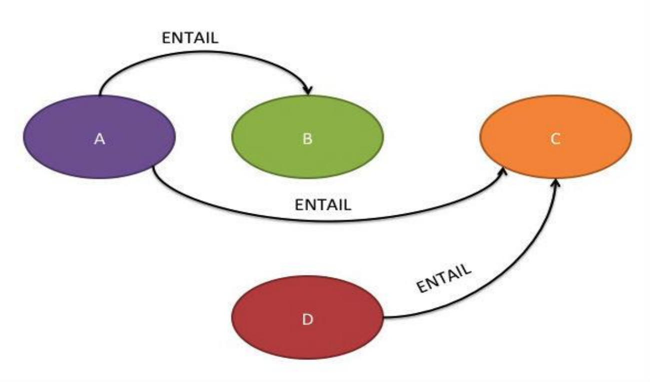

**Fig. 4.** Entailment by our TE System

The system result has shown in Table VI.

TABLE VI
EVALUATION RESULT FOR ENTAILMENT DETECTION

| **Details** | **Result** |
|---|---|
| Total No of Sentence | 3540 |
| Total No of Pairs “ENTAIL” by Manual Annotation | 278 |
| Total No of Pairs “ENTAIL” by Our TE System | 310 |
| Total No of Pairs “ENTAIL” Correctly by Our TE System | 182 |
| Precision | 0.58 |
| Recall | 0.65 |
| F-Score | 0.61 |

## VI. Discussion

In this paper own developed textual entailment system has applied to different text and TE system take different texts as input and output as different single group. Our manual annotation was total 10 groups and TE system detects 8 groups successfully. Next target is that we will generate the large number or corpus and try to apply for clustering.

## Acknowledgment

I acknowledge the support from Department of Computer and Information Science (IDI), Norwegian University of Science and Technology (NTNU), Norway and also support from “ALAIN BENSOUSSAN” FELLOWSHIP PROGRAMME, 2012-2013.

## References

[1] Dagan, I., and Glickman, O. 2004. Probabilistic textual entailment: generic applied modeling of language variability, In PASCAL Workshop on Learning Methods for Text Understanding and Mining, Grenoble, France

[2] B. MacCartney, T. Grenager, M. C. de Marneffe, D. Cer, and C. D. Manning. 2006. Learning to Recognize Features of Valid Textual Entailments. In Proceedings of NAACL-06, New York.

[3] Eugene Agichtein, Walt Askew, and Yandong Liu. Combining lexical, syntactic, and semantic evidence for textual entailment classification. In Text Analysis Conference, 2008.

[4] Malakasiotis, P. (2009). Paraphrase recognition using machine learning to combine similarity mea- sures. In Proc. of the 47th Annual Meeting of ACL and the 4th Int. Joint Conf. on Nat. Lang. Processing of AFNLP, Singapore.

[5] Johan Bos and Katja Markert. 2005. Recognising textual entailment with logical inference. In Proceedings of EMNLP.

[6] Milen Kouylekov and Matteo Negri. 2010. An open- source package for recognizing textual entailment. In Proceedings of ACL Demo.

[7] Asher Stern, Ido Dagan. BIUTEE: A Modular Open-Source System for Recognizing Textual Entailment. In Proceedings of the ACL 2012 System Demonstrations, Jeju, Korea, July 2012.

[8] Fellbaum, C. 1998. WordNet: An Electronic Lexical Database. MIT Press, Cambridge, Mass.

[9] Grover, C., Mikheev, A., and Matheson, C. 1999. LT TTT version 1.0: text tokenisation software.

[10] Uchida, H., and Zhu, M. 2001. The Universal Networking Language beyond Machine Translation. "International Symposium on Language in Cyberspace” held at 26 - 27 September 2001, Seoul of Korea, organized by The Korean National Commission for UNESCO and The Cyber Communication Academic Society in Korea, and sponsored by Institute of Information Technology Assessment, Rep. of Korea and UNESCO.