Towards Automatic Wayang Ontology Construction using Relation Extraction from Free Text

Hadaiq Rolis Sanabila
Faculty of Computer Science
Universitas Indonesia
hadaiq@cs.ui.ac.id

Ruli Manurung
Faculty of Computer Science
Universitas Indonesia
maruli@cs.ui.ac.id

Abstract

This paper reports on our work to automatically construct and populate an ontology of *wayang* (Indonesian shadow puppet) mythology from free text using relation extraction and relation clustering. A reference ontology is used to evaluate the generated ontology. The reference ontology contains concepts and properties within the *wayang* character domain. We examined the influence of corpus data variations, threshold value variations in the relation clustering process, and the usage of entity pairs or entity pair types during the feature extraction stages. The constructed ontology is examined using three evaluation methods, i.e. cluster purity (CP), instance knowledge (IK), and relation concept (RC). Based on the evaluation results, the proposed method generates the best ontology when using a consolidated corpus, the threshold value in relation clustering is 1, and entity pairs are used during feature extraction.

1 Introduction

As a country rich in cultural diversity, Indonesia certainly has an outstanding wealth of national culture. *Wayang* (shadow puppets performance art) is one instance of Indonesian culture that has cultural values and noble character. Although the stories are generally taken from the Mahabharata and Ramayana books, they involve the wisdom and greatness of the Indonesian culture. *Wayang* shows rely heavily on the knowledge and creativity of the puppeteer (*dalang*). Often, the story and knowledge about the shadow puppets is known only to the puppeteer and not set forth in writing. Such a lack of knowledge transfer process results in a lot of knowledge that is known only by the puppeteer cannot be shared to others, which leads to the loss of cultural richness. The knowledge held by the puppeteer ought to be propagated to future generations in order to be learned and developed.

Information about the shadow puppets can be represented as textual data describing hundreds of characters. Constructing an ontology manually from such a large data source is time consuming and labor intensive.

Work on relation extraction has already been conducted in the past. Initially, supervised learning approaches were used, for example feature-based supervised learning (Kambhatla, 2004; Zhao and Grishman, 2005). Some features that are generally used are words that lie among the entities, the entity type, the number of words between two entities, and the number of entities between two entities. In addition, there are several studies that use kernel-based approach. The kernel $K(x, y)$ defines the similarity between objects x and y in the high-dimensional objects. There are various elements used to construct kernels such as word subsequence (Bunescu and Mooney, 2005) and parse trees (Zelenko et al., 2003; Culotta et al., 2004).

In addition, several studies use semi-supervised learning. DIPRE (Brin, 1998) tries to find the relationship between the author interest and the book he/she had written. Snowball (Agichtein and Gravano, 2000) uses an architecture that is not very different from DIPRE to determine the relationship between an organization and its location. Meanwhile, Knowitall (Etzioni at al., 2005) examines relation extraction in heterogeneous domains of text data

Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) @ EACL 2014, pages 128–136,
Gothenburg, Sweden, April 26 2014.

from the web automatically. Finally TextRunner (Banko at al., 2007) is a system that automatically searches the relationships between entities that exist in a corpus. This method produces a binary relation (e_1, r, e_2) where e_1 and e_2 are entities and r is a relation between them.

Work on automatic ontology construction has been done by several researchers. Celjuska et al. (2004) developed a semi-automatic ontology construction system named Ontosophie. The system generates an ontology with the instances derived from unstructured text. Shamsfard et al. (2004) developed an automatic ontology construction approach which utilizes a kernel based method. Alani et al. (2003) tries to construct an ontology using data from the web. The system, named Artefakt, performs information summarization about the artist. Furthermore, the constructed ontology is used to generate personalized narrative biographies. The system consists of three components, namely knowledge extraction, information management, and biography construction component.

The majority of the information extraction methods mentioned above require reliable NLP tools and resources. Unfortunately these are not readily available for Indonesian, the language our wayang data is in. To overcome this challenge, we employ information extraction methods that only require simple resources such as gazetteers **and stopword** lists, which are potentially used in a variety of problem domains. In this study, we explore methods to automatically construct an ontology using a corpus of wayang character descriptions using relation extraction and clustering. This method requires a gazetteer which contains a list of entities from the text. The entity types that are contained in the gazetteer are the name of the puppet characters, their kingdoms of origin, and their various artefacts such as weapons or spells. We realize our method does not yet fully **constitute** the **develop**ment of a complete

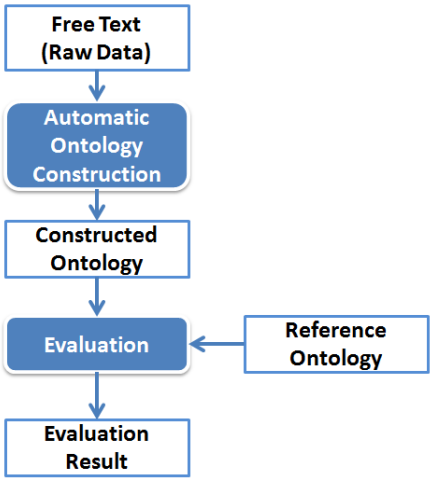

Figure 1. Automatic ontology construction and evaluation stages

ontology, but provides an important step towards that direction, namely the identification of relations to be found within the ontology**.**

2 Automatic Ontology Construction

We aim to automatically build a *wayang* ontology from free text. The information or knowledge that is contained within the text is extracted by employing relation extraction. This method will extract instance candidates that are subsequently clustered using relation clustering. Furthermore, the ontology will be evaluated using a reference ontology to examine the quality of the constructed ontology. The stages of automatic ontology construction and evaluation are depicted in Figure 1.

2.1 Automatic Ontology Construction

During this stage, the system attempts to find all possible relationships that occur between any two entities. These relationships are further analysed to obtain a set of valid relationships between entities. The valid relations will be used to construct the ontology. The ontology construction stage is depicted in Figure 2.

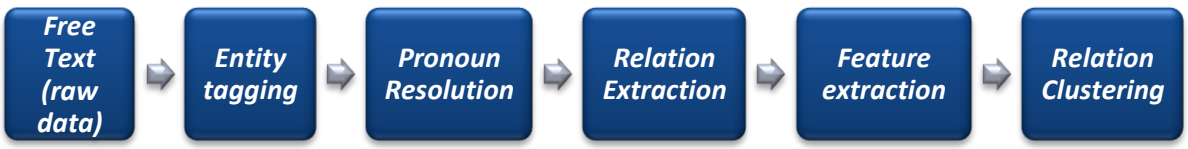

Figure 2. The ontology construction stages

<**Person**> *Anoman* <**/Person**> *kera berbulu putih seperti kapas. Ia adalah anak* <**Person**> *Betara Guru* <**/Person**> *dengan* <**Person**> *Dewi Anjani* <**/Person**>*, seorang putri bermuka dan bertangan kera.* <**Person**> *Anoman* <**/Person**> *juga bernama* <**Person**> *Maruti* <**/Person**>*, karena mempunyai angin, seperti juga Raden* <**Person**> *Werkudara* <**/Person**> *dan oleh karenanya* <**Person**> *Anoman* <**/Person**> *disebut juga saudara* <**Person**> *Werkudara* <**/Person**> *yang berkesaktian angin;* <**Person**> *Anoman* <**/Person**> *juga bernama* <**Person**> *Ramadayapati* <**/Person**>*, berarti yang diaku anak oleh Sri* <**Person**> *Rama* <**/Person**>*;.* <**Person**> *Anoman* <**/Person**> *juga bernama* <**Person**> *Bayutanaya* <**/Person**>*, berarti yang diaku anak* <**Person**> *Betara Bayu* <**/Person**>*;.* <**Person**> *Anoman* <**/Person**> *juga bernama* <**Person**> *Kapiwara* <**/Person**>*,. Bermula* <**Person**> *Anoman* <**/Person**> *hidup pada jaman Sri* <**Person**> *Rama* <**/Person**>*, membela Sri Ramapada waktu kehilangan permaisurinya, Dewi* <**Person**> *Sinta* <**/Person**>**,***yang dicuri oleh raja raksasa Prabu* <**Person**> *Dasamuka* <**/Person**> *dari negara* <**Kingdom**> *Alengka* <**/ Kingdom >**

Figure 3. Tagging result using non-detailed entities

<**BangsaKera**> *Anoman* <**/BangsaKera**> *kera berbulu putih seperti kapas. Ia adalah anak* <**DewaDewi**> *Betara Guru* <**/DewaDewi**> *dengan* <**BangsaKera**> *Dewi Anjani* <**/BangsaKera**>*, seorang putri bermuka dan bertangan kera.* <**BangsaKera**> *Anoman* <**/BangsaKera**> *juga bernama* <**BangsaKera**> *Maruti* <**/BangsaKera**>*, karena mempunyai angin, seperti juga Raden* <**Pandawa**> *Werkudara* <**/Pandawa**> *dan oleh karenanya* <**BangsaKera**> *Anoman* <**/BangsaKera**> *disebut juga saudara* <**Pendawa**> *Werkudara* <**/Pendawa**> *yang berkesaktian angin.* <**BangsaKera**> *Anoman* <**/BangsaKera**> *juga bernama* <**BangsaKera**> *Ramadayapati* <**/BangsaKera**>*, berarti yang diaku anak oleh Sri* <**KerabatAyodya**> *Rama* <**/KerabatAyodya**>*;.* <**BangsaKera**> *Anoman* <**/BangsaKera**> *juga bernama* <**BangsaKera**> *Bayutanaya* <**/BangsaKera**>*, berarti yang diaku anak* <**DewaDewi**> *Betara Bayu* <**/DewaDewi**>*;.* <**BangsaKera**> *Anoman* <**/BangsaKera**> *juga bernama* <**BangsaKera**> *Kapiwara* <**/BangsaKera**>*Bermula* <**BangsaKera**> *Anoman* <**/BangsaKera**> *hidup pada jaman Sri* <**KerabatAyodya**> *Rama* <**/KerabatAyodya**>*, membela Sri Ramapada waktu kehilangan permaisurinya, Dewi* <**KerabatAyodya**> *Sinta* <**/KerabatAyodya**>*,yang dicuri oleh raja raksasa Prabu* <**KerabatAlengka**> *Dasamuka* <**/KerabatAlengka**> *dari negara* <**Kingdom**> *Alengka* <**/Kingdom**>**.**

Figure 4. Tagging result using detailed entities

The raw data is free text that consists of several paragraphs describing short biographies of *wayang* characters. Firstly, the free text is tagged using gazetteer data, i.e. a list of entities contained in the text. Every word contained in the gazetteer will be tagged in accordance to its entity type. The number of entities in the gazetteer is still general. Thus, the entities are subdivided into more specific groups. The entity group is based on Pitoyo Amrih (Amrih, 2011) which consists of 29 groups. In this study we used two tagging methods, i.e. by using a *wayang* entity that has not been detailed and by using detailed entities (based on the type of *wayang* entity). Different tagging treatment was conducted to examine whether this affects the ontology result or not. The example of tagged text using wayang entity that has not been detailed and detailed entities can be seen in Figures 3 and 4.

Subsequently, pronoun resolution is employed to resolve the entity reference of a pronoun. The system will then perform relation extraction by analyzing the words occurring between tagged entities. This process will generate candidate relationship patterns between entities *(X, r, Y)*, where *X* and *Y* are entities and *r* is the textual pattern that defines the relationship between the two entities.

The patterns that are obtained from the previous process are passed on to the next step that is the process of eliminating irrelevant information, so that only valid are used in the next process. It runs as follows:

1. Discard stopwords and honorifics.
2. If there is a comma and punctuation located at the beginning of a pattern then the relation

a) <Person> *Anoman* </Person> anak <Person> Guru </Person>
b) <Person> Anoman </Person> bernama <Person> Maruti </Person>
c) <Person> Anoman </Person> disebut saudara <Person> Werkudara </Person>
d) <Person> Anoman </Person> bernama <Person> Ramadayapati </Person>
e) <Person> Anoman </Person> bernama <Person> Bayutanaya </Person>
f) <Person> Bayutanaya </Person> berarti diaku anak <Person> Bayu </Person>
g) <Person> Anoman </Person> bernama <Person> Kapiwara </Person>
h) <Person> Anoman </Person> hidup jaman <Person> Rama </Person>
i) <Person> Rama </Person>membela Ramapada waktu kehilangan permaisurinya <Person> Sinta </Person>
j) <Person> Sinta </Person> dicuri raja raksasa <Person> Dasamuka </Person>
k) <Person> Dasamuka </Person> negara <Kingdom> Alengka </Kingdom>

Figure 5 The list of patterns as a result of eliminating irrelevant information

is considered valid.

3. Discard punctuation and do the trimming.
4. If there is a pattern that is empty or exceeds 5 words, the pattern is considered invalid.
5. Change the pattern to lowercase.

The result of the data in Figure 3 after this process can be seen in Figure 5.

Subsequently, we perform feature extraction by converting the textual data into matrix form. This matrix contains the occurrence of candidate patterns between all possible pairs of entities. There are two types of feature extraction tried out in this study, i.e. based on entity pairs and entity type pairs. The cell in row *i* and column *k* of this feature matrix is the occurrence frequency of the i^{th} pattern and the k^{th} entity pair. The matrix form of Figure 5 when using feature extraction based on entity pairs is depicted in Figure 6. The next step is to perform relation clustering using semantic relational similarity as a similarity measure in a feature domain. The text patterns contained in each cluster are deemed to represent the same relationship. The clustering process will ignore candidate patterns that occur less than twice in the corpus. The result of this process is a set of clusters that each contains textual patterns that have a greater or equal similarity degree to a given threshold. The pseudocode of this algorithm is depicted in Figure 7.

The generated clusters in this process comprise the relations found in the constructed ontology. The representative pattern, i.e. the candidate pattern that has the highest occurrence frequency within a cluster, will be used as a property that describes the relationship represented by a cluster. Suppose there is a

Pattern \ Entity Pair	A,G	A,M	A,W	A,Ra	A,B	B,Ba	A,K	A,R	R,S	S,D	D,Al
anak	1	0	0	0	0	0	0	0	0	0	0
bernama	0	1	0	1	1	1	1	0	0	0	0
disebut saudara	0	0	1	0	0	0	0	0	0	0	0
hidup jaman	0	0	0	0	0	0	0	1	0	0	0
Membela ramapada waktu kehilangan permaisurinya	0	0	0	0	0	0	0	0	1	0	0
Dicuri raja raksasa	0	0	0	0	0	0	0	0	0	1	0
negara	0	0	0	0	0	0	0	0	0	0	1

A= Anoman, Al = Alengka B = Bayutanaya, Ba = Bayu, D = Dasamuka, G = Guru, K = Kapiwara

M = Maruti, R = Rama, Ra = Ramadayapati, S = Sinta, W = Werkudara,

Figure 6. The matrix form of Figure 5

Relation Clustering Algorithm

Input : *pattern* P = {p_1, p_2, .., p_n}, *threshold* θ
Output: *cluster* C

```
1:  SORT (P)
2:  C ←{}
3:  for pattern p_i ϵ P do
4:      max ← -∞
5:      c* ← null
6:      for cluster cj ϵ C do
7:          sim ← cosine (p_i,c_j)
8:          if sim > max then
9:              max ← sim
10:             c* ← c* ⊕ c_j
11:         end if
12:     end for
13:     if max ≥ θ then
14:         c* ← c* p_i
15:     else
16:         C ← C ∪ ⊕
17:     end if
18: end for
19: return C
```

Figure 7. Relation Clustering Pseudocode

cluster that contains three candidate patterns, e.g. "*anak*" (child of) with an occurrence frequency of 40, "*putera*" (son of) with an occurrence frequency of 30, and "*mendekati*" (come near to), with an occurrence frequency of 3. By using the representative pattern "*anak*" as a property, it is assigned as the relation between pairs of entities found within this cluster. The illustration of the constructed ontology after clustering is depicted in Figure 8.

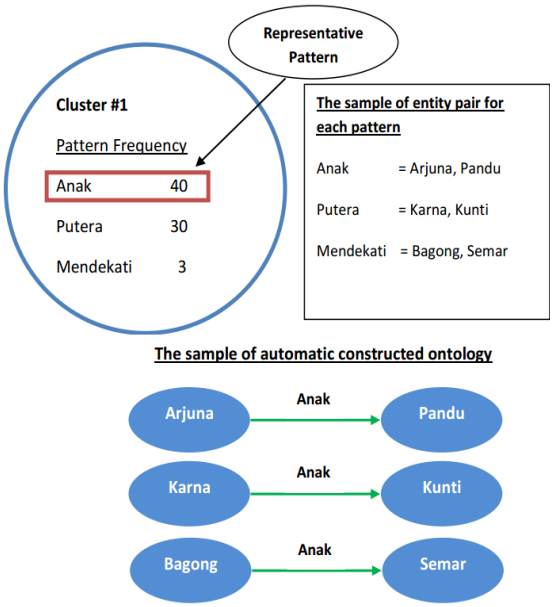

Figure 8. The illustration of constructed ontology subsequent to relation clustering

2.2 Evaluation

2.2.1 Reference Ontology

To measure and ensure that the quality of the constructed ontology is in accordance with what is desired, we evaluate the constructed ontology against a reference ontology. The reference ontology acts like a "label" on the testing data in machine learning. The testing data label used in the evaluation process is used to determine how accurate and reliable the model established by machine learning is in recognizing unseen data. The evaluation process is performed by comparing the relations in the constructed ontology with the labeled testing data. As well as the data labels in machine learning, the reference ontology will be used to test how accurate the system is able to generate ontology from free text.

We define several ontology components that can be obtained from the knowledge of a particular topic. This knowledge is obtained by looking at the types of entities and relations among them. It can also be obtained by looking at the group/category of any entity in the text. Each group/category defines the entity relationship that will occur between one entity to another one.

The ontology components which are defined in the reference ontology are concept and property. An illustration of the relationship between concept and property can be seen in Figures 9 and 10. A concept is something that is described in the ontology and it can be one of: objects, category or class. Concepts in the reference ontology are entities that are incorporated within the gazetteer categories i.e. puppet character, spell, weapons, and nations.

The ontology property describes the relationship between one concept to another. By

Figure 9. The relation between concept and property in ontology

Figure 10. The example of concept and property relation

observing the entity and relationship between them we can obtain the potential properties. For example, there are several entity groups, e.g. puppet character, kingdoms, weapons, and spell. Between each group there is the relationship that may occur. This relationship may occur between entities within the group/category or among entities contained in different group/categories.

In this reference ontology, the authors define certain properties that potentially appear in the text. There are 14 properties which consist of 11 properties describing the relationship between person and person, 1 property describing the relationship between person and country, 1 property describing the relationship between person and weapon, and 1 property describing the relationship between person and spell. The relationship between concepts in the reference ontology is depicted in Figure 11.

2.2.2 Evaluation method

After relation clustering, each cluster is grouped based on the reference ontology property. This grouping is performed based on the synonym of the representative pattern on particular cluster and the property of reference ontology. If the representative pattern does not match (i.e. does not contain a synonym) with the ontology reference property then it is ignored.

In this research we use three evaluation methods i.e. cluster purity, instances of knowledge, and relations concept.

1. **Cluster Purity (CP)**

Cluster purity (CP) is the ratio between the

Figure 11. The relationship amongst concept in a reference ontology

number of representative patterns and the number of all patterns in a cluster. Cluster Purity (CP) calculation ignores singleton clusters, i.e. when there is only one pattern in a cluster. It can be formulated as seen below:

$$\text{CP} = \frac{1}{N}\sum_{1}^{j}\Omega_j$$

where Ω (Ω_1, Ω_2, ..., Ω_j) is the set of representative patterns for each cluster and *N* is the number of patterns in a set of clusters.

Each cluster contains textual patterns and its occurrence frequency. For example, the result of relation clustering can be seen below.

***Cluster* 1**	*anak 32* *putra 12*
***Cluster* 2**	*raja 3*
***Cluster* 3**	*negara 24* *menangis 3*

The CP value of that relation clustering is $\frac{(32+24)}{(32+12+24+3)} = 78.87\%$

2. Instances Knowledge (IK)

Instances Knowledge (IK) evaluation is intended to measure the information degree on each property. There is the possibility that the relationship among two entities is valid but the knowledge therein is not as expected. This evaluation is performed by conducting queries of multiple instance samples. The queries are instance samples that have valid knowledge and are taken randomly from the corpus for each property. It can be formulated as seen below:

$$\text{IK}\ (Prop_i) = Avg\left(\frac{1}{N}\sum_{1}^{j}Q_{j_{Prop_i}}\right)$$

where $Prop_i$ is the i^{th} property, *j* is a query for the i^{th} property, and *N* is the number of queries for the i^{th} property.

For example, there are 6 instances for property *anak* (child of). The instances are *Kakrasana putra* Basudewa , *Werkudara putra Pandu., Kakrasana anak Baladewa, Rupakenca putra Palasara, Basukesti negara Wirata*, and *Dandunwacana negara Jodipati.*

Then there are 5 queries for this property i.e. *Kakrasana putra Basudewa, Werkudara anak Pandu, Arjuna putra Pandu, Rupakenca putra Palasara*, and *Aswatama anak Durna.*

Based on that query, 3 instances are valid (1st, 2nd, 4th) and the rest is invalid. Thus, the IK value is $\frac{3}{5} = 60\%$

3. Relation Concept (RC)

Relation Concept is a measure to examine the valid relations in each property. A valid relation is an instance that has an appropriate relationship with the defined property in the reference ontology. This evaluation can be formulated below:

$$(\text{RC}\ (Prop_i) = \frac{1}{N}\sum_{1}^{j} valid(I_{j_{Prop_i}})$$

where $Prop_i$ is the i^{th} property , $valid(I_{j_{Prop_i}})$ is the valid instances of the i^{th} property, and *N* is the number of pattern.

For example, there are 6 instances for property *anak* (child of). The instances are *Kakrasana putra Basudewa ,Werkudara putra Pandu, Kakrasana anak Baladewa, Rupakenca putra Palasara, Basukesti negara Wirata* and *Dandunwacana negara Jodipati.*

There are 4 instances (1st-4th) that are appropriate and 2 instance (5th-6th) that are not appropriate to property *anak* (child of). So that, the RC value is $\frac{4}{6} = 66.66\%$

3 Experimental Data and Setup

In this research we obtain our raw web data from two separate sources: ki-demang.com and Wikipedia. Ki-demang.com is a website that contains various Javanese culture such as *wayang*, *gamelan* (Javanese orchestra), Javanese songs, Javanese calendar and Javanese literature. Meanwhile Wikipedia is the largest online encyclopedia, it provides a summary of Ramayana and Mahabharata characters.

In this study, we will only use corpora in the Indonesian language, and use 3 types of corpora, namely ki-demang corpus (derived from ki-demang.com), Wikipedia corpus (derived from id.wikipedia.org) and consolidated corpus (combination of ki-demang and Wikipedia corpus).

Ki-demang corpus is containing *wayang* character annotations according to Javanese cultural community. The ki-demang corpus

writing and spelling is not as good as the Wikipedia corpus. Punctuation and spelling errors frequently occur, as well as fairly complex sentence structures. This corpus consists of 363 *wayang* characters; where there are 187 puppet characters that have annotations and 176 puppet characters that do not have annotations.

The Wikipedia corpus has substances of *wayang* character annotation from the Mahabaratha and the Ramayana book and it also contains the description of particular characters in Indonesian culture. The Wikipedia corpus consists of 180 puppet characters, which all have their respective annotations.

The last corpus is a combination of ki-demang and Wikipedia corpus. Merging data from both corpora is expected to enrich the annotation of *wayang* characters. Combining these data led to two perspectives in *wayang* character annotation, which is based on Mahabaratha/Ramayana book and based on the Javanese culture community.

In this study, we will perform some experiments to examine the influence of various parameters. The parameters include the corpus data variety, the threshold value in the clustering process, and the usage of entity pair or entity type pair during feature extraction.

4 Result and Analysis

We conduct experiments for various parameters. The constructed ontology is evaluated using cluster purity (CP), instances knowledge (IK), and relation concept (RC). The experiment results and details of various parameters can be seen in Figures 12 and 13.

For the first experiment we want to evaluate the corpus variation. The objective of this experiment is to find the most representative corpus used in ontology construction. Based on the experiment, when the system is employing entity type pairs in feature extraction, ki-demang corpus has a high CP (76.54%) rate and a lower IK (11.49%) and RC (44.8%) rate. When the CP rate is high, it means that the pattern variation in particular cluster is modest and tends to be a singleton (only one pattern in a cluster). It is the impact of the information homogeneity of ki-demang corpus compared to the other corpora. The IK and RC rate of Wikipedia corpus and consolidated corpus is better than ki-demang corpus. The Wikipedia corpus has better information content compared to the ki-demang corpus, thus the consolidated corpus has a better RC and IK rate compared to individual corpora.

Meanwhile, when the system employs entity pairs during feature extraction stage, the the consolidated corpus has a fairly better result compare to single corpus. It means that the consolidated corpus has richer information than ki-demang or Wikipedia corpus.

The second experiment was conducted to evaluate the threshold value in clustering process. The objective of this experiment is to find the best threshold value for relation clustering. For further analysis in a corpus variation, we used the average value of cluster purity (CP), instances knowledge (IK) and relation concept (RC) for all corpora. When the system employs entity type pairs during feature extraction, the CP rate is 97.15%, IK rate is 49.43%, and RC rate is

Threshold / *Corpus*	1			0.75			0.5			0.25		
	CP	**IK**	**RC**	**CP**	**IK**	**RC**	**CP**	**IK**	**RC**	**CP**	**IK**	**RC**
Ki-demang	**96.53**	**19.54**	**63.95**	**96.52**	**19.54**	**63.95**	**95.88**	**19.54**	**62.02**	**94.27**	**12.64**	**58.83**
Wikipedia	**99.38**	**79.31**	**75.60**	**98.66**	**79.31**	**76.24**	**88.71**	**75.86**	**67.14**	**65.31**	**75.86**	**61.10**
Consolidated	**98.50**	**93.10**	**80.08**	**62.29**	**91.95**	**79.82**	**53.95**	**91.95**	**75.61**	**46.94**	**88.51**	**71.41**

Figure 12. The evaluation result of entity pair usage in feature extraction

Threshold / *Corpus*	1			0.75			0.5			0.25		
	CP	**IK**	**RC**	**CP**	**IK**	**RC**	**CP**	**IK**	**RC**	**CP**	**IK**	**RC**
Ki-demang	**96.30**	**14.94**	**60.02**	**95.80**	**14.94**	**58.45**	**58.74**	**13.79**	**50.05**	**55.34**	**2.30**	**10.70**
Wikipedia	**97.57**	**55.17**	**61.62**	**83.02**	**17.24**	**42.43**	**27.92**	**10.34**	**16.61**	**12.29**	**5.75**	**10.86**
Consolidated	**97.58**	**78.16**	**71.60**	**42.74**	**57.47**	**63.49**	**59.01**	**12.64**	**8.97**	**44.24**	**14.94**	**21.05**

Figure 3. The evaluation result of entity pair type usage in feature extraction

64.41% for threshold value is 1. This result is always higher than using other threshold value.

Hereafter, when the system employs entity pairs during feature extraction, the CP rate is 98.14%, the IK rate is 49.43%, and RC rate is 64.41% for threshold value is 1. Given the experiment result, it is clear that a threshold value of 1 always gives a better result than the other threshold values. The higher pattern similarity in a cluster will yield a better constructed ontology result.

The last experiment was conducted to evaluate the consequence of using entity pairs or entity type pairs during feature extraction to the constructed ontology. For further analysis in a feature extraction variation, we used the average value of cluster purity (CP), instances knowledge (IK) and relation concept (RC) for all threshold value in a clustering process.. Based on the experiment result above, the usage of entity pairs in feature extraction always brings a better result than the entity type pairs. When using entity type pairs in feature extraction, it will reduce some detail of extracted feature. The feature only describes the relationship of entity type, not the entity itself. This leads to suboptimally constructed ontologies.

5 Conclusion

This paper presented a model for automatic ontology construction from free text. Firstly, relation extraction is used to retrieve the candidate patterns. Furthermore, relation clustering is used to group relations that have the same semantic tendency. An experiment has been carried out on various parameters such as on the corpus variety, the threshold value in relation clustering process, the usage of simple process for eliminating irrelevant information and the usage of entity pairs or entity type pairs during feature extraction.

Based on the experimental result, the consolidated corpus (combination of ki-demang and Wikipedia corpus) is most beneficial in ontology construction. By integrating the corpus, it will increase the information quality which yields a better result. Meanwhile for the other parameters, the most beneficial result is obtained when using 1 as a threshold value in clustering process, and using entity pairs during feature extraction. The higher pattern similarity in a cluster will yield a better resulting ontology. Furthermore, simple processing is employed to remove some punctuation, stopwords and honorifics which are a source of noise in the extracted patterns. The usage of entity type pairs during feature extraction will result in reduced or lost detail of pattern features and bring a detrimental consequence to the ontology result.

References

Agichtein, Eugene, & Gravano, Luis. 2000. Snowball: Extracting relations from large plain-text collections. Proceedings of the Fifth ACM International Conference on Digital Libraries,

Alani, Harith, Kim, Sanghee, Millard, David. E., Weal, Mark J., Hall, Wendy, Lewis, Paul. H. and Shadbolt, Nigel. R. 2003. Automatic Ontology-Based Knowledge Extraction from Web Documents. IEEE Intelligent Systems, 18 (1). pp. 14-21,.

Amrih, Pitoyo. Galeri Wayang Pitoyo.com. http://www.pitoyo.com/duniawayang/galery/index.php (accessed at November 4th, 2011)

Banko, Michele, Michael J. Cafarella, Stephen Soderland,Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. InIJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence, pages 2670–2676.

Brin, Sergey. 1998 . Extracting patterns and relations from the world wide web. WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT

Bunescu, Razvan. C., & Mooney, Raymond. J. 2005. A shortest path dependency kernel for relation extraction. HLT '05: Proceedings of the conference on Human LanguageTechnology and Empirical Methods in Natural Language Processing (pp. 724–731). Vancouver, British Columbia, Canada: Association for Computational Linguistics

Celjuska, David and Vargas-Vera, Maria. 2004. Ontosophie: A Semi-Automatic System for Ontology Population from Text. In Proceedings International Conference on Natural Language Processing ICON., Hyderabad, India

Culotta, Aron, McCallum, Andrew, & Betz, Jonathan. 2006. Integrating probabilistic extraction models and data mining to discover relations and patterns in text. Proceedings of the main conference on Human Language Technology Conference of the

North American Chapter of the Association of Computational Linguistics (pp. 296–303). New York, New York: Association for Computational Linguistics.

Etzioni, Oren, Cafarella, Michael, Downey, Doug, Popescu, Anna-Mariana, Shaked, Tal, Soderland, Stephen, Weld, Daniel S., & Yates, Alexander. 2005. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. Artificial Intelligence (pp. 191–134).

Kambhatla, Nanda. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. Proceedings of the ACL

Shamsfard Mehrnoush , Barforoush Ahmad Abdollahzadeh. 2004. Learning Ontologies from Natural Language Texts, International Journal of Human- Computer Studies, No. 60, pp. 17-63,

Zelenko, Dmitry, Aone, Chinatsu, & Richardella, Anthony. Kernel methods for relation extraction. Journal of Machine Learning Research, 2003 .

Zhao, Shubin, & Grishman, Ralph. Extracting relations with integrated information using kernel methods. Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (pp. 419–426, 2005