Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

BUEES: a Bottom-Up Event Extraction System

Xiao Ding[†‡1], Bing Qin[†2], Ting Liu[†3]

[1]Research Center for Social Computing and Information Retrieval, , Harbin Institute of Technology, Harbin 150001, China
[2]Research Center for Social Computing and Information Retrieval, , Harbin Institute of Technology, Harbin 150001, China
[3]Research Center for Social Computing and Information Retrieval, , Harbin Institute of Technology, Harbin 150001, China
[†]E-mail: xding@ir.hit.edu.cn; bqin@ir.hit.edu.cn; tliu@ir.hit.edu.cn

Abstract: Traditional event extraction systems mainly focus on event type identification and event participants extraction based on pre-specified event type paradigms and manually annotated corpora. However, different domains have different event type paradigm. When transferring to a new domain, we have to build a new event type paradigm and annotate new corpus from scratch. This kind of up-bottom event extraction system requires massive human effort, and hence prevents event extraction from widely applicable. In this paper, we present BUEES - a Bottom to Up Event Extraction System, which extracts events from the Web in a completely unsupervised way. The system automatically build event type paradigm in the input corpus, and then proceeds to extract a large number of instance patterns of these events. Subsequently, the system extracts event arguments according to these patterns. In a series of experiments, we demonstrate that the successful performance of BUEES and compare it to a state-of-the-art Chinese event extraction system - a supervised event extraction system. The experimental results show that BUEES performs comparably to it (5% higher F-Measure in event type identification and 3% higher F-Measure in event argument extraction), but without any human efforts.

Key words: Event Extraction, Bootstrapping, Bottom-Up, Unsupervised
doi: **Document code:** A **CLC number:** TP391.4

1 Introduction

Information Extraction (IE) is the task of identifying factual description (entities, relations and events) from unstructured natural language text and extracting information related to those descriptions Grishman (1997). Event extraction remains the most challenging task, because a larger field of view is often needed to understand how facts tie together, and it is situated at the end of an IE pipeline thus suffers from propagation of errors from word segmentation, named entity recognition, coreference resolution, etc. Ahn (2006). Although event extraction is a challenging problem, it has been widely used in several different specific domains, such as musical reports Xiao *et al.* (2011), financial analysis Lee *et al.* (2003), biomedical investigation Pham *et al.* (2013) and legal documents Schilder (2007).

The main approaches used by most event extraction systems are based on knowledge engineering technology or machine learning technology. The knowledge engineering based event extraction systems use extraction patterns or rules to identify and extract the relevant information Riloff (1996); Soderland (1999); Yangarber *et al.* (2000). Most of these systems use annotated training data to learn pattern matching rules, based on lexical, syntactic, or semantic information. These systems traditionally were the top performers in most event extraction benchmarks, such as MUC Chinchor *et al.* (1993) and ACE Yeh *et al.* (2002). In the machine learning approach, domain experts label instances of the target concepts in a set of documents Miwa *et al.* (2010); Hong *et al.* (2011); Ritter *et al.* (2012). The system then learns a model of the extraction task, which can be applied to new documents automatically.

Both of these approaches require substantial human effort, and hence prevent event extraction system become domain adaptation and more widely applicable. Recently, there appear many semi-supervised event extraction systems that aim to reduce the annotated data required, ideally to a set of seed instances of the target events. One such system is Liao and Grishman Liao and Grishman (2010). They use two state-of-the-art bootstrapping-based event extraction systems, then rank their candidate patterns and accept the top-ranked patterns in each iteration.

However, for all such approaches it is still necessary to specify the target events in advance. In this paper, we explore the possibility to construct a completely unsupervised and bottom-up event extraction system, which does not need to pre-specify the target event types.

Although the task is important and emergent, the challenge of it at least lies into two aspects.

- How to automatically build event type paradigm. As pre-specifying interested event types in a domain needs rich background knowledge, event type paradigm is traditionally built by domain experts. It is a costly work.
- How to extract event arguments in a totally unsupervised way is a challenging problem. All of event extraction systems reported in the literature more or less need manual efforts.

To address the above challenges, in this paper, we design and develop a bootstrapping-based **BUEES** (a **B**ottom-**U**p **E**vent **E**xtraction **S**ystem). The system automatically builds event type paradigm from scratch, based on the definition of event trigger: *the words that most clearly expresses an event's occurrence*, and our key observations: *triggers are the most important lexical units to represent events. A set of triggers with similar meaning or usage represents the same event type.* Event types can be discovered based on trigger clustering.

When the target event types are available, the next step is to learn a set of patterns to extract event arguments from the web documents. The system takes as input a small set of event seeds. It then uses these seeds to search the web to get more documents that contain the event seeds. The extraction patterns will be learned from these documents. Finally, we can extract event arguments by using these useful patterns.

Event: Mao Zedong was born in Xiangtan, Hunan Province in 1893.
Event Type: Life
Event Subtype: Be-Born
Trigger: born
Arguments:
- **Person:** Mao Zedong
- **Time:** 1893
- **Place:** Xiangtan, Hunan Province

Figure 1 Event extraction example

The major contributions of the work presented in this paper are as follows.

- To the best of our knowledge, our work is the first to propose the Bottom-Up Event Extraction System. We automatically build event type paradigm. Based on the paradigm, we proceed to implement traditional event extraction tasks.
- Our system is completely unsupervised. To our knowledge, all of the bootstrapping-based semi-supervised event extraction systems need the manually constructed seeds in advance. However, our system can generate and select event seeds automatically.

The rest of this paper is organized as follows. Section 2 describes the task. Section 3 introduces the architecture of the system and then describes each component in detail. Section 4 evaluates the proposed method. We review related work of event extraction in Section 5, and finally conclude this paper in Section 6.

2 Task Description

2.1 ACE Event Extraction Task

The event extraction task we addressing is that of the Automatic Content Extraction (ACE) evaluations LDC (2005), where an event is defined as a specific occurrence involving participants. Event extraction task requires that certain specified types of events should be detected. We first introduce some ACE terminology to understand this task more easily:

- **Entity:** an object or a set of objects in one of the semantic categories of interest
- **Entity mention:** a reference to an entity (typically, a noun phrase)

- **Event trigger:** the main word which most clearly expresses an event occurrence
- **Event arguments:** the entity mentions that are involved in an event
- **Event mention:** a phrase or sentence within which an event is described, including trigger and arguments
- **Event type:** a particular event category, such as "Conflict/Attack", "Life/Die", etc.

The ACE 2005 evaluation has 8 types of events, with 33 subtypes; for the purpose of this paper, we will treat these simply as 33 separate event types and do not consider the hierarchical structure among them. Besides, the ACE evaluation plan defines the following standards to determine the correctness of an event extraction:

- *A trigger is correctly labeled* if its event type and offset (viz., the position of the trigger word in text) match a reference trigger.
- *An argument is correctly identified* if its event type and offsets match any of the reference argument mentions, in other word, correctly recognizing participants in an event.
- *An argument is correctly classified* if its role matches any of the reference argument mentions.

Figure 1 shows an example of an ACE event, where "born" is the trigger word. Its event type is "Life" and the subtype is "Be-Born". This event consists of three arguments, namely, "Mao Zedong", "1893", "Xiangtan, Hunan Province", which corresponds to three role labels in the Life/Be-Born event template of "Person", "Time-Within" and "Place", respectively.

2.2 New Task for Event Extraction

In addition to traditional event extraction tasks introduced above, we propose a new task of building event type paradigm. ACE manually annotates 8 types and 33 subtypes of events and construct the event type paradigm that are shown in Table 1. However, building an event type paradigm in this way not only requires massive human effort but also tends to be very data dependent. As a result, it may prevent

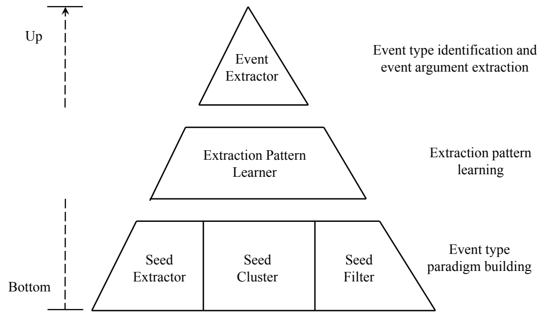

Figure 2 The architecture of BUEES

Table 1 ACE event type paradigm

Types	Subtypes
Life	Be-Born, Marry, Divorce, Injure, Die
Movement	Transport
Transaction	Transfer-Ownership, Transfer-Money
Business	Start-Org, Merge-Org, Declare-Bankruptcy, End-Org
Conflict	Attack, Demonstrate
Contact	Meet, Phone-Write
Personnel	Start-Position, End-Position, Nominate, Elect
Justice	Arrest-Jail, Release-Parole, Trial-Hearing, Charge-Indict, Sue, Convict, Sentence, Fine, Execute, Extradite, Acquit, Appeal, Pardon

the event extraction from being widely applicable. Since event types among domains are different, the event type paradigm of ACE, which does not define music related events, is useless for the music domain event extraction. So we have to build a totally different event type paradigm for the music domain from scratch.

3 Description of BUEES

The goal of BUEES is to extract instances of events without any human supervision. The system is built based on the framework of bootstrapping and its architecture is shown in Figure 2. Traditional bootstrapping-based semi-supervised event extraction systems use manual construction of seed examples to learn extraction patterns, and then identify event types and recognize event arguments. Since the number of seeds is limited, the quality and coverage of seeds highly affect the performance of extraction patterns. However, in this paper, we propose to automatically construct the set of seeds (Section 3.1 - 3.3) and explore a novel extraction pattern learning algorithm (Section 3.4 - 3.5).

The system works in three stages. During the first stage, the system builds the event type paradigm and prepares seed instances. During the second stage, the system learns extraction patterns based on the seed set. During the third stage, the system identifies event types and extracts event arguments based on the learned patterns. In the following sections, we introduce each component of BUEES in detail.

3.1 Event Type Building and Seed Extractor

Since event trigger is the word that most clearly expresses an event's occurrence, the key idea of this paper is to automatically construct an event type paradigm by clustering event triggers. For example, in the ACE corpus, a set of event triggers {倒闭, 闭门, 关闭, 停业, 解散} ({bankrupt, shut down, close, close down, dismiss}) represents the sense of the event type "Business/End-Org". In addition to extract the event trigger in the sentence, we also extract its subject and object as the seed instance (*subject*, *trigger*, *object*). The event seeds are used in two ways. First, event type paradigm is built by clustering seeds. Second, the seeds are used for learning extraction patterns (refers to Section 3.4)

Sudo *et al.* (2003) summarized three classical models for representing events. All of these three models rely on the syntactic tree structure and the trigger is specified as a predicate in this structure. In order to accurately extract event seeds, we employ the predicate-argument model Yangarber *et al.* (2000); Ding *et al.* (2013) which is based on a direct syntactic relation between a predicate and its arguments. We extract the syntactic relation for predicate-argument model by means of the HIT (Harbin Institute of Technology) Dependency Parser Che *et al.* (2009). Based on the predicate-argument model, we propose a seed extraction algorithm (**SE**). The details are shown in Algorithm 1.

Take the following sentence as an example:

$$\frac{\text{毛泽东}}{1}\ \frac{\text{1893年}}{2}\ \frac{\text{出生}}{3}\ \frac{\text{于}}{4}\ \frac{\text{湖南湘潭}}{5} \rightarrow \frac{MaoZedong}{1}\ \frac{was}{2}\ \frac{born}{3}\ \frac{in}{4}\ \frac{Xiangtan\ Hunan}{5}\ \frac{in}{6}\ \frac{1893}{7}$$

The HIT Chinese Dependency Parser dependencies are:

SBV (出生-3, 毛泽东-1)
→ (born-3, Mao Zedong-1)
VOB (出生-3, 湖南湘潭-5)
→ (born-3, Xiangtan, Hunan Province-5)
ADV (出生-3, 1893年-2)
→ (born-3, 1893-7)
POB (湖南湘潭-5, 于-4)
→ (Hunan Province-5, in-4)

where each atomic formula represents a binary dependence from the governor (the first token) to the dependent (the second token). The SBV relation, which stands for the subject-predicate structure, means that the head is a predicate verb and the dependent is a subject of the predicate verb; the VOB dependency relation, which stands for the verb-object structure, means that the head is a verb and the dependent is an object of the verb; the ADV relation, which stands for the adverbial structure, means that the head is a verb and the dependent is an adverb of the verb; the POB relation, which stands for the prep-object structure, means that the head is an object and the dependent is a preposition of the object.

Since $V_{SBV} = V_{VOB} = V_t =$ 出生(born) in this case, based on the predicate-argument model, the word "出生(born)" should be extracted as a candidate event trigger and (Mao Zedong, born, Xiangtan Hunan) should be extracted as a candidate event seed instance.

Algorithm 1 SE algorithm

Input: Raw corpus D
Output: Candidate seeds
for document d in raw corpus D **do**
  $d \leftarrow$ Paragraph Splitting
  $d \leftarrow$ Sentence Splitting
  for sentence s in document d **do**
    $s \leftarrow$ Word Segmentation
    $s \leftarrow$ Chinese Dependency Parsing
    $s \leftarrow$ Identify subject-predicate relation (SBV) pair (V_{SBV}, Sub) and verb-object relation (VOB) pair (V_{VOB}, Obj)
    if $V_{SBV} = V_{VOB} = V_t$ **then**
      Extract V_t as candidate trigger
      Extract (sub, V_t, obj) as candidate seed
    end if
  end for
end for

3.2 Seed Cluster

As we discuss above, a set of triggers with the same meaning and usage represents the same event type. We propose to cluster event seeds (i.e., event trigger and its corresponding subject and object) based on their semantic distances, and each of these clusters represents a type of event. Details are shown in Algorithm 2.

For every two seeds p_i and p_j in Algorithm 2, the similarity function $Sim(p_i, p_j)$ is calculated using semantic information provided by HowNet Dong and Dong (2006) as.

$$Sim(p_i, p_j) = \frac{2Sum_s}{Sum_i + Sum_j} \tag{1}$$

$$Sum_s = N_s + S_s + O_s,\ Sum_i = N_i + S_i + O_i,$$
$$Sum_j = N_j + S_j + O_j \tag{2}$$

where S_s and O_s denotes the number of identical sememes in the DEF_s (the concept definition in HowNet) of Sub_i and Sub_j, Obj_i and Obj_j; Si and Sj denote the number of sememes in the DEF_s of Sub_i and Sub_j, respectively; O_i and O_j denote the number of sememes in the DEF_s of Obj_i and Obj_j, respectively. HowNet uses sememes to interpret concepts. Sememes are regarded as the basic unit of the meaning. For example, "paper" can be viewed as a concept, and its sememes are "white", "thin", "soft", "flammable", etc.

A group of event seeds are aggregated to a seed cluster according to their semantic distance, and we view each seed cluster as one kind of event type. Then all these event types are finally employed to construct an event type paradigm.

Algorithm 2 Seed cluster algorithm (**SC**)

```
Input:  Candidate seeds (sub, V_t, obj)
        the Threshold θ
Output: Event clusters EC
EC ← [ ]
for seed p in the set of seeds P do
    Compute the similarity (Sim) between p and the
    rest of other seeds, using function 1 and 2
    if Sim ≥ θ then
        add V_t to the related event type ET_re ∪ {V_t}
    else if Sim < θ then
        set up a new event type ET_new
        EC ← ET_new
    end if
end for
```

3.3 Seed Filter

Although we obtain some useful candidate seeds, certain meaningless candidate seeds come along in the results of the seed extractor as well. Therefore, we introduce a seed filter which uses heuristic rule and ranking algorithm to filter out these less informative antecedent candidates.

Since event trigger words are extracted based on the predicate-argument model, most of these candidate trigger words are verb terms. However, not all of verb terms can be used as trigger words. For example, the copular verb (e.g. "is") rarely acts as the event trigger. To investigate which categories of verbs can serve as event triggers, we classify Chinese verbs into eight subclasses listed in Table 2. Such classification makes each subclass function as one grammatical role. For example, a modal verb will never be the predicate of a sentence and a nominal verb will always function as a noun.

We perform the verb sub-classification model based on the work by Liu *et al.* (2007)[1]. Statistically, about 94% of ACE Chinese event triggers are general verbs or nominal verbs and other types of verbs are rarely as trigger words. In order to en-

[1] The tool is provided by Research Center for Social Computing and Information Retrieval in Harbin Institute of Technology, China

Table 2 The scheme of verb subclass

Verb	Description	Examples
vx	copular verb	他**是** 对的 (He is right)
vz	modal verb	你**应该** 努力工作 (You should work hard)
vf	formal verb	他要求**予以** 澄清 (He'd demand an explanation)
vq	directional verb	他认识**到** 困难 (He has realized the difficulties)
vb	resultative verb	他看**完** 了电影 (He has seen the movie)
vg	**general verb**	他**喜欢** 踢足球 (He likes playing football)
vn	**nominal verb**	参加我们的**讨论** (Take part in our discussion)
vd	adverbial verb	产量**持续** 增长 (Production increases steadily)

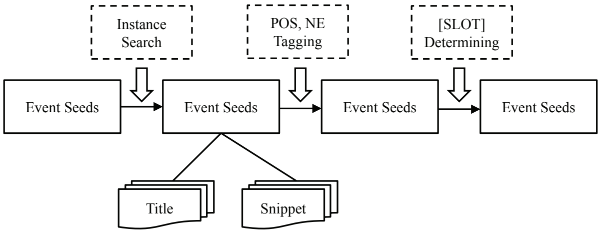

Figure 3 The procedure of pattern learning

sure the accuracy of event seed instances, we stress that the trigger word in candidate event seed must be general verb or nominal verb.

3.4 Instance Collector and Pattern Learner

The Instance Collector is currently implemented in a very simple way. It takes as input event seeds, and then a search engine is used to get documents that contain at least one of the seed words.

Figure 3 depicts the general procedure of pattern learning. Firstly, the collected sentences are used to generate the event instances which are tagged with part of speech (POS) tagging and named entity (NE) tagging. Then the NE labels are replaced by [SLOT] marks.

For example, assuming there is a seed (*Mao Zedong*, *born*, *Xiangtan Hunan*), we can get the related sentence from search engine as shown in Example 1.

Example 1 Mao Zedong was born in Xiangtan, Hunan Province in 1893.

The sentence will be represented as an event instance as shown in Example 2.

Example 2 Mao Zedong/Nh was/v born/v in/p Xiangtan, Hunan Province/Ns in/p 1893/Nr ./wp

Named entity is replaced by [SLOT] marks as shown in Example 3.

Example 3 [SLOT1]/Nh [SLOT2]/Nr born/v in/p [SLOT3]/Ns ./wp

The event instance patterns are generated as mentioned above.

3.5 Soft-Pattern Learner

In order to improve the generalization ability of the extraction pattern, we employ soft-pattern of sequential pattern as the final output of the pattern learner.

In our system, the soft-pattern is constituted by the following symbols.

(1) slots: matching event entities;

(2) tokens: including non named entity and POS tagging;

(3) skips (denoted *): matching zero or more arbitrary tokens.

Soft-Patterns are generalized from the set of event instance patterns. We exploit Soft-Pattern Learning algorithm (SPL) as shown in Algorithm 3.

Algorithm 3 Soft-Pattern Learning algorithm (**SPL**)

```
for event type T do
  for setence pair S_i, S_j from Pattern Instance Set (T) do
    S_i is generated by seed (e_i1, e_i2)
    S_j is generated by seed (e_j1, e_j2)
    if Entity Type (e_i1 = e_j1) and Entity Type (e_i2 = e_j2) then
      Let Pattern = Generalization(S_i, S_j)
      Add Pattern to Soft-Pattern Set (T)
    end if
  end for
end for
```

The core algorithm in *Generalization* function is Best Match algorithm Friedman *et al.* (1977) which is based on LCS (Longest Common Sequence) algorithm Hirschberg (1977). We modify LCS algorithm with matching cost:

(1) Two match units are identical: cost=0;

(2) Two match units share the same NE type but different NE value: cost=5;

(3) If both of two match units are noun, verb, adjective or adverb, compare their labels in a thesaurus TongYiCi CiLin Jiaju *et al.* (1983) (expansion version)[2]. If there are some overlapping labels between two match units, the cost is 5, else cost=10;

(4) No match of two units: cost=10.

After the best match is found, the event instance is converted into a soft-pattern by copying matched identical elements, adding skips and slots. For example, assume that we get the following two sentences by feeding the event seed into the search engine.

周杰伦本年度新专辑《魔杰座》将于10月9日全亚洲同步发行。

→ *This year, Jay Chou's new album "Mo Jie Zuo" will be released all over the Asia on October 9th.*

[2]The dictionary is recorded and expanded by Research Center for Social Computing and Information Retrieval, Harbin Institute of Technology.

Table 3 The example of Best Match algorithm

Pattern Instance 1	Pattern Instance 2	Cost
$[SLOT1]\,/Nh$	$[SLOT1]\,/Nh$	0
本/n (this)		10
年度/n (year)		10
新/a (new)	全新/b (new)	5
专辑/n (album)	大碟/n (album)	5
$[SLOT2]\,/Nb$	$[SLOT2]\,/Nb$	0
将/d (will be)		10
	已/d (was)	10
于/p (on)	于/p (on)	0
$[SLOT3]\,/Nr$	$[SLOT3]\,/Nr$	0
全/a (all)		10
	全球/n (world)	10
亚洲/Ns (Asia)		10
同步/v (simultaneous)	同步/v (simultaneous)	0
发行/v(release)	发行/v(release)	0
。/wp	。/wp	0
		Total: 80

蔡依林全新大碟《花蝴蝶》已于2009年3月27日全球同步发行。

→ Jolin Tsai's new album "Butterfly" was released all over the world on March 27, 2009.

Event instances can be generated as follows by using the approach introduced in Section 3.4.

[SLOT1]/Nh 本/n 年度/n 新/a 专辑/n [SLOT2]/Nb 将/d 于/p [SLOT3]/Nr 全/a 亚洲/Ns 同步/v 发行/v 。/wp

[SLOT1]/Nh 全新/b 大碟/n [SLOT2]/Nb 已/d 于/p [SLOT3]/Nr 全球/n 同步/v 发行/v 。/wp

The best match can be found based on the LCS algorithm. Then, the soft-pattern can be generated as follows.

** [SLOT1]/Nh * * Eb28A01=新/a Dk21B07=专辑/n [SLOT2]/Nb * 于/p [SLOT3]/Nr * * Jb01A10# Ka23A01同步/v Hd13D02=He03B09=发行/v * **

Note that the numbers ("Eb28A01") before "新" etc. are synonym labels from TongYiCi CiLin (expansion version). According to the costs defined above, the Soft-Pattern Learning algorithm is able to find the best generalization of any two event instance patterns. The example of the Best Match algorithm is shown in Table 3.

4 Experiments

To evaluate the effectiveness of our BUEES, we design a set of experiments. We first evaluate the performance of the proposed event type paradigm building approach. Then, we compare our event extraction approach with a state-of-the-art baseline.

4.1 Data Description

We use ACE 2005 corpus for our experiment which is totally the same as the baseline system. The corpus contains 633 Chinese documents which are categorized by three genres: Newswire, Broadcast News and Weblog. We randomly select 558 documents for event type paradigm building and 66 documents as test set for event extraction. We use ACE 2005 event type paradigm as the gold standard paradigm to evaluate our proposed approach.

To evaluate the robust of our approach, we also use two specific domain data sets: Financial News[3] and Musical News[4], collected by ourselves. The domain specific corpus contains 6000 sentences from financial news and 6000 sentences from musical news, respectively. We carefully conducted user studies into two specific domain corpora. For each sentence in the data, two annotators were asked to label and cluster all potential triggers. The agreement between our two annotators, measured using Cohen's kappa coefficient, is substantial (kappa = 0.75). We asked the third annotator to adjudicate the trigger clusters on which the former tow annotators disagreed. Each trigger cluster is used to represent one type of event. All these events construct our final event type paradigm.

4.2 Event Type Paradigm Building

We first propose the task of building event type paradigm in this paper. To evaluate the effectiveness of our approach, we explore several reasonable evaluation metrics and implement a natural baseline method. The detailed introductions are as follows.

Evaluation Metrics.

We adopt *F-Measure* (F) and *Purity* used by Halkidi *et al.* (2001) to determine the correctness of an event cluster:

[3]http://www.10jqka.com.cn/
[4]http://yue.sina.com.cn/

Table 4 Experimental results of event type paradigm building

Method	Corpus	F-Measure	Purity
Baseline	ACE	63.21%	68.17%
Our	ACE	**69.57%**	**70.24%**
Baseline	Financial News	71.52%	74.81%
Our	Financial News	**74.42%**	**76.18%**
Baseline	Musical News	63.21%	68.17%
Our	Musical News	**75.08%**	**80.28%**

$$p(i,r) = \frac{n(i,r)}{n_r},\ r(i,r) = \frac{n(i,r)}{n_i} \tag{3}$$

$$f(i,r) = \frac{2 \cdot p(i,r) \cdot r(i,r)}{p(i,r) + r(i,r)} \tag{4}$$

$$F = \sum_i \frac{n_i}{n} max\{f(i,r)\} \tag{5}$$

$$Purity = \sum_r \frac{n_r}{n} max\{p(i,r)\} \tag{6}$$

where i is the gold standard event seed cluster, and r is the event seed cluster which has the most identical seeds with i. So n_i is the number of seeds in cluster i; n_r is the number of seeds in cluster r; n is the number of all seeds; and $n(i,r)$ is the number of identical seeds between i and r. For every cluster we first compute $p(i,r)$, $r(i,r)$ and $f(i,r)$, then we obtain *F-Measure* and *Purity* for the whole clustering result. Note that the evaluation is based on word instances rather than word types.

Baseline Method.

The task of building a bottom-up event extraction system is first proposed by this paper. There is no existing work to compare with. We build event type paradigm based on clustering event seeds that consist of event trigger, and its corresponding subject and object. A natural baseline method for this problem is only clustering event triggers. A group of triggers are aggregated to a trigger cluster according to their semantic distance, and we view each trigger cluster as one kind of event type. Then all these event types are finally employed to construct an event type paradigm.

Results and Analysis.

We first evaluate the task of event type paradigm building. All the evaluation results are shown in Table 4. There is no previous work on this problem, so the comparison experiments are implemented on our two different approaches. Baseline method only clusters trigger words, however, our approach clusters event seeds.

Table 5 Experimental errors of event type paradigm building

Error types	Proportion
Trigger extraction	33.0%
Trigger ambiguous	28.3%
Trigger filter	19.5%
Others	19.2%

Table 4 shows that the F-Measure score is boosted from 63.21% to 69.57% and the Purity score is boosted from 60.17% to 70.24% by using our approach compared to the baseline method. We analyze the reasons are as follows.

Trigger word itself is not enough for representing event. The trigger and its corresponding subject and object play an important role in the event type discovery algorithm. As referred in Section 3.1, most of trigger words are verb terms. Polysemic verbs are a major issue in Natural Language Processing (NLP) community, such as “to fire a gun” and “to fire a manager”, where “fire” has two different meanings. The state-of-the-art verb sense disambiguation approach Wagner *et al.* (2009) stresses that verbs which agree on their selectional preferences belong to a common semantic class. For example, “to arrest the suspect” and “to capture the suspect”. Hence, our approach can achieve better performance than the baseline method.

We also run the comparison experiment using three different corpora (ACE 05, Financial News and Musical News) to evaluate the robustness and domain adaptiveness of our system. The performances on the specific domain corpora are better than that on the ACE corpus (about 5% absolute improvement on F-Measure and 6%-10% on Purity). The main reason is that the events in specific domain are more specific. In addition, the experiment results on both specific domain corpora can achieve good performance. This indicates that our system is domain independent.

Analysis of Experimental Errors. We first inspect the errors produced by our approach. The errors are mainly caused by the sparse event triggers in corpus. Table 7 shows the distribution of the errors in detail.

After error analysis, we find that the most number of errors are caused by trigger extraction. The main reasons are as follows. First, not all of event triggers are verbs, such as “婚姻(marriage)” for “Life/Marry” event. Although it is reasonable to assume that event triggers are verbs because on average, there are more than 95% event triggers are verbs in ACE 2005 corpus. Second, since only verbs with subject and object are extracted, non-predicate verbs and the verbs without subject/object will not be extracted as candidate triggers. However, the coverage of possible triggers by our trigger extraction algorithm is reasonable good (more than 85%), because most of the trigger words appear repeatedly in the corpus, and their usages are varied. As long as one of their usages is fit for our extraction algorithm, they can be extracted as candidate triggers. Note that the goal of this paper is to build an event type paradigm for new domains. We concern more on the coverage of event type rather than event triggers. The event triggers extracted by us can cover all of event types. We will exploit more effective trigger extraction algorithm in future work.

Trigger ambiguity also accounts for a big proportion of the errors. As discussed above, we cannot judge the event type only by the trigger itself, such as “撤(withdraw/dismiss)” for both “Personnel/End-Position” event and “Movement/Transport” event. This kind of errors can be partially fixed by the PAC model. For example, we cluster “撤职务(dismiss duties)” for “Personnel/End-Position” event and “撤军队(withdraw troop)” for “Movement/Transport” event. These examples indicate that selectional preferences seem to be a reasonable feature even for highly ambiguous verbs like “撤” which encourages to improve argument extraction.

There are still some errors caused by trigger filter. This is mainly due to the fact that not all of triggers are general verb or nominal verb. More effective filter rules will be exploited in future.

Some other errors are caused by NLP tools, such as word segmentation, part-of-speech tagging and dependency parsing. We believe that our algorithms can be improved with the improvement of these NLP tools. In addition, there are about 10% of good event triggers extracted but put into the wrong cluster by trigger cluster.

Experiment with different values of event seed clustering threshold. Different values of threshold

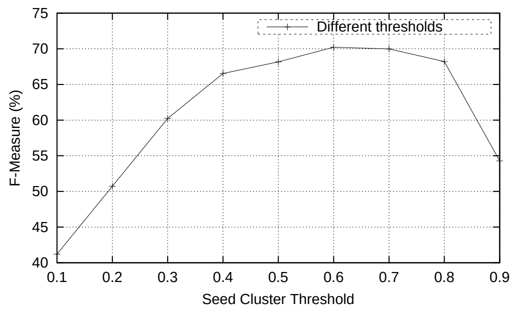

Figure 4 Experimental results of different values of event seed clustering threshold

in Algorithm 2 can dramatically affect the performance of event seed clustering. We experiment with different values of event seed clustering threshold to find the best value. Figure 4 presents the effect on F-Measure of varying the threshold for seed clustering. This figure shows that the best performance of seed clustering can be obtained by selecting the threshold 0.6 for the ACE corpus, 0.7 for the Financial News corpus and 0.9 for the Musical ACE 05 corpus. Figure 4 also suggests that the performance of seed clustering do not dramatically change with the volatility of the threshold from 0.5 to 0.8. Hence, we can firstly set the threshold = 0.6 for new domains.

4.3 Event Type Identification and Argument Recognition

Based on the event type paradigm automatically built by us, we proceed to implement traditional event extraction tasks i.e., event type identification and event argument recognition. We adopt *Precision* (P), *Recall* (R) and *F-Measure* (F) to evaluate the effectiveness of our approach, and compare it with a state-of-the-art event extraction system.

Baseline Method.

We use a state-of-the-art Chinese Event Extraction System as our baseline which is developed by Chen and Ji (2009). This system extracts events with annotated corpus. Its training and testing procedures are follows.

The system combines word-based classifier with character-based classifier. The event types are specified in advance. For every event mention in the ACE training corpus, features are extracted according to some language specific issues. In addition, a set of Maximum Entropy based classifiers are trained:

- Event Type Identification: to distinguish event mentions from non-event-mentions, to classify

Table 6 Overall experimental results

System/Human \ Performance	Event Type Identification			Event Argument Recognition		
	P	R	F	P	R	F
Baseline	65.7%	50.9%	57.4%	53.1%	36.2%	43.1%
BUEES	72.7%	50.7%	59.7%	51.3%	45.9%	48.5%
Human Annotator1	75.2%	74.6%	74.9%	58.6%	60.9%	59.7%
Human Annotator2	82.7%	80.3%	81.5%	66.8%	69.6%	68.2%

event mentions by type;

- Event Argument Recognition: to distinguish event arguments from non-arguments.

In the testing procedure, each document is scanned for instances of triggers from the training corpus. If an instance is found by trigger classifier, the system tries to assign some of the mentions in the sentence as arguments of a potential event mention. The argument classifier is applied to the remaining mentions in the sentence, for any argument passing that classifier; the role classifier will assign a role to it. At last, the system will report the event with type and arguments.

Results and Analysis.

Table 6 shows the overall *Precision* (P), *Recall* (R) and *F-Measure* (F) scores of the baseline system and our BUEES. The table also lists the performance of two human annotators from Chen and Ji (2009).

BUEES outperforms the baseline without annotated corpus. The performance of event type identification is 2.3% higher than the baseline system, and the performance of event argument recognition is 5.4% higher. Several conclusions can be drawn from Table 6.

(1) BUEES does not use any labeled corpus and achieves comparable performance with the baseline system.

(2) Our approach on event type identification enhances the precision (7%) with little loss (0.2%) in recall compared to the baseline method. This recall loss is caused by the limited number of the event seeds. The precision of event type identification is only 2.5% worse than one human annotator, which indicates that the precision of our approach is reasonable good. As our approach is pattern-based, the recall of it needs to be improved.

(3) Our approach on event argument recognition enhances the F-Measure (5%) compared to the baseline method. It indicates that our pattern learning algorithm is effective. We also find that the performance of human annotation on event argument recognition is not high enough (59.7% in F-Measure), and hence it is a difficult problem requires further study.

(4) Table 6 also shows that BUEES system improves the recall (9.7%) performance of event argument recognition over the baseline system. It shows that unsupervised event extraction system can also achieve comparable or better performance with supervised system. However, it is obvious that the system recall is substantially lower than the system precision. Though bootstrapping-based approach can improve recall performance of the soft-pattern by increasing the number of iteration. When the number of iterations increased to a certain value, the recall will not be improved and the precision may be decrease. It is mainly because the more number of iterations the more noise information will be involved in our system.

We should also note that the result of human annotators use the perfect entity mentions, but our system extracts named entities automatically. So the gap is also partially due to wrong named entities.

5 Related Work

Our system is designed to address two issues of event extraction: event type paradigm building and traditional task of event extraction (i.e., event type identification and event argument recognition). The approach of event type paradigm building is related to some prior work on word cluster discovery (e.g. Barzilay and McKeown (2001); Lin and Pantel (2001); Ibrahim *et al.* (2003); Pang *et al.* (2003) Miller *et al.* (2004) Hasegawa *et al.* (2004) Rosenfeld and Feldman (2006)). Most of these works are based on machine translation techniques to solve paraphrase extraction problem. However, several recent researches have stressed the benefits of using word clusters to improve the performance of information extraction tasks. For example, Miller et al. Miller *et al.* (2004) proved that word clusters could significantly improve English name tagging performance. In the same vein, some studies work on the problem of relation extraction (Chambers and Jurafsky (2009, 2011); Poon and Domingos (2008, 2009); Yates and Etzioni (2009)). In these works, "relation words" were extracted and clustered. In this paper, our work confirmed that event seed clusters are also effective for event type paradigm building. The problem of event seeds extraction and clustering is also a challenge problem.

The approach of event extraction is related to a weakly supervised pattern learning algorithm. Yangarber *et al.* (2000) used bootstrapping based method learning simple surface patterns for extracting information. Stevenson and Greenwood (2005) proposed similarity-centric bootstrapping which tried to find patterns with high lexical similarities. Liao and Grishman (2010) filtered ranking for bootstrapping in event extraction. They used two state-of-the-art bootstrapping-based event extraction systems then rank their candidate patterns and accept the top-ranked patterns at each iteration. The BUEES is similar to these systems in the general approach, but its surface patterns allow gaps that can be matched by any sequences of tokens, which make the patterns much more general, and allows to recognize more instances than the simple surface patterns.

Some English event extraction systems based on pattern or machine learning have been reported by researchers (Patwardhan and Riloff (2006); Yangarber *et al.* (2000); Grishman (2001); Ji and Grishman (2008)). However, to our knowledge, the non-English event extraction has rarely been reported by earlier researchers. The baseline system is based on ACE Chinese events. Its contribution is to exploit language specific feature for Chinese event extraction. However, the reported precision of the results was lower than English event extraction. In contrast, the performance of BUEES which is absolutely unsupervised is better than the baseline system and not lower than the state-of-the-art English system.

Web-scale information extraction has received considerable attention in the last few years. Preemptive Information Extraction and Open Information Extraction (Open IE) are the first paradigms that relax the restriction of a given vocabulary of relations and scale to all relation phrases expressed in text (Shinyama and Sekine (2006); Banko *et al.* (2007); Banko *et al.* (2008); Wu and Weld (2010); Etzioni *et al.* (2011); Fader *et al.* (2011)). Preemptive IE relies on document and entity clustering, which is too costly for Webscale IE. Open IE favors speed over deeper processing, which aids in scaling to Web-scale corpora. Comparing with Pre-emptive Information Extraction and Open Information Extraction, the main differences of this paper are on the following aspects. **First,** the previous work mainly focuses on relation extraction, however, this paper aims to extracts events from web corpus. **Second,** Open IE cannot give the event (or relation) type paradigm which is useful for application.

6 Conclusion and Future Work

We have presented the BUEES system which discovers interesting events; learns extraction patterns and extracts the event instances from the Web. BUEES neither relies on manually produced extraction patterns nor on manually annotated training corpus.

BUEES performs by clustering event seeds, generating seed instances and bootstrapping its pattern. Based on general sequential pattern, this paper proposed soft-pattern learning algorithm. Soft-pattern has much greater generalization ability and can reach a high performance for successful bootstrapping.

One of the future work we would like to explore more perfect patterns. Patterns in this paper are very simple that just use several cost values and rules to generate soft-pattern. We want to see if we can achieve higher performance with more complex and perfect patterns.

7 Acknowledgments

We thank the anonymous reviewers for their constructive comments, and gratefully acknowledge the support of the National Basic Research Program (973 Program) of China via Grant 2014CB340503, the National Natural Science Foundation of China (NSFC) via Grant 61133012 and 61202277.

References

Ahn, David, 2006. The stages of event extraction. Proceedings of the Workshop on Annotating and Reasoning about Time and Events, p.1–8.

Banko, Michele, Cafarella, Michael J, Soderland, Stephen, Broadhead, Matthew, Etzioni, Oren, 2007. Open information extraction for the web. IJCAI, **7**:2670–2676.

Banko, Michele, Etzioni, Oren, Center, Turing, 2008. The tradeoffs between open and traditional relation extraction. ACL, **8**:28–36.

Barzilay, Regina, McKeown, Kathleen R, 2001. Extracting paraphrases from a parallel corpus. Proceedings of the 39th Annual Meeting on Association for Computational Linguistics, p.50–57.

Chambers, Nathanael, Jurafsky, Dan, 2009. Unsupervised learning of narrative schemas and their participants. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2, p.602–610.

Chambers, Nathanael, Jurafsky, Dan, 2011. Template-based information extraction without the templates. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, p.976–986.

Che, Wanxiang, Li, Zhenghua, Li, Yongqiang, Guo, Yuhang, Qin, Bing, Liu, Ting, 2009. Multilingual dependency-based syntactic and semantic parsing. Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task, p.49–54.

Chen, Zheng, Ji, Heng, 2009. Language specific issue and feature exploration in chinese event extraction. Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers, p.209–212.

Chinchor, Nancy, Lewis, David D, Hirschman, Lynette, 1993. evaluating message understanding systems: an analysis of the third message understanding conference (muc-3). *Computational linguistics*, **19**(3):409–449.

Ding, Xiao, Qin, Bing, Liu, Ting, 2013. Building chinese event type paradigm based on trigger clustering. Proceedings of the Sixth International Joint Conference on Natural Language Processing, Nagoya, Japan, p.311–319, Available from http://www.aclweb.org/anthology/I13-1036

Dong, Zhendong, Dong, Qiang, 2006. Hownet and the computation of meaning. World Scientific.

Etzioni, Oren, Fader, Anthony, Christensen, Janara, Soderland, Stephen, Mausam, Mausam, 2011. Open information extraction: The second generation. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume One, p.3–10.

Fader, Anthony, Soderland, Stephen, Etzioni, Oren, 2011. Identifying relations for open information extraction. Proceedings of the Conference on Empirical Methods in Natural Language Processing, p.1535–1545.

Friedman, Jerome H, Bentley, Jon Louis, Finkel, Raphael Ari, 1977. an algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software (TOMS)*, **3**(3):209–226.

, 1997. Information extraction: Techniques and challenges.

Grishman, Ralph, 2001. Adaptive information extraction and sublanguage analysis. IJCAI-2001 Workshop on Adaptive Text Extraction and Mining.

Halkidi, Maria, Batistakis, Yannis, Vazirgiannis, Michalis, 2001. on clustering validation techniques. *Journal of Intelligent Information Systems*, **17**(2-3):107–145.

Hasegawa, Takaaki, Sekine, Satoshi, Grishman, Ralph, 2004. Discovering relations among named entities from large corpora. Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, p.415.

Hirschberg, Daniel S, 1977. algorithms for the longest common subsequence problem. *Journal of the ACM (JACM)*, **24**(4):664–675.

Hong, Yu, Zhang, Jianfeng, Ma, Bin, Yao, Jianmin, Zhou, Guodong, Zhu, Qiaoming, 2011. Using cross-entity inference to improve event extraction. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, p.1127–1136.

Ibrahim, Ali, Katz, Boris, Lin, Jimmy, 2003. Extracting structural paraphrases from aligned monolingual corpora. Proceedings of the second international workshop on Paraphrasing-Volume 16, p.57–64.

Ji, Heng, Grishman, Ralph, 2008. Refining event extraction through cross-document inference. ACL, p.254–262.

Jiaju, Mei, Yiming, Zhu, Yunqi, Gao, Hong-Xiang, Yin, 1983. tongyici cilin. *Shanghai Dictionary Publishing Press, Shanghai.*

LDC, 2005. Ace (automatic content extraction) english annotation guidelines for events. http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf.

Lee, Chang-Shing, Chen, Yea-Juan, Jian, Zhi-Wei, 2003. ontology-based fuzzy event extraction agent for chinese e-news summarization. *Expert Systems with Applications*, **25**(3):431–447.

Liao, Shasha, Grishman, Ralph, 2010. Filtered ranking for bootstrapping in event extraction. Proceedings of the 23rd International Conference on Computational Linguistics, p.680–688.

Lin, Dekang, Pantel, Patrick, 2001. Dirt@ sbt@ discovery of inference rules from text. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, p.323–328.

Liu, Ting, Ma, Jinshan, Zhang, Huipeng, Li, Sheng, 2007. subdividing verbs to improve syntactic parsing. *Journal of Electronics (China)*, **24**(3):347–352.

Miller, Scott, Guinness, Jethran, Zamanian, Alex, 2004. Name tagging with word clusters and discriminative training. HLT-NAACL, **4**:337–342.

Miwa, Makoto, Sætre, Rune, Kim, Jin-Dong, Tsujii, Jun'ichi, 2010. event extraction with complex event classification using rich features. *Journal of bioinformatics and computational biology*, **8**(01):131–146.

Pang, Bo, Knight, Kevin, Marcu, Daniel, 2003. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, p.102–109.

Patwardhan, Siddharth, Riloff, Ellen, 2006. Learning domain-specific information extraction patterns from the web. Proceedings of the Workshop on Information Extraction beyond the Document, p.66–73.

Pham, Xuan Quang, Le, Minh Quang, Ho, Bao Quoc, 2013. a hybrid approach for biomedical event extraction. *ACL 2013*:121.

Poon, Hoifung, Domingos, Pedro, 2008. Joint unsupervised coreference resolution with markov logic. Proceedings of the conference on empirical methods in natural language processing, p.650–659.

Poon, Hoifung, Domingos, Pedro, 2009. Unsupervised semantic parsing. Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1, p.1–10.

Riloff, Ellen, 1996. Automatically generating extraction patterns from untagged text. Proceedings of the national conference on artificial intelligence, p.1044–1049.

Ritter, Alan, Etzioni, Oren, Clark, Sam, et al., 2012. Open domain event extraction from twitter. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, p.1104–1112.

Rosenfeld, Benjamin, Feldman, Ronen, 2006. Ures: an unsupervised web relation extraction system. Proceedings of the COLING/ACL on Main conference poster sessions, p.667–674.

, 2007. Event extraction and temporal reasoning in legal documents.

Shinyama, Yusuke, Sekine, Satoshi, 2006. Preemptive information extraction using unrestricted relation discovery. Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, p.304–311.

Soderland, Stephen, 1999. learning information extraction rules for semi-structured and free text. *Machine learning*, **34**(1-3):233–272.

Stevenson, Mark, Greenwood, Mark A, 2005. A semantic approach to ie pattern induction. Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, p.379–386.

Sudo, Kiyoshi, Sekine, Satoshi, Grishman, Ralph, 2003. An improved extraction pattern representation model for automatic ie pattern acquisition. Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1, p.224–231.

Wagner, Wiebke, Schmid, Helmut, Im Walde, S Schulte, 2009. Verb sense disambiguation using a predicate-argument-clustering model. Proceedings of the CogSci Workshop on Distributional Semantics beyond Concrete Concepts, p.23–28.

Wu, Fei, Weld, Daniel S, 2010. Open information extraction using wikipedia. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, p.118–127.

Xiao, DING, Fan, SONG, Bing, QIN, Ting, LIU, 2011. research on typical event extraction method in the field of music. *Journal of Chinese Information Processing*, **2**:005.

Yangarber, Roman, Grishman, Ralph, Tapanainen, Pasi, Huttunen, Silja, 2000. Automatic acquisition of domain knowledge for information extraction. Proceedings of the 18th conference on Computational linguistics-Volume 2, p.940–946.

Yates, Alexander, Etzioni, Oren, 2009. unsupervised methods for determining object and relation synonyms on the web. *Journal of Artificial Intelligence Research*, **34**(1):255.

Yeh, Alexander, Hirschman, Lynette, Morgan, Alexander, 2002. background and overview for kdd cup 2002 task 1: information extraction from biomedical articles. *ACM SIGKDD Explorations Newsletter*, **4**(2):87–89.