Journal of Theoretical and Applied Information Technology
31st March 2017. Vol.95. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EMPLOYING SEMANTIC KNOWLEDGE ON EVENT TRIGGER CLUSTERING

[1]ADE ROMADHONY, [2]DWI H WIDYANTORO, [3]AYU PURWARIANTI

[1,2,3]School of Electrical Engineering and Informatics, Bandung Institute of Technology, Indonesia

E-mail: [1]ade_romadhony@students.itb.ac.id, [2,3]{dwi, ayu}@stei.itb.ac.id

ABSTRACT

Formerly, most of information extraction systems require the predefined template in order to extract the structured information. Extracting information without predefined template leads to the needs of extracting the template. We propose an approach on event template extraction by clustering the event trigger using semantic similarity information from WordNet synset gloss. We demonstrate in the experiment that the semantic information from WordNet synset gloss improves the event trigger clusters quality. The evaluation result shows that the clusters from WordNet synset gloss achieve the top performance on 8 out of 16 event types, outperform the other approaches. The other approaches that we compared on evaluation including: using co-occurrence information only, using relation similarity from UMBC system, and the combination of co-occurrence and relation similarity.

Keywords: *WordNet, Semantic Knowledge, Synset Gloss, Event Trigger, Clustering*

1. INTRODUCTION

Extracting an event means to identify the event triggers and the arguments. Earlier information extraction system including event extraction used manually predefined template to discover the event attributes. Without predefined template, we should discover the event structure automatically.

The definition of an event by LDC is: a specific occurrence involving participant, could be described as a change of state also [1]. The elements of an event consist of:

- Event type:specific event class
- Event trigger: the main word in a sentence that describes an event
- Event arguments: the words in a sentence that describe the event participant

We provide two sentence examples to describe the event extraction:

S1: *An American sailor* has been **jailed** over the murder of a Japanese taxi driver.

S2: Nigerian national *Olatunbosun Ugbogu* was **sentenced** in the *Yokohama district court*.

In the sentence example, the event triggers are denoted with the bold text and the event argument with the italic text. In order to discover the event template, we have to be able to seek the event trigger and the event argument rules. For instance, the template of Arrest-Jail event type will consist of several words that describe the event trigger such as the word jailed in the S1.

How to discover the event triggers of specific event type? The most widely approach to extract the event trigger is by employing the co-occurrence information. The intuition behind it is the words that frequently occur together in a document are representing a similar topic. However, the approach has a limitation because it relies on the word frequencies and the document size. To overcome the limitation on the corpus based approach, several studies combine the information sources from external knowledge [2][3].

Not many previous study on event template extraction employed the semantic knowledge from external resource, such as the use of relation similarity information from UMBC semantic similarity in [4]. The experimental result of clustering event trigger using relation similarity from UMBC API only shows a comparable performance with the one using co-occurrence information. Therefore we propose to exploit the semantic knowledge from WordNet in order to cluster the event triggers. We adapt the Lesk semantic similarity algorithm to catch the semantic relatedness between words. The Lesk algorithm has been used extensively in several tasks including word sense disambiguation [5]. Our hypothesis is that closely related words most likely will appear in the WordNet synset gloss.

Before clustering the event trigger, the system needs to identify the event trigger candidate. The procedure to identify the event trigger candidate usually used syntactic information from the text dependency tree or Semantic Role Labels [6][7][8]. Relational tuple from Open Information Extraction system has also proven as a robust structure representation for several tasks [9]. Previous studies employing relational tuple for event template/schema induction including [10][4]. However, the resulted clusters in Balasubramanian work could not be inferred as specific event type templates, and the work in [4] did not evaluate the clusters as an event type representation semantically.

The task of evaluating event trigger clustering has been a challenging task since there is no exact guideline on what is the correct/valid of an event template. Several competitions such as Text Analysis Conference defined the event type including the event trigger and arguments. However, by observing the human expert when annotating the text with event information, we found that the combination of event trigger word could be very diverse. We presumed that the information from human annotator on real dataset is more accurate than the predefined trigger list provided on the guideline. Therefore we propose the evaluation method by preparing the gold label dataset to be compared with the system clusters.

Our contribution in this study is employing the semantic similarity information from semantic knowledge (WordNet) on event trigger clustering. The semantic evaluation was performed by comparing the clusters with the dataset from human annotated document. The structure of this paper is started by the introduction, continue by related work in the second section. The third section explains the method on acquiring the relation similarity from WordNet. The fourth section contains the event trigger clustering process, followed by the experiment section and conclusion.

2. RELATED WORK

There are two approaches in solving information extraction task: template-based and relation-based. A template defines a specific event type and a set of semantic roles (slots) for entities that participate on the event [6]. The goal of template-based information extraction is slot assignment for every event entities. The template-based information extraction commonly requires a predefined template. On the other hand, the focus of relation-based information extraction is on learning the atomic facts to discover the relation [6].

The advantage of template-based information extraction is the extracted attributes is more complete, whereas the disadvantage is the requirement of the predefined template makes the system usually was performed in limited domain. Defining the template manually is a daunting task, and as the information volume and types grow rapidly, several works have proposed a method to discover the template automatically. Without prior knowledge of corpus domain and the event type contained in the document, the systems try to extract the template.

Early works on template-based information extraction has been performed by [11] and [6]. The Chambers work used clustering approach to create the event trigger clusters and argument clusters. The event trigger clustering was also performed in novel entity discovery [7]. The Chambers work performed the clustering based on the co-occurrence distance function and Lis work performed the clustering based on distributional semantic similarity. Using the co-occurrence information only on clustering the triggers has a shortcoming because it needs a significant volume of documents containing the target event types. One of the solutions that have been implemented in Chambers work is expanding the dataset through information retrieval procedure.

Later development of event template induction is based on generative model approach [12][8][13]. The approach models the document as a bag of entities and tries to do the topic modeling based on the entities chain. All approaches employ a procedure on identifying the event trigger. The structure representations commonly used to identify the trigger candidate are dependency parse and semantic role labels. Only few studies performed the event template extraction or schema generation using relational tuples from Open Information Extraction system [10][4]. However, the relational tuples have shown better performance in several tasks and the comparative study of trigger clustering comparing the trigger from dependency parse structure and the relational tuples showed a comparable performance [4].

The evaluation of event template extraction varies among several procedures: employing the human judgement to analyze the template coherence [10], comparing with other knowledge bases [14] and by performing the information extraction task [6]. Chambers also

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

defined the two types of evaluation through information extraction task: flat mapping and schema mapping. The flat mapping evaluation treats the argument types as an independent element to be evaluated that the procedure does not check the correctness of the schemas. The schema mapping maps the learned slots into one schema only. However, there is no gold label on what are the description of the event trigger and arguments on certain event type. The semantic analysis of trigger clusters in previous study [4] revealed that the best cluster found for each template does not always contain the words that represent the event type appropriately.

3. ACQUIRING RELATION SIMILARITY FROM WORDNET

The words that represent an event template are having tight correlation each other. Synonym and analogy are among the word similarity forms. However, the similarity between words that represent an event trigger is not always on the snynoym or analogy form. Our hypothesis is a group of words having strong correlation, either on synonym or non-synonym form, represents an event template correspondingly.

Turney defines two types of similarity: attributional similarity and relation similarity [15]. Attributional similariy is an inter-attribute correspondence or generally known as synonym. Whereas relation similarity describes inter-relation correspondence.

Other study defines textual similarity type is the UMBC [2]. They provide two types of similarity on their semantic similarity system: concept similarity and relation similarity. Concept similarity defines similarity between nouns or noun phrases. Relation similarity could detect similarity between words with different POSTag. For instance, the phrase marry to has a relation similarity with the phrase is the wife of. We could conclude that concept similarity is synonym and relation similarity defines close relationship between words in a topic.

WordNet is a large knowledge base containing words definition and taxonomy. There are several word similarity methods that were defined based on the information inside WordNet [16]. Among WordNet similarity methods that were depicted in Table 1, the methods that are able to measure the similarity between words with different POSTag are LESK and HSO methods.

Table 1 WordNet Similarity Methods [16]

No	Similarity Method	Description
1	Wu Palmer	Similarity between words are based on their synset depth in WordNet and the depth on LCS.
2	JCN	Similarity based on distance measurement between two words using conditional probability and includes synset child information.
3	LCH	Similarity based on the shortest path between two synsets.
4	LIN	Similarity with JCN, with a little modification.
5	RES	Similarity based on information content from most specific common subsumer.
6	PATH	Similarity based on the number of nodes in the synsets shortest path on Is-A hyrarchy.
7	LESK	Similarity based on the number of overlap words on synset definition.
8	HSO	Similarity based on the synsets path and its number of change direction.

Table 2 Example of Word Synset and Its Gloss

Word	Synset	Gloss
explode	explode, detonate, blow up, set off	cause ~~to~~ burst ~~with a~~ violent release ~~of~~ energy ~~We~~ exploded ~~the~~ nuclear **bomb**
bomb	bombard, bomb	throw **bombs** ~~at or~~ attack ~~with~~ bombs ~~The~~ Americans **bombed** Dresden

This study proposes the adaptation of LESK similarity method to capture the relation similarity between two words.

Table 2 shows an example of the comparison of the explode and bomb gloss/definition. The definition of both words contains overlap words that show the relation similarity between them. We adapt the LESK formula by including the words from the synset members and gloss only.

Journal of Theoretical and Applied Information Technology
31st March 2017. Vol.95. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$RS(w_i,w_j) = \alpha\, gg(w_i,w_j) + \beta\, s(w_i)g(w_j) + \gamma\, g(w_i)s(w_j) \quad (1)$$

The formula of relation similarity between two words w_i and w_j is shown in (1). The overlap between words glossary is denoted by the gg function and the overlap between the words synset and the words glossary are denoted by the $s(w_i)g(w_j)$ and $g(w_i)s(w_j)$ function. We set the $\alpha = 0.6$ and $\beta = \gamma = 0.2$ to make the inter-gloss overlap give higher contribution. On the example, the number of overlap words is 3 and the maximum gloss length = 8 (from the explode word), therefore the value of $gg(w_i,w_j) = 0.375$.

4. EVENT TRIGGER CLUSTERING

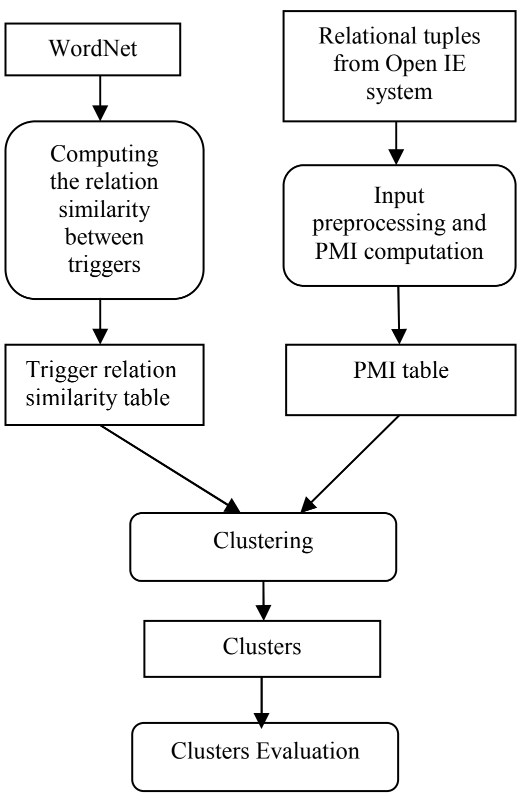

Figure 1 Event Trigger Clustering Process Diagram

Our proposed approach on clustering the event trigger consists of several processes including the preprocessing, preparing the data for the clustering and the evaluation as the post-process. Figure 1 shows the whole process of clustering the event trigger. The input of the clustering process is the relational tuples from Open Information Extraction System. We extracted the relational tuples from the document employing the modified Exemplar system [17][18].

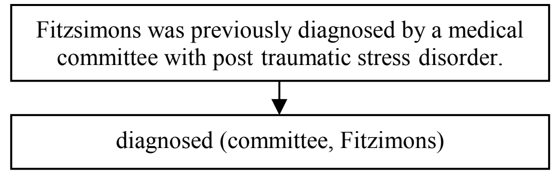

Figure 2 An Example of Relational Tuple Produced from A Sentence

A relational tuple commonly consists of a relation/trigger and two or more arguments. The Exemplar system also output the tuple contains only one argument, which we still use, since the main part to be processed is the relation/trigger. Figure 2 shows an example of relational tuple representation from a sentence.

The next step is the preprocessing of relational tuples [4] and building a table contains the relation trigger co-occurrence information. The co-occurrence information was used to compute the Pointwise Mutual Information (PMI) between two triggers. We followed the method in [6] to compute the PMI.

Besides using the PMI for clustering, we also employ the words semantic similarity derived from WordNet. We computed the value of relation similarity between two triggers with the method that was already described in section 3. Thus we have two informations to be used as similarity function in clustering process.

$$sim(w_i,w_j) = 0.5{*}NPMI(w_i,w_j) + 0.5{*}RS(w_i,w_j) \quad (2)$$

To analyze the effect of the information from the corpus and from the external knowledge, we performed three types of experiment. The experiment types are based on the similarity source: clustering using the PMI only, using the combination of PMI and relation similarity from WordNet and using the WordNet relation similarity only. On combining the PMI and relation similarity, we used the linier combination. Formula 2 described the similarity with the combination of PMI and RS. We set the parameter of PMI and RS component = 0.5 and normalized the PMI to make the contribution of each component proportional. We also performed the clustering using UMBC relation similarity to do the comparison with previous work [4].

We performed the clustering using the agglomerative clustering method, similar with the one used in [6]. The clusters were then evaluated by comparing it with the gold standard clusters. The description of the evaluation procedure is explained in section 5.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. EXPERIMENT

We performed the experiment to study the effect of WordNet semantic similarity information on grouping the words that have tight correlation as the triggers representing an event type. The experiment was performed on ASTRE dataset [19]. To evaluate the clusters semantically, we employed CMatch metric, a clustering evaluation that matches the target cluster and the system cluster [20]. The target cluster is the gold cluster derived from the ASTRE development dataset that has been manually annotated by human.

We conducted several experiments to evaluate the clusters. The experiment settings were set based on the information used in the clustering process. The settings are: using the PMI/co-occurrence information only, using the combination of PMI and WordNet relation similarity information, and using the WordNet relation similarity information only. The additional settings are using the combination of PMI and UMBC relation similarity and using the UMBC only. We also performed several cluster size settings and observed the system cluster with the highest CMatch value on each event type. The analysis is then performed based on the CMatch value and the semantic evaluation of the best cluster members.

5.1 Dataset

Most of the works on information extraction task used MUC34 dataset. The MUC34 dataset contains text from news, radio recording, etc. with the terrorism in South America as the topic. The dataset was annotated by human experts containing the event type and the corresponding slot values information. It has 6 event types, but most researches only focus on 4 types: kidnap, attack, arson, and bombing. The rest two event types were discarded because its low occurrence on the document collection.

Although the MUC34 has been widely used as the standard dataset in information extraction task, there are several disadvantages. Nguyen et.al analyzed the limits of the MUC34 dataset: the size, non-representative, and similarity of roles across templates [19]. The fact that the MUC34 dataset annotation contains no information about the event type trigger made it not suitable with our goal, which will measure the trigger clustering performance externally. We need a gold label dataset to compare the clusters produced by our proposed system with the one that curated by human expert.

T6	Person 107 111	Brit
T7	Person 119 129	Australian
T8	Time 133 144	August 2009
T9	Attack 88 97	murdering
E2	Attack:T9 Target-Arg:T6 Target-Arg2:T7 Time-Arg:T8	

Figure 3: Example of Dataset Annotation

Nguyen et.al proposed a new dataset for open event extraction task (ASTRE dataset). They used the WikiNews article on Law and Justice category as the development dataset and expand it through search engine to produce the training dataset. The development dataset is annotated manually by human expert. The annotators annotated certain words that describe an event type (event trigger), the event arguments, and event argument co-reference. The annotation was conducted using BRAT annotation tools. An example of annotation information is shown in Figure 3. The event is denoted by the E mark. The process of getting the event trigger was started by locating the E entry and then getting the trigger from the event information. On the example we can see that the E2 event is an attack event and the trigger is T9 (murdering).

Based on the annotation, we obtained the event triggers for each event type and used it as the gold label for the clustering evaluation. Table 10 shows the triggers for each event type. We normalized the triggers to the basic form.

The event trigger clustering process was performed using the ASTRE training dataset. The training dataset consists of 1174 documents. Although the dataset creator mentioned that they had cleaned the dataset from HTML or XML tag, we still found few cases. We removed 5 training documents contain HTML or XML tag.

The training dataset has no event type and event trigger annotation. However, since it was obtained based on the development dataset, the event types contained in the text are similar. Thus make the clustering evaluation by comparing the gold labels from the development dataset and the resulted clusters appropriate.

5.2 Result

We tried several cluster size settings ranging from 10-100 and found that the clusters members are unchanged when the cluster size is >= 70 . The clusters were evaluated using cluster comparison approach, CMatch. CMatch measures the cluster quality based on the overlap of the target (golden-truth) cluster Ct and the system cluster Cc.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 3 Max Overlap using PMI Only

Event Type	Cluster Size						
	10	20	30	40	50	60	70
Sentence	0	1	1	2	2	**3**	3
Arrest-Jail	1	1	**2**	2	2	2	2
Attack	2	4	4	4	**5**	5	5
Injure	1	**2**	2	2	2	2	2
Trial-Hearing	1	1	1	1	**2**	2	2
Charge-Indict	0	2	**3**	3	3	3	3
Convict	**1**	1	1	1	1	1	1
Release-Parole	0	**2**	2	2	2	2	2
Appeal	**1**	1	1	1	1	1	1
Extradite	**1**	1	1	1	1	1	1
Sue	**1**	1	1	1	1	1	1
Die	**2**	2	2	2	2	2	2
Pardon	0	**1**	1	1	1	1	1
Fine	**1**	1	1	1	1	1	1
Execute	**1**	1	1	1	1	1	1
Acquit	**1**	1	1	1	1	1	1

Table 4 Max Overlap using Combination of PMI and RS

Event Type	Cluster Size						
	10	20	30	40	50	60	70
Sentence	1	1	**2**	2	2	2	2
Arrest-Jail	**2**	2	2	2	2	2	2
Attack	3	3	4	**5**	5	5	5
Injure	**1**	1	1	1	1	1	1
Trial-Hearing	1	1	**2**	2	2	2	2
Charge-Indict	**1**	1	1	1	1	1	1
Convict	**1**	1	1	1	1	1	1
Release-Parole	**2**	2	2	2	2	2	2
Appeal	**1**	1	1	1	1	1	1
Extradite	**1**	1	1	1	1	1	1
Sue	**1**	1	1	1	1	1	1
Die	2	4	**5**	5	5	5	5
Pardon	**1**	1	1	1	1	1	1
Fine	**1**	1	1	1	1	1	1
Execute	**1**	1	1	1	1	1	1
Acquit	**2**	2	2	2	2	2	2

$$Cmatch\ (Ct, Cc) = \frac{||\in(Ct) \cap \in(Cc)||}{||\in(Ct) \cup \in(Cc)||} \quad (3)$$

The formula of CMatch (3) was adapted from Jaccard coefficient. The overlap value is normalized according to the size of target and system cluster. Without normalization, the system cluster with bigger size has higher probability to get higher CMatch value.

Table 5 Max Overlap using RS Only

Event Type	Cluster Size						
	10	20	30	40	50	60	70
Sentence	**2**	2	2	2	2	2	2
Arrest-Jail	**2**	2	2	2	2	2	2
Attack	**3**	3	3	3	3	3	3
Injure	**1**	1	1	1	1	1	1
Trial-Hearing	**2**	2	2	2	2	2	2
Charge-Indict	**1**	1	1	1	1	1	1
Convict	**1**	1	1	1	1	1	1
Release-Parole	**2**	2	2	2	2	2	2
Appeal	**1**	1	1	1	1	1	1
Extradite	**1**	1	1	1	1	1	1
Sue	**1**	1	1	1	1	1	1
Die	**2**	2	2	2	2	2	2
Pardon	**1**	1	1	1	1	1	1
Fine	**1**	1	1	1	1	1	1
Execute	**1**	1	1	1	1	1	1
Acquit	**2**	2	2	2	2	2	2

Table 6 Max Overlap using PMI+UMBC

Event Type	Cluster Size						
	10	20	30	40	50	60	70
Sentence	1	1	**2**	2	2	2	2
Arrest-Jail	**2**	2	2	2	2	2	2
Attack	7	13	13	15	15	20	**20**
Injure	**1**	1	1	1	1	1	1
Trial-Hearing	**1**	1	1	1	1	1	1
Charge-Indict	**2**	2	2	2	2	2	2
Convict	1	1	**2**	2	2	2	2
Release-Parole	**2**	2	2	2	2	2	2
Appeal	**1**	1	1	1	1	1	1
Extradite	**1**	1	1	1	1	1	1
Sue	2	**3**	3	3	3	3	3
Die	**3**	3	3	3	3	3	3
Pardon	**1**	1	1	1	1	1	1
Fine	**1**	1	1	1	1	1	1
Execute	**1**	1	1	1	1	1	1
Acquit	**1**	1	1	1	1	1	1

Table 3-Table 7 shows the maximum overlap of cluster members using five different similarity settings. Table 8 shows the comparison of the best CMatch from each setting, while the details from various cluster sizes were shown in Table 11-Table 15. We observed the overlap number to analyze the effect of cluster size setting. The best CMatch on an event type from a cluster size setting was picked as the best cluster.

Journal of Theoretical and Applied Information Technology
31st March 2017. Vol.95. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 7 Max Overlap using UMBC Only

Event Type	Cluster Size						
	10	20	30	40	50	60	70
Sentence	**1**	1	1	1	1	1	1
Arrest-Jail	**2**	2	2	2	2	2	2
Attack	4	13	15	15	**18**	18	18
Injure	**1**	1	1	1	1	1	1
Trial-Hearing	**1**	1	1	1	1	1	1
Charge-Indict	**2**	2	2	2	2	2	2
Convict	1	1	**2**	2	2	2	2
Release-Parole	**2**	2	2	2	2	2	2
Appeal	**1**	1	1	1	1	1	1
Extradite	**1**	1	1	1	1	1	1
Sue	**2**	2	2	2	2	2	2
Die	**3**	3	3	3	3	3	3
Pardon	**1**	1	1	1	1	1	1
Fine	**1**	1	1	1	1	1	1
Execute	**1**	1	1	1	1	1	1
Acquit	**1**	1	1	1	1	1	1

Table 8 Comparison of the Best CMatch

Event Type	1	2	3	4	5
Sentence	0.06	0.05	**0.08**	0.05	0.05
Arrest-Jail	0.04	0.07	0.07	0.07	**0.08**
Attack	0.05	0.04	0.04	0.11	**0.13**
Injure	0.10	**0.11**	0.09	0.10	0.10
Trial-Hearing	0.11	0.07	**0.13**	0.07	0.07
Charge-Indict	**0.33**	0.11	0.11	0.11	0.17
Convict	**0.06**	**0.06**	**0.06**	0.05	0.05
Release-Parole	0.14	0.17	0.20	0.13	**0.29**
Appeal	0.25	**0.33**	**0.33**	0.17	0.17
Extradite	**0.17**	**0.17**	**0.17**	**0.17**	**0.17**
Sue	0.07	0.08	0.08	**0.15**	**0.15**
Die	0.15	0.20	**0.22**	0.13	0.19
Pardon	0.09	0.14	**0.20**	**0.20**	**0.20**
Fine	0.17	0.13	0.11	**0.25**	**0.25**
Execute	**0.20**	0.10	0.17	0.04	0.06
Acquit	0.13	**0.17**	**0.17**	0.13	0.13

1: PMI only, 2: combination of RS+PMI, 3: RS only, 4: combination of UMBC+PMI, 5: UMBC only

Based on the overlap number, we can see that the RS function has the best performance on the smallest cluster size setting. The RS function outperforms the other two functions on the cluster size = 10, but increasing the cluster size has no effect on adding the overlap number.

Table 9 The Sparseness of Similarity Value

Similarity	Non Zero Value (%)
Co-occurrence/PMI	23.49
UMBC	8.87
Lesk WordNet	1.75

The PMI+RS combination has the best performance on overlap number and increasing the cluster size gives improvement on it, but the overlap number stopped increasing on the lower cluster size setting than the maximum overlap number on PMI only setting.

The condition shows that the WordNet semantic similarity value represents a strong correlation between semantically related words, but the values is sparse. To support the analysis, we observed the sparseness of similarity value between trigger words using co-occurrence/PMI only, UMBC value only, and Lesk WordNet only. Table 9 shows the comparison of the non-zero similarity value from each similarity method.

Based on the CMatch value, the PMI only setting has the best performance on the charge-indict and execute event type. Whereas the combination setting only has the best performance on the injure event type. The RS only setting is dominating the best CMatch and on several event types the CMatch value is similar with the one on combination setting.

The reason of the varying best setting depending on the event type might be caused by the relation similarity information from the synset gloss was not available and distributed fairly. The RS value is derived from the overlap in word synset and gloss. Meanwhile, the occurrence of the word synset and gloss overlap between two words is less frequent than the corpus-based co-occurrence frequency.

The best performance of RS only setting on small size cluster and the overlap number remains unchanged despite the increasing of the cluster size shows that the RS information has strong effect on defining the relationship between words. However, the RS value availability is limited, could be found only for certain word groups. Hence make the probability of expanding the cluster member is less than the use of corpus-based co-occurrence. The CMatch results show that both co-occurrence and RS component are suitable to be used on different conditions. By using both informations, the cluster quality could be better rather than only using one information type.

We also observed the best CMatch cluster members from each similarity setting as depicted in Table 16. We notify that the cluster members from different similarity setting is different, especially the ones produced by the PMI only and the WordNet or UMBC only. We infer that the use of different similarity source is able to capture different characteristics: the related words that co-occurred frequently in the corpus and the related words that less frequent co-occurred but have tight semantic similarity according to the external knowledge base.

6. CONCLUSION

We present the use of relation similarity from semantic knowledge base (WordNet) on event trigger clustering and perform the cluster evaluation by comparing the system cluster with the gold cluster. The result shows that by incorporating the relation similarity, the cluster quality on certain event types increase. The result fits with our hypothesis that the knowledge base resource could improve the trigger cluster quality when the information from corpus-based co-occurrence is not sufficient.

The overall result also shows that the use of WordNet similarity information outperforms the use of the co-occurrence, UMBC relation similarity, and the combination among them. Based on the result, we show that WordNet as a manually curated knowledge base could serve as the primary resource for grouping the event triggers. Moreover, by analyzing the cluster members produced by different settings, we found that the use of various source for clustering made the complementary clusters. The combination of the trigger clusters from different settings would improve the cluster quality.

For further development, we plan to design a method to detect the event type domain and a method for combining the clustering results. Without knowing the event type domain, all of trigger words will be clustered. Meanwhile, not all trigger words are important to be clustered. Another improvement will be on defining the clustering integration method that will produce better quality event trigger cluster.

Table 10 Event Trigger Gold Cluster

Event Type	Trigger
Sentence	get, face, sentence, subject, receive, seek, eligible, carry, pursue, put, replace, hand, issue, order, in, penalty, warrant
Arrest-Jail	arrest, custody, apprehend, spend, place, remain, include, serve, jail, be, held, serve, imprison, caught, capture, detention, capture, keep, arraign, sweep, operation, hand, behind bars, enter, detain
Attack	assault, beat, scuffle, murder, kill, punch, push, gunpoint, shot, hit, grab, force, fire, blow_up, bomb, murder, bring down, cast, throw, bombard, initiate, attack, bombing, abuse, execution, torture, stab, left, put, rape, jostling, throw, assassination, target, shoot, carjack, overpower, wound, held, shooting, which, fight, displace, molest, act, kidnap, abduct, rob, robbery, touch, assassinate, action, rushed, harm, try, injure, sedate, kick, exploit, crash, threaten, offence, drop, pull, stun, shine, induce, bop, crucify, pin, keep, victim, molestation, war, airstrike, pummeled, incident, wrestled, get_beaten
Injure	suffer, wound, fall, injure, destroy, injury, uninjured
Trial-Hearing	try, trial, hearing, hear, step, reversal, retrial, arraignment, appear, face, retry, court
Charge-Indict	charge, accuse, indict, indictment, face, charge, sentence, consider
Convict	guilty, verdict, convict, pled_guilty, accept, conviction, found_guilty, rule, upheld, conviction, reconvict, decision, find, plead, proven guilty
Release-Parole	release, parole, free, releasing, free, probation, bail
Appeal	appeal
Extradite	extradition, return, handed_over, extradite
Sue	pursue, sue, suit, lawsuit, file, challenge, accuse, case, one, allegation, accusation
Die	die, kill, death, killing, murder
Pardon	pardon, amnesty
Fine	fine, pay
Execute	execute, execution, situation
Acquit	clear, dismiss, rule, guilty, acquit, reach_verdict

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 11 Maximum CMatch using PMI Only (Cluster size 10-70)

Event Type	10	20	30	40	50	60	70
Sentence	0.00	0.05	0.05	**0.06**	0.06	0.05	0.05
Arrest-Jail	**0.04**	0.04	0.04	0.04	0.03	0.03	0.03
Attack	0.03	0.04	0.04	0.03	0.03	**0.05**	0.05
Injure	**0.10**	0.10	0.08	0.04	0.04	0.03	0.03
Trial-Hearing	0.07	0.07	0.07	0.07	**0.11**	0.02	0.02
Charge-Indict	0.00	0.25	**0.33**	0.33	0.23	0.03	0.03
Convict	0.05	**0.06**	0.06	0.06	0.04	0.04	0.04
Release-Parole	0.00	**0.14**	0.07	0.07	0.04	0.04	0.04
Appeal	**0.25**	0.09	0.04	0.03	0.01	0.01	0.01
Extradite	**0.17**	0.08	0.08	0.06	0.06	0.05	0.05
Sue	0.06	0.04	**0.07**	0.07	0.05	0.03	0.03
Die	**0.15**	0.15	0.15	0.15	0.15	0.15	0.15
Pardon	0.00	**0.09**	0.04	0.04	0.02	0.02	0.02
Fine	**0.17**	0.17	0.08	0.03	0.02	0.02	0.02
Execute	**0.20**	0.02	0.02	0.01	0.01	0.01	0.01
Acquit	**0.13**	0.07	0.07	0.07	0.07	0.07	0.07

Table 12 Maximum CMatch using Combination of PMI and RS (Cluster size 10-70)

Event Type	10	20	30	40	50	60	70
Sentence	**0.05**	0.05	0.04	0.04	0.04	0.04	0.04
Arrest-Jail	**0.07**	0.06	0.04	0.04	0.04	0.04	0.04
Attack	**0.04**	0.04	0.04	0.04	0.04	0.04	0.04
Injure	**0.11**	0.08	0.08	0.08	0.08	0.08	0.08
Trial-Hearing	**0.07**	0.06	0.06	0.05	0.05	0.05	0.05
Charge-Indict	**0.11**	0.11	0.06	0.06	0.06	0.05	0.05
Convict	**0.06**	0.04	0.04	0.04	0.04	0.02	0.02
Release-Parole	**0.17**	0.10	0.10	0.10	0.10	0.06	0.06
Appeal	**0.33**	0.08	0.03	0.03	0.03	0.03	0.03
Extradite	**0.17**	0.11	0.11	0.11	0.11	0.11	0.11
Sue	**0.08**	0.06	0.06	0.06	0.06	0.06	0.06
Die	**0.20**	0.19	0.14	0.14	0.14	0.14	0.14
Pardon	**0.14**	0.08	0.01	0.01	0.01	0.01	0.01
Fine	**0.13**	0.05	0.03	0.03	0.03	0.03	0.03
Execute	**0.10**	0.05	0.03	0.03	0.03	0.03	0.03
Acquit	**0.17**	0.12	0.06	0.06	0.06	0.02	0.02

Table 13 Maximum CMatch using RS Only (Cluster size 10-70)

Event Type	10	20	30	40	50	60	70
Sentence	**0.08**	0.05	0.05	0.05	0.05	0.05	0.05
Arrest-Jail	**0.07**	0.06	0.06	0.06	0.06	0.06	0.06
Attack	**0.04**	0.04	0.04	0.04	0.04	0.04	0.04
Injure	**0.09**	0.09	0.09	0.09	0.09	0.09	0.09
Trial-Hearing	**0.13**	0.13	0.12	0.12	0.12	0.12	0.12
Charge-Indict	**0.11**	0.11	0.11	0.11	0.11	0.11	0.11
Convict	**0.06**	0.04	0.04	0.04	0.04	0.04	0.04
Release-Parole	**0.20**	0.20	0.09	0.09	0.09	0.09	0.09
Appeal	**0.33**	0.08	0.08	0.08	0.08	0.08	0.08
Extradite	**0.17**	0.17	0.09	0.09	0.09	0.09	0.09
Sue	**0.08**	0.08	0.08	0.08	0.08	0.08	0.08
Die	**0.22**	0.18	0.14	0.14	0.14	0.14	0.14
Pardon	**0.20**	0.08	0.08	0.08	0.08	0.08	0.08
Fine	**0.11**	0.05	0.05	0.05	0.05	0.05	0.05
Execute	**0.17**	0.08	0.08	0.08	0.08	0.08	0.08
Acquit	**0.17**	0.17	0.11	0.11	0.11	0.11	0.11

Table 14 Maximum CMatch using PMI+UMBC (Cluster size 10-70)

Event Type	10	20	30	40	50	60	70
Sentence	**0.05**	0.05	0.05	0.05	0.05	0.05	0.05
Arrest-Jail	**0.07**	0.07	0.07	0.07	0.07	0.07	0.07
Attack	0.08	**0.11**	0.11	0.11	0.11	0.11	0.11
Injure	**0.10**	0.10	0.10	0.10	0.10	0.07	0.07
Trial-Hearing	**0.07**	0.07	0.06	0.04	0.04	0.04	0.04
Charge-Indict	**0.11**	0.11	0.11	0.05	0.05	0.05	0.05
Convict	**0.05**	0.05	0.04	0.05	0.05	0.05	0.05
Release-Parole	**0.13**	0.13	0.13	0.13	0.13	0.10	0.10
Appeal	**0.17**	0.17	0.03	0.03	0.03	0.02	0.02
Extradite	**0.17**	0.17	0.09	0.09	0.09	0.09	0.09
Sue	**0.15**	0.14	0.14	0.14	0.14	0.11	0.11
Die	**0.13**	0.09	0.06	0.04	0.04	0.03	0.03
Pardon	**0.20**	0.20	0.08	0.07	0.07	0.07	0.07
Fine	**0.25**	0.25	0.17	0.13	0.13	0.03	0.03
Execute	**0.04**	0.02	0.02	0.01	0.01	0.01	0.01
Acquit	0.08	**0.13**	0.13	0.13	0.13	0.09	0.09

Table 15 Maximum CMatch using UMBC Only (Cluser size 10-70)

Event Type	10	20	30	40	50	60	70
Sentence	**0.05**	0.05	0.05	0.05	0.05	0.05	0.05
Arrest-Jail	**0.08**	0.07	0.07	0.07	0.07	0.07	0.07
Attack	0.05	0.11	0.12	0.12	**0.13**	0.13	0.13
Injure	**0.10**	0.10	0.10	0.10	0.10	0.10	0.10
Trial-Hearing	**0.07**	0.07	0.06	0.06	0.03	0.03	0.03
Charge-Indict	**0.17**	0.11	0.11	0.11	0.11	0.11	0.11
Convict	**0.05**	0.05	0.04	0.04	0.04	0.04	0.04
Release-Parole	**0.29**	0.13	0.13	0.13	0.13	0.13	0.13
Appeal	**0.17**	0.17	0.03	0.03	0.03	0.03	0.03
Extradite	**0.17**	0.17	0.09	0.09	0.09	0.09	0.09
Sue	**0.15**	0.13	0.12	0.12	0.12	0.12	0.12
Die	**0.19**	0.09	0.06	0.06	0.06	0.06	0.06
Pardon	**0.20**	0.14	0.14	0.14	0.14	0.14	0.14
Fine	**0.25**	0.17	0.17	0.17	0.08	0.08	0.08
Execute	**0.06**	0.02	0.02	0.02	0.01	0.01	0.01
Acquit	0.08	**0.13**	0.06	0.06	0.04	0.04	0.04

Table 16 Best Cluster Members

Event Type	PMI Only	PMI+RS	RS Only	PMI+UMBC	UMBC Only
Sentence	[attack] [get acquire] [rest] [sentence] [help aid assist] [keep maintain hold]	[spree] [pursue engage]	[arrive] [order] [get acquire] [start begin]	[salute] [hand]	[tackle] [face]
Arrest-Jail	[last] [arraign]	[murder slay] [imprison jail incarcerate] [assassinate]	[murder slay] [imprison jail incarcerate] [assassinate]	[persecute] [imprison jail incarcerate] [round]	[persecute] [imprison jail incarcerate]
Attack	[detonate explode] [try essay attempt] [toll] [influence] [bow] [abolish] [plant] [row]	[throw] [cast]	[throw] [cast]	[save] [spare]	[rape] [abuse maltreat mistreat ill-treat] [assault assail] [molest]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	[blast blare] [tend incline] [preside] [phase] [carjack] [wind wound] [bomb] [bomb]				
Injure	[wind wound] [plant]	[harm] [injure]	[harm] [injure] [infiltrate]	[tighten] [wind wound]	[stab knife] [injure]
Trial-Hearing	[charge] [face] [accuse] [convict] [court]	[hear] [listen]	[retry] [hear] [listen]	[tackle] [face]	[tackle] [face]
Charge-Indict	[charge] [face] [accuse]	[indict] [accuse]	[indict] [accuse]	[tackle] [face]	[suspect] [allege] [charge] [accuse]
Convict	[convict] [court]	[escape] [convict]	[escape] [convict]	[plead] [appeal] [implore beg]	[plead] [appeal] [implore beg]
Release-Parole	[release] [pardon excuse] [hail] [free] [visit]	[release] [fall fell] [free] [drop]	[release] [fall fell] [free]	[release] [write pen] [free] [publish print] [issue]	[release] [free]
Appeal	[sway rock] [appeal]	[plead] [appeal]	[plead] [appeal]	[plead] [appeal] [implore beg]	[plead] [appeal] [implore beg]
Extradite	[extradite] [object]	[extradite] [punch]	[extradite] [punch]	[extradite] [apprehend]	[extradite] [apprehend]
Sue	[charge] [face] [accuse]	[entitle] [file]	[entitle] [file]	[sue action] [file]	[sue action] [file]
Die	[inspire] [anger] [skip] [be_death death s death]	[stab knife] [be_death death s death]	[kill] [massacre] [killing be_killing]	[lose] [die perish]	[kill] [murder slay] [kidnap abduct] [killing be_killing] [execute] [gun]
Pardon	[release] [pardon excuse] [hail] [free] [visit]	[apologise] [pardon excuse] [explain]	[apologise] [pardon excuse]	[forgive] [pardon excuse]	[forgive] [pardon excuse]
Fine	[test] [fine] [content]	[delight please] [hand turn_over] [pay]	[refer mention cite] [remark note] [fine]	[finance] [pay]	[finance] [pay]
Execute	[hang] [execute]	[proceed] [remain stay] [continue] [uphold preserve]	[punish penalize] [execute]	[kill] [murder slay] [kidnap abduct] [killing be_killing] [fire open_fire] [fire be_fire] [execute] [gun] [shoot]	[kill] [murder slay] [kidnap abduct] [killing be_killing] [execute] [gun]
Acquit	[mean] [clear]	[avoid] [earn] [acquit exonerate] [clear]	[avoid] [earn] [acquit exonerate] [clear]	[land] [clear]	[land] [clear]

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REFERENCES:

[1] G. R. Doddington, A. Mitchell, M. A. Przybocki, L. A. Ramshaw, S. Strassel, and R. M. Weischedel, The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation., in *LREC*, 2004, vol. 2, p. 1.

[2] L. Han, A. Kashyap, T. Finin, J. Mayfield, and J. Weese, UMBC EBIQUITY-CORE: Semantic textual similarity systems, *Atlanta, Georgia, USA*, vol. 44, 2013.

[3] M. Yu and M. Dredze, Improving Lexical Embeddings with Semantic Knowledge., in *ACL (2)*, 2014, pp. 545–550.

[4] A. Romadhony, D. H. Widyantoro, and A. Purwarianti, Using relation similarity on Open Information Extraction-Based Event Template Extraction, in *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016.

[5] S. Patwardhan, S. Banerjee, and T. Pedersen, Using measures of semantic relatedness for word sense disambiguation, in *International Conference on Intelligent Text Processing and Computational Linguistics*, 2003, pp. 241–257.

[6] N. Chambers and D. Jurafsky, Template-based information extraction without the templates, in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, 2011, pp. 976–986.

[7] H. Li, X. Li, H. Ji, and Y. Marton, Domain-Independent Novel Event Discovery and Semi-Automatic Event Annotation., in *PACLIC*, 2010, pp. 233–242.

[8] K.-H. Nguyen, X. Tannier, O. Ferret, and R. Besançon, Generative event schema induction with entity disambiguation, in *Proceedings of the 53rd annual meeting of the Association for Computational Linguistics (ACL-15)*, 2015.

[9] G. Stanovsky, I. Dagan, and Mausam, Open IE as an Intermediate Structure for Semantic Tasks, in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2015, pp. 303–308.

[10] N. Balasubramanian, S. Soderland, O. E. Mausam, and O. Etzioni, Generating Coherent Event Schemas at Scale., in *EMNLP*, 2013, pp. 1721–1731.

[11] Y. Shinyama and S. Sekine, Preemptive information extraction using unrestricted relation discovery, in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, 2006, pp. 304–311.

[12] N. Chambers, Event Schema Induction with a Probabilistic Entity-Driven Model., in *EMNLP*, 2013, vol. 13, pp. 1797–1807.

[13] L. Sha, S. Li, B. Chang, and Z. Sui, Joint Learning Templates and Slots for Event Schema Induction, in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 428–434.

[14] N. Chambers and D. Jurafsky, Unsupervised learning of narrative schemas and their participants, in *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*, 2009, pp. 602–610.

[15] P. D. Turney, Similarity of semantic relations, *Computational Linguistics*, vol. 32, no. 3, pp. 379–416, 2006.

[16] T. Pedersen, S. Patwardhan, and J. Michelizzi, WordNet:: Similarity: measuring the relatedness of concepts, in *Demonstration papers at HLT-NAACL 2004*, 2004, pp. 38–41.

[17] F. Mesquita, J. Schmidek, and D. Barbosa, Effectiveness and Efficiency of Open Relation Extraction, in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 447–457.

[18] A. Romadhony, D. H. Widyantoro, and A. Purwarianti, Phrase-based clause extraction for open information extraction system, in *2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2015, pp. 155–162.

[19] K.-H. Nguyen, X. Tannier, O. Ferret, and R. Besançon, A Dataset for Open Event Extraction in English, in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, 2016.

[20] D. H. Widyantoro, Concept drift learning and its application to adaptive information filtering, Texas A&M University, 2003.