ISSN: 2277-9655
Impact Factor: 4.116

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

EVENT EXTRACTION FROM NATURAL LANGUAGE TEXT

Dr. D. Ramesh , Dr. S. Suresh Kumar *
Department of C.S.E, J.N.T.U.H College of Engg Jagtial,Karimanagar Dist, Telangana, INDIA
* Department of I.T, J.N.T.U.H College of Engg Jagtial,Karimanagar Dist, Telangana, INDIA

DOI: 10.5281/zenodo.58638

ABSTRACT

Extraction and representation of events plays an important role in solving many natural Language processing applications, namely questioning answering system, named entity Recognition, text summarization etc. Events are defined as happening or Situations that occur in the real world. Several methods were defined to annotate the events manually. This paper aim to provide a framework that automatically extract and represent the events that occur in the natural language text. Experiments were conducted on TIME BANK Corpus which consist of nes articles. Most of the events were extracted by our method when compared with other events extraction methods, the results of our method were found to be encouraging.

KEYWORDS: TIMEML, TARSQI, Events, POS Tagging.

INTRODUCTION

Event is a cover term for situations that happen or occur. The role of events are important for applications ranging like time line construction, text summarization, question answering system etc. Events are also defined as predicates that describes the states or circumstances in which something obtains or holds true. Events may be expressed by means of tensed or un tensed verbs nominalizations adjectives, predicative clauses, or prepositional phrases.
Events may be expressed by means of:

1. **tensed verbs** : Ex:- A fresh flow of lava, gas and debris **erupted** there Saturday.
2. **un tensed verbs** : Ex:- Prime Minister Benjamin Netanyahu called the prime minister of the Netherlands to **thank him** for thousands of gas masks his country has already contributed.
3. **nominalizations** : Ex:- Israel will ask the United States to delay a military **strike** against Iraq until the Jewish state is fully prepared for a possible Iraqi **attack**
4. **adjectives** : Ex:- A Philippine volcano, **dorman**t for six centuries, began exploding with searing gases, thick ash and deadly debris.
5. **predicative clauses** : Ex:- " There is no reason why we would not **be prepared** ", Mordechai told the Yediot Ahronot daily.
6. **Prepositional phrases** : Ex:- All 75 people on board the Aeroflot Airbus **died**. All mentioned above events can be used in natural language processing.

TIMEML[1][2] is one of the markup language designed for extraction and representation of event that may exist in a given text. The TimeML project's goal is to create a standard markup language for representation of events in a document. There are four major tags that are specified in TimeML, they are EVENT[3], TIMEX3, SIGNAL and LINK. Event Extraction in TimeML performs two tasks majorly
1) Event Recognition with distinguish classes
2) Analysis of grammatical features such as tense and aspect.
Event identification is performed based on the notations of event as defined in TimeML. Various strategies have been used for recognizing events within categories of verb. Event identification is based on a lexical lookup, accompanied by minimal contextual parsing in order to exclude weak predicates like be or have or should or could. Identifying events expressed by nouns, on the other hand, involves a disambiguation phase in addition to lexical lookup. TimeML

considers events as situations that happen or **occur**. Events can be ***punctual*** or last period of time. In addition, subordinate verbs that express events which are clearly temporally located, but whose complements are generics are not tagged. For example consider the statement "He **said** participants are prohibited from mocking one another". Even though the verb **said** is temporally located, but it is not tagged due to its complement participants are prohibited from mocking one another, is generic. As for event attributes definitions are concerned, TIMEML use seven abstract event classes. Order of those subdivisions of items were presented as follows:

1) Occurrence: die, crash, build.
2) State: on board, kidnapped.
3) Reporting: say, report.
4) I-Action: attempt, try, promise.
5) IState: believe, intend, want.
6) Aspectual: begin, stop, and continue.
7) Perception: see, hear, watch, and feel.

Apart from this, The Backus Naur Forms (BNF) rules required to tag the Event.
Following lines represents BNF rules.
attributes::= eid class
eid::= e<integer>
class:: = 'REPORTING' | 'PERCEPTION' | 'ASPECTUAL' |'I_ACTION' | 'I_STATE' |'STATE' | 'OCCURRENCE'
Stem::= CDATA.

Events are primarily represented with EVENT tag in TimeML. So we have to know about the event tag. Every event tag consist of event id and event class.

Attributes for EVENT:

a) Event ID number (eid): Non-optional attribute. Each event has to be identified by a unique ID number.
b) Class: Non-optional attribute. Each event belongs to one of the following classes.

- **REPORTING:** These are generally verbs such as: say, report, tell, explain, state.
- **PERCEPTION:** Such events are typically expressed by verbs like: see, watch, glimpse, behold, view, hear, listen, overhear.
- **ASPECTUAL:** Initiation: begin, start, commence, set out, set about, lead off, originate, initiate.
- I **ACTION:** For example verbs like attempt, try, and scramble.
- **I_STATE:** For example verbs like believe, think, suspect, imagine, doubt, feel, be conceivable, and be sure.
- **STATE:** States describe circumstances in which something obtains or holds true. However, we will only annotate:
- **OCCURRENCE:**This class includes all the many other kinds of events describing something that happens or occurs in the world. Eg: The Defence Ministry said 16 planes have **landed** so far with protective equipment against biological and chemical warfare.

TARSQI :

It is the acronym of " Temporal Awareness and Reasoning Systems for Question Interpretation" [4][5][6] It tags the events and other temporal information in the given text. It just annotates the given text and gives the output in the form of xml file. The TARSQI system can be used stand-alone or as a means to alleviate the tasks of human annotators. Parts of it have been integrated in Tango, a graphical annotation environment for event ordering the system is set up as a cascade of modules that successively add more and more TimeML annotation to a document. The input is assumed to be part-of-speech tagged and chunke TTK identifies temporal expressions and events in natural language texts, and parses the document to order events and to anchor them to temporal expressions. TTK contains the following components:

- GUTime - extraction of time expressions
- Evita - event extraction
- Slinket - modal parsing
- S2T - temporal repercussions of modal relations
- Blinker - opportunistic pattern-based parsing of temporal relations, based on GutenLink
- Classifier - MaxEnt classifier trained on TimeBank

ISSN: 2277-9655
Impact Factor: 4.116

- Sputlink - constraint propagation (aka temporal closure)
- Link Merger - uses Sputlink to ensure consistency of all relations

The long-term vision with TTK[7][8] is to provide a set of tools that can be adapted to new data and that would let the developer set up a processing chain to fit his or her particular needs. To that end, the toolkit will include tools to inspect and edit dictionaries and pattern sets as well as tools to train machine learning components on new data.

CALLISTO :

Callisto[9][10] is another tool used to add events, timexes and signals to the given text. This tools gives the graphical output about the events in given text. The project focuses on automatic extraction of events from given natural language text. All the present tools are based on manual annotation of the events in the text. The automatic extraction lists out all the events in the given text. The extracted events can be further used for various purposes mainly the reasoning of temporal context.

The remaining part of this paper is organized as follows: Section 2 introduces a framework that can identify the events from natural language text. Experiments and results are described in section 3. Conclusion and future directions are drawn at last.

ARCHITECTURE

This paper proposes a method that can extract the events from any given document. This framework consist of three sub tasks namely Preprocessor,POS Tagger and Event Extraction Modules.

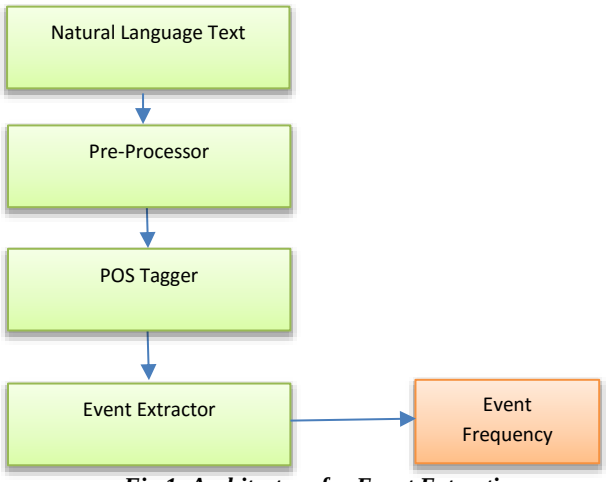

Fig 1: Architecture for Event Extraction

In the architecture a paragraph of text containing Natural languge information is accepted as input. Pre-processing steps are performed on the input in order to eliminate the stop words from the given text and remove the punctutaions and generate tokens at last. For each token POS tags are assigned by using the POS Tagger module. Each and every token in the token that are tagged by the pos tagger are compared with the rules described in the event extractor. If the token is in verb form or action noun it is considered as events and it separated from the other tokens. The procedure is repeated with all the other tokens to identify all the events present in the given text. As a next step the event frequency module identifies the the number of times a particular event has occurred in the given natural language text.

For example consider the following text

"Finally today, we learned that the space agency has finally taken a giant leap forward. Air Force Lieutenant Colonel Eileen Collins will be named commander of the Space Shuttle Columbia for a mission in December. Colonel Collins has been the co-pilot before, but this time she ' s the boss. Here ' s ABC ' s Ned Potter. Even two hundred miles up in space, there has been a glass ceiling. It wasn't until twenty years after the first astronauts were chosen that NASA finally included six women, and they were all scientists, not pilots. No woman has actually been in charge of a

mission until now. Just the fact that we ' re doing the job that we ' re doing makes us role models. That was Eileen Collins, after she flew as the first ever co-pilot. Being commander is different. It means supervising the rest of the crew in training and leading them in flight. It is, in short, the kind of management job many American women say they ' ve had to fight for. In space, some say female pilots were held up until now by the lack of piloting opportunities for them in the military. Once Colonel Collins was picked as a NASA astronaut, she followed a normal progression within NASA. Nobody hurried her up. No one held her back. Many NASA watchers say female astronauts have become part of the agency ' s routine. But they still have catching up to do two hundred and thirty four Americans have flown in space, only twenty six of them women. Ned Potter, ABC News."

When the above paragraph of text is given as input, it is processed through several steps like POS tagger, and Event extractor finally a class of terms which belongs to action verbs are extracted as events without any duplications. The list of events obtained after the execution is depicted in the figure below

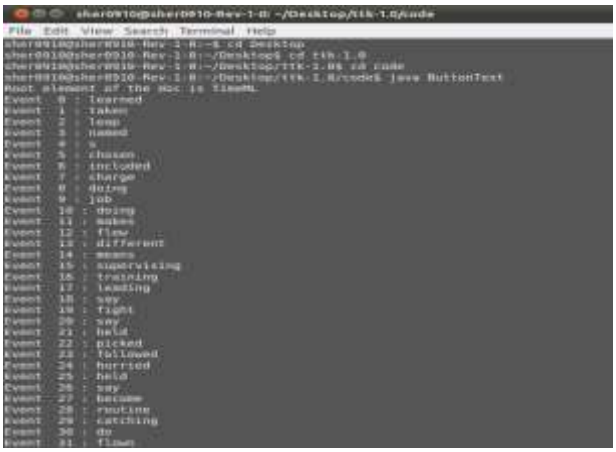

Fig 2: Output screen depicting the list of Events Extracted

Experiments were performed on the data set from Time Bank corpus which is a collection of news articles. The proposed method found to perform better with a simplified approach when compared with other event extraction methods. The proposed architecture found to obtain encouraging performance.

CONCLUSION

Event extraction aims to retrieve events from natural language text and represent them in a structured knowledge form. Automatic Extraction of events removes all the difficulties of manual annotation of events like going through the whole file to get the events. It is found to be significant work in computational linguistics and artificial intelligence.This paper projects a framework that make use of event extractor tool to identify the events in the given natural language text. Experiments were conducted on the TIME Bank corpus consisting of news articles. When the results are compared with other methods it is observed that the results of the proposed method were encouraging.

REFERENCES

[1] Ingria, Robert and James Pustejovsky," TimeML Specification 1.0", 2002, available at http://time2002.org.
[2] Pustejovsky, James and Robert Gaizauskas, "Time and Event Recognition in Natural Language", CSLI Publications, Stanford, 2003.
[3] J. Pustejovsky, et al., "The TIMEBANK corpus," 2003.
[4] R. Sauri et al., "Evita : a robust event recognizer for QA systems", Human Language Technology and Empirical Methods in Natural Language Processing, 2005.
[5] S. Bethard and J. H. Martin, " Identification of event mentions and their semantic class", Empirical Methods in Natural Language Processing(EMNLP), 2006.

[6] Magnini, Bernardo, Borovetz, Bulgaria, "Open Domain Question Answering : Techniques, Systems and Evaluation", Conference on Recent Advances in Natural Language Processing (RANLP), 2005.
[7] I. Androutsopoulos, "Exploring Time, Tense and Aspect in Natural Language Database Interfaces", John Benjamins, Amsterdam and Philadelphia, 2002.
[8] Z. Huang, K. F. Wong, W. Li, D. Song and P. Bruza, "Back to the future: a logical frame work for temporal information representation and inferencing from financial news", in Proc. of 2003 International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE03), Beijing, 2003.
[9] J. R. Hobbs and F. Pan, "An ontology of time for the semantic web", ACM Trans. Asian Lang. Info. Process. 3(1), 2004.
[10] D. Ahn, et al., "Extracting Temporal Information from Open Domain Text: A Comparative Exploration", Digital Information Management, 2005.