Temporal and Spatial Feature Extraction of Events Based on LTP

Jiang Gaoyu
School of Mechanical, Electrical &Information Engineering
Shandong University
Weihai, China
jgyxyyxy@163.com

Pan Jingchang
School of Mechanical, Electrical &Information Engineering
Shandong University
Weihai, China
jingchangpan@163.com

***Abstract*—Natural language in the form of text is an important achievement of human wisdom. Event extraction belongs to the level of information extraction in natural language processing, which aims to present an event in natural language with structured results. With the development of Internet technology, the richness of the high inflation of the network information, the Internet has become a large document library and the potential for big data sets. In order to extract events and their temporal and spatial features from such massive information, this paper proposes a LTP-based preparative filling mechanism, which can effectively use the case of missing event features.**

Keywords—NLPt, event extraction, style, temporal and spatial features, LTP

I. Introduction

The meaning of event extraction is to present an event as a structured result, such as who, when, where and what has been done. Event extraction technology usually first determines the trigger words of events, and then determines the arguments that need to be extracted according to the classification results of event trigger words. At present, there are two main directions for event extraction, one is based on pattern matching[1,2,3], the other is based on machine learning[4,5]. The method based on pattern matching or knowledge relies on expert analysis to obtain patterns, which is more suitable for specific fields. The method based on machine learning uses classifiers such as SVM and maximum entropy to make event extraction as a classification task, so the domain correlation of this kind of method is lower.

There are also many advances in event extraction in Chinese. In this paper, the original text information of multi-domain entities will be extracted to events. The problem of event classification is of low concern, mainly focusing on the temporal and spatial attributes of the core events of entities. Here, LTP language technology platform is used to assist in extracting event features.

There are many definitions of events. Some focus on the definition of events in the field of language processing[6,7], others on information retrieval[8]. Although there are some differences at the point of departure, there are still some similarities in the definition of events.

- Events can be regarded as a thing occurring at a certain time and place, which is a kind of knowledge composition
- Events are expressive units with certain structures, which are almost structured and have some attributes, most of which are around predicate verbs.

Based on this, the event object in this paper is defined as follows. Suppose that event E must have a set of attributes p_1, p_2, p_3... p_i...p_n satisfies propositions Q_1, Q_2, Q_3...Q_i...Q_n respectively, there are rules.

$$E \rightarrow Q_1 \wedge Q_2 \wedge Q_3 \wedge \cdots \wedge Q_i \wedge \cdots \wedge Q_n \qquad (1)$$

And

$$Q_1 \wedge Q_2 \wedge Q_3 \wedge \cdots \wedge Q_i \wedge \cdots \wedge Q_n \rightarrow E \qquad (2)$$

Formula 1 shows that the event itself can get a group of attributes satisfying their propositions. Formula 2 shows that the event can be uniquely determined by this group of attributes.

II. Ltp Related Modules

LTP is a natural language processing system for Chinese developed by Harbin University of Technology. It covers rich, efficient and accurate natural language processing modules including Chinese word segmentation, part-of-speech tagging, named entity recognition, dependency parsing, semantic role tagging, etc. It also provides program interfaces, visualization tools and language network services.

This section describes the LTP-related modules used in this paper and the annotation meaning of its processing results.

A. Part-of-speech Tagging

This module judges the part-of-speech categories of each word in a sentence, and then marks the noun, verb, adjective and other part-of-speech categories on the result of participle. It plays an important role in the parsing module and is also an indispensable part of the event feature extraction algorithm. In this paper, for the result of part-of-speech tagging of sentences, the parts related to location, time information and entity names are mainly focused, as shown in Table I.

TABLE I. LIST OF RELEVANT PART-OF-SPEECH

Tag	Description	Sample
nh	Person name	Du Fu, Tom
ni	Organization	Congress
nl	Location noun	Suburb
ns	Geographical name	New York
nt	Temporal noun	Modern times

B. Dependency Parsing

The main work of this module is to analyze the syntactic dependencies contained in sentences and find out the grammatical structures in sentences. In this paper, we mainly focus on the core relationship, the juxtaposition relationship and so on. The specific meaning of the annotated relationship is shown in Table II.

TABLE II. LIST OF ANNOTATION RELATIONS FOR DEPENDENT PARSING

Relation type	Tag	Sample
Subject-predicate relation	SBV	He gives me a car (give → him)
Verb-object relation	VOB	He gives me a car (give → car)
Parallel relation	COO	Apples and pears (apple → pear)
Core relation	HED	Core Relation in Sentences

C. Semantic Role Labeling

This module is responsible for discovering the arguments of a given predicate, i.e. the semantic roles, such as the agent of the action, acceptance, time feature and location feature of the predicate. Semantic role annotation plays an important role in event feature extraction algorithm. In this paper, we focus on two core semantic roles and some additional semantic roles, as shown in Table III.

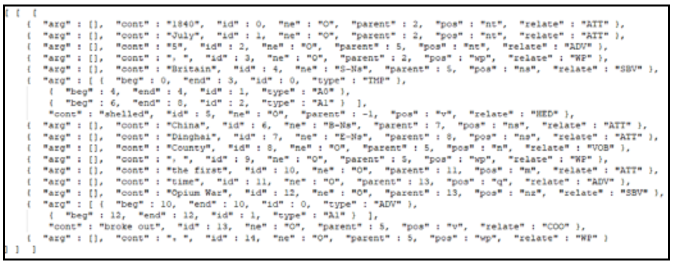
TABLE III. LIST OF RELEVANT SEMANTIC ROLE

Tag	Explanation
A0	Actor of action
A1	Effect of action
LOC	locative
TMP	temporal

III. EVENT FEATURE EXTRACTION

In this section, an integration algorithm is proposed to obtain specific attribute information of events from the return result string of LTP-Server, and then to obtain a structured event or event list.

After LTP-Server analysis, the results of word segmentation, named entity tagging, dependency parsing and semantic role tagging are returned. For the sentence "On July 5, 1840, Britain shelled Dinghai County, China. The first Opium War broke out." in Chinese, the analysis results obtained after submitting LTP Server are shown in Fig. 1. For the convenience of reading, some translations have been made in the Chinese part of the processing results.



```
[ [  [
    { "arg" : [], "cont" : "1840", "id" : 0, "ne" : "O", "parent" : 2, "pos" : "nt", "relate" : "ATT" },
    { "arg" : [], "cont" : "July", "id" : 1, "ne" : "O", "parent" : 2, "pos" : "nt", "relate" : "ATT" },
    { "arg" : [], "cont" : "5", "id" : 2, "ne" : "O", "parent" : 5, "pos" : "nt", "relate" : "ADV" },
    { "arg" : [], "cont" : "，", "id" : 3, "ne" : "O", "parent" : 2, "pos" : "wp", "relate" : "WP" },
    { "arg" : [], "cont" : "Britain", "id" : 4, "ne" : "S-Ns", "parent" : 5, "pos" : "ns", "relate" : "SBV" },
    { "arg" : [ { "beg" : 0, "end" : 3, "id" : 0, "type" : "TMP" },
      { "beg" : 4, "end" : 4, "id" : 1, "type" : "A0" },
      { "beg" : 6, "end" : 8, "id" : 2, "type" : "A1" } ],
      "cont" : "shelled", "id" : 5, "ne" : "O", "parent" : -1, "pos" : "v", "relate" : "HED" },
    { "arg" : [], "cont" : "China", "id" : 6, "ne" : "B-Ns", "parent" : 7, "pos" : "ns", "relate" : "ATT" },
    { "arg" : [], "cont" : "Dinghai", "id" : 7, "ne" : "E-Ns", "parent" : 8, "pos" : "ns", "relate" : "ATT" },
    { "arg" : [], "cont" : "County", "id" : 8, "ne" : "O", "parent" : 5, "pos" : "n", "relate" : "VOB" },
    { "arg" : [], "cont" : "，", "id" : 9, "ne" : "O", "parent" : 5, "pos" : "wp", "relate" : "WP" },
    { "arg" : [], "cont" : "the first", "id" : 10, "ne" : "O", "parent" : 11, "pos" : "m", "relate" : "ATT" },
    { "arg" : [], "cont" : "time", "id" : 11, "ne" : "O", "parent" : 13, "pos" : "q", "relate" : "ADV" },
    { "arg" : [], "cont" : "Opium War", "id" : 12, "ne" : "O", "parent" : 13, "pos" : "nz", "relate" : "SBV" },
    { "arg" : [ { "beg" : 10, "end" : 10, "id" : 0, "type" : "ADV" },
      { "beg" : 12, "end" : 12, "id" : 1, "type" : "A1" } ],
      "cont" : "broke out", "id" : 13, "ne" : "O", "parent" : 5, "pos" : "v", "relate" : "COO" },
    { "arg" : [], "cont" : "，", "id" : 14, "ne" : "O", "parent" : 5, "pos" : "wp", "relate" : "WP" }
] ]  ]
```

Fig. 1. Example of LTP Return

The first step of the algorithm is to find the trigger word of the event. Here, the words whose semantic role is not empty are regarded as the candidate set of trigger words of the event. When the syntactic analysis result of a trigger word is "HED", we think that the trigger word is the core predicate trigger of the sentence, such as the trigger word "shelled" in the example, the event marked by which is the core event of this sentence. When the relate attribute of a candidate trigger word is "COO" and its parent attribute points to the core predicate trigger word, this word is identified as a sub-trigger word, such as the trigger word "broke out" in the sample. Events marked by sub-triggers are sub-events of the sentence.

After determining the trigger words of core-events and sub-events, the subject of events and the characteristics of time and space need to be extracted. According to the TMP tag of trigger words, the temporal feature is obtained. The spatial feature is obtained from the LOC tag, and the agent of the event is obtained from the A0 tag. For example, the time characteristic of the trigger word "shelled" is "July 5, 1840", the agent feature is "Britain", and there is no semantic label about the location feature. The trigger word "broke up" has no semantic label about the time and location feature.

In the result of SRL returned by LTP-Server, the loss of features such as time, location and so on often occurs for some trigger words. There are two reasons to cause this situation. One is that there is no relevant information in the original text, which is due to the poor quality of information in the original text and needs to rely on obtaining more and better original text. The other is that this information is omitted in the analysis process of natural language processing tools. For example, according to the semantic understanding of the text, the time characteristic of trigger word "broke up" is "July 5, 1840". However, this feature has not been found by natural language processing tools. Therefore, in order to prevent the occurrence of the second situation as much as possible and ensure the validity of the event extraction results, two strategies are added to the feature extraction algorithm according to the characteristics of Chinese grammar.

- When the feature of sub-event is lost and the feature of core-event is not missing, then the lost is replaced by its value of feature of core event.
- When the feature of the core-event is lost, the preparative word calculated from the result of part-of-speech tagging is used as the feature value.

Preparative words are feature candidates based on part-of-speech tagging results when features are lost. The result of the final execution of the algorithm hopes to provide users with as much feature information as possible, even the most possible value of some features, so the preparative words are particularly important in this demand scenario. Taking the preparative geographical words as an example, the selection rules are as follows.

- In the result of part-of-speech tagging, if the POS attribute of at least one word nodes are "ns", there is preparative geographical word in the sentence. If there are id-connected words in these nodes, these nodes are connected in the order of ID to form the preparative geographical word together.
- When there are multiple preparative words in a feature, the one that is the most advanced in the original text sentence will be chosen.
- When there are multiple preparative words for a feature, the one whose value is different from that of other features of each trigger word will be chosen.

Algorithms I illustrates the preparative word acquisition algorithm with an example of prepositional geographical word.

The input of the algorithm is the result of part-of-speech tagging of the sentence and the initial other feature set of event obtained from SRL. According to the results of part-of-speech tagging, the set of words that may constitute the preparative geographical words is obtained in 1-4 steps of the algorithm; and in 5-14 steps, candidate preparative word nodes are assembled in the order of their ID from front to back in the sentence to determine whether they are in the set of other features of each event, if not, the preparative word is obtained, and if so, the next candidate preparative word is continued to be searched. 15-18 steps are used to return the results. The possible results of the preparative word acquisition algorithm are as follows:

- Find the corresponding preparative word of the missing feature, satisfy the requirement that it is not in other feature sets, and then the preparative value is returned.
- No qualified preparative word is found, but the list of candidate preparative words is not empty. Under this kind of circumstances, the first phrase result which is added into the list of candidates is returned.
- No qualified preparative word is found, and the list of candidate prepositions is empty. The NULL value will be returned and filled in automatically when edited manually.

ALGORITHMS I PREPARATIVE GEOGRAPHICAL WORD ACQUISITION

```
INPUT:   List of result words in Part of Speech Tagging W,
         the list size is n,
         Temporal feature set obtained from SRL T,
         Agent feature set O
OUTPUT:  Prepositional geographical word preLoc

00 init list ns = {}, B={} // ns to store the word ID whose pos attribute is ns, and B to store the candidate
01 for i = 0…n:
02   if W[i].pos = "ns":
03     ns.add(i) // add the word ID to the list
04 m = ns.size // the size of list is m
05 while(i<=m)
06   preLoc = W[i]
07   while (getNeighbor(i)) // until the connected word is not in the list
08     preLoc = preLoc +W[i+1] // assembling preparative word
09     i = i + 1
10   B.add(preLoc) // add to candidate set
11   if preLoc ∉ T ∪ O: // whether the assembled word are in another feature set
12     break // get selected preparative word
13   else:
14     i =i +1 // continue searching for possible preparative word
15 if preLoc ∉ T ∪ O:
16   return preLoc
17 else:
18   return B[1] // When all preparative words are repeated with other features, return the first
```

Similarly, according to whether there are "nl", "nt" and "nh" in POS attributes of word nodes, we can get preparative location word, preparative time word and preparative entity word. When there are preparative location words and preparative geographical words at the same time, preparatory geographical words are selected as the spatial characteristics of events. In the example, the preparative temporal feature is "July 5, 1840". The preparative spatial feature has the choice of "Britain" and "China Dinghai", but "Britain" is agent feature of the core-event, so the preparative spatial feature here is "China Dinghai".

The integration algorithm can finally get the event set with their features of agent, time and location. Finally, the time feature is regularized and the event list extracted from LTP-aided analysis can be obtained by eliminating the events with unclear time information. For the example in Figure 1, two events can be separated: the core- event (time: July 5, 1840, location: Dinghai, China, agent: Britain, details: shelling Dinghai County Town of China) and sub-event (time: July 5, 1840, location: Dinghai, agent: Britain, details: the outbreak of the Opium War). Overall flow chart of the algorithm is shown in Fig. 2.

In the case of building LTP-Server cluster, the server can withstand the analysis process of a large number of data. The original documents uploaded by all clients can be semantically analyzed, and the analysis results are returned to the background of the system for parsing. Similarly, a large number of analysis requests submitted by a single client can also be processed efficiently, therefore, after natural language text is preprocessed by clauses, the algorithm can be used not only for information extraction from single sentence corpus, but also for paragraphs and articles.

In the analysis of a sentence, it is possible to get event objects of multiple entities about multiple actions, or event objects of multiple entities about the same action. Event lists can be obtained during paragraph and text analysis.

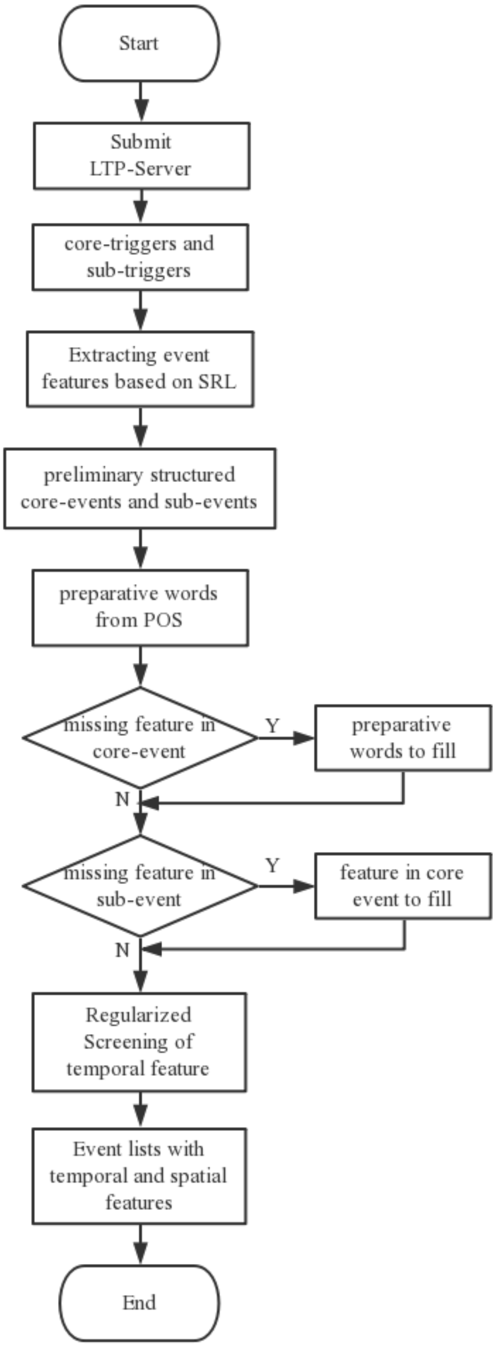

Start
Submit LTP-Server
core-triggers and sub-triggers
Extracting event features based on SRL
preliminary structured core-events and sub-events
preparative words from POS
missing feature in core-event
Y
preparative words to fill
N
missing feature in sub-event
Y
feature in core event to fill
N
Regularized Screening of temporal feature
Event lists with temporal and spatial features
End

Fig. 2. Flow Chart of the Algorithm

IV. CONCLUSION

In this paper, an assistant macro-event extraction method based on LTP cloud platform is proposed. According to Chinese grammatical features, preparative words of event features are extracted to fill in the missing temporal and spatial features of events. The core part of the event and its feature extraction algorithm is to determine some event features according to the semantic role annotation of the text. Then, according to the characteristics of Chinese grammar, preparative words are obtained from the results of part-of-speech annotation of the text. When the natural language processing tool does not obtain the corresponding features, they will be filled with corresponding preparative values, and events are filled in time and space domain as much as possible.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation (U1931209) and Key R & D Project of Shandong Province (2018GGX101023)

REFERENCES

[1] Huang R, Riloff E. Peeling Back the Layers: Detecting Event Role Fillers in Secondary Contexts.[C]// Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011.

[2] Liao S, Grishman R. Can document selection help semi-supervised learning?: a case study on event extraction[C]// Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers. 2011.

[3] Grishman R. Filtered Ranking for Bootstrapping in Event Extraction[C]// The. 2010.

[4] Yu H, Zhang J, Ma B, et al. Using cross-entity inference to improve event extraction[C]// Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011.

[5] Mcclosky D, Surdeanu M, Manning C D. Event extraction as dependency parsing for BioNLP 2011[C]// Bionlp Shared Task Workshop. 2011.

[6] Fellbaum C. WordNet[J]. Theory & Applications of Ontology Computer Applications, 2010:231-243.

[7] Grohmann K K . Events as Grammatical Objects: The Converging Perspectives of Lexical Semantics and Syntax (review)[J]. Language, 2003, 79(3):671-672.

[8] Smucker M D, Allan J, Carterette B. A comparison of statistical significance tests for information retrieval evaluation[J]. 2007.