CHAPTER 5

EXPERIMENTAL SETTING AND RESULTS

This chapter presents the experimental setting and results. The experimental data, which are used in this research, are Thai words and English words. We present the results of our experiments, each of which is designed to compare and contrast the various choices

5.1 Dataset

5.1.1 Thai Dataset

5.1.1.1 Source of the Data

Since there is no available Thai corpus-based, which contains Thai polysemous words in public, so in this research, we use a Thai corpus, which contains Thai polysemous words หัว /hua4/ and เก็บ /kep1/ is created by (Kanokrattanukul, 2001). According to (Kanokrattanukul, 2001) the polysemous words and their contexts were randomly extracted from the corpus of "Bangkok Business" newspaper from November 1st, 1999 to October 31st, 2000 with the total size of 132 MB. The corpus of หัว /hua4/ contains sense of หัว /hua4/ and has 2,200 samples. The corpus of เก็บ /kep1/ contains sense of เก็บ /kep1/ and has 2,200 samples. Each instance of หัว /hua4/ and เก็บ /kep1/ was hand-tagged with its sense defined in the *Thai Royal Institute Dictionary BE (Buddhist Era) 2525* and the information from the data that were not provided in the dictionary. The characteristic of Thai text language is that there is no word boundary in Thai written text. Therefore, the collected data which contained the polysemous words หัว /hua4/ and เก็บ /kep1/ must be word-segmented. The segmentation was processed automatically by (SWATH, 2002) which is a Thai word segmentation program from the NECTEC (SWATH, 2002). The error correction was verified manually based on the context. There are twenty senses of หัว /hua4/ and

nine senses of เก็บ /kep1/ in the corpus. The definitions, derived senses and examples of หัว /hua4/ and เก็บ /kep1/ are presented in Table A.1 and Table A.2 of Appendix A, respectively.

5.1.1.2 Scope of the Data

All occurrences of หัว /hua4/ and เก็บ /kep1/ as an individual word are the data of this study. Since we would like to have all possible meaning of หัว /hua4/ and เก็บ /kep1/, we include หัว /hua4/ and เก็บ /kep1/ that co-occurr immediately with other words. The meaning of each unit does not change from its original meaning. However, some occurrence of หัว /hua4/ and เก็บ /kep1/ are not included in the scope of data. The pattern of data which are beyond our scope are following:

1. หัว /hua4/ and เก็บ /kep1/ co-occurs with other lexical units. They are:

   1.1 Idiom, or idiom-like units.

   Eg. ...สโมสรพากันรุมจีบจน**หัว**บันไดบ้านไม่แห้งทีเดียว...

   1.2 Compound, repetitive and reduplicative words that have unclear meaning. The have the new meanings, or the meaning of each part is totally changed from its original meaning.

   Eg. ...อบายมุขมอมเมาเยาวชนและผู้ชรา**หัว**อสรพิษที่มักนิยมบริโภคหญ้าอ่อน...

   1.3 Proper names.

   Eg. ...มีน้ำป่าจากเขาพุราง เขา**หัว**ล้าน และเขาตอง ได้ไหล่บ่า....

2. หัว /hua4/ which has parts of speech other than noun and เก็บ /kep1/ which has parts of speech other than verb will be excluded. For example:

(i) ...มิหนำซ้ำอาจจะแอบ***ยิ้มหัว***อยู่ในใจว่าเดี๋ยวก็รู้เมื่อไทยเตรียมเสิร์ฟ เมนูอาหาร...

(ii) ...อะไรไว้ ไม่มีทรัพย์สิน มรดกมี***เงินเก็บ***นิดๆหน่อยๆ ก็หมดไปกับการ...

5.1.2 English Dataset

Although this thesis focuses on developing methodology of word sense disambiguation in Thai, we want to support and verify that the research approach can also work well for English words. We use English word in the experiments to show that our method can also work well for English word.

English corpus-based which is distributed by the Computing Research Laboratory (CLR) (crl.nmsu.edu) is used in this work. The data set consists of sentences from the ACL/DCI Wall Street Journal corpus that contains the noun interest. English corpus contains 2,369 sentences which have *interest* word. Table 5.4 presents sense distribution of the training data of *interest* word. Each instance of "interest" has been hand-tagged with one of the six senses defined in the Longman Dictionary of Contemporary English (LDOCE) (Procter 1978). The definition of senses is presented in Table A.3 of Appendix A.

5.2 Experimental Setting

We evaluate our method using sources of sense-tagged corpus. In supervised learning, sense-tagged corpus is used to induce a classifier that is then applied to classify test data. Our approach, however, is purely unsupervised. When we perform the experiments with test data, we need to use sense-tagged corpus to evaluate the maximum accuracy of our approach in which test instance is clustered with its true sense tag.

Two Thai corpus, หัว /hua4/ corpus and เก็บ /kep1/ corpus, contains sentences, which have sense of หัว /hua4/ and เก็บ /kep1/. For training data of หัว /hua4/

and เก็บ /kep1/, we select all 2,200 sentences from each Thai corpus of หัว /hua4/ and เก็บ /kep1/. For the test data, we select 30 % randomly of size of training data of หัว /hua4/ corpus and เก็บ /kep1/ corpus. So each test data of sense of หัว /hua4/ and เก็บ /kep1/ contains 660 sentences. The number of size selection of test data is arbitrary selection. The bigger size of test data is the more coverage of sense distribution that we obtain. Table 5.1 and table 5.2 presents sense distribution of the training data of หัว /hua4/ and เก็บ /kep1/ respectively.

Table 5.1: Sense Distribution of หัว /hua4/

Sense of หัว /hua4/	**No. of Senses**
Head	506
Entity	460
Viewpoint	238
Bulb	159
Brain	138
Front	133
Intelligence	88
Top	77
Titles or names	56
Concentrate	55
Topics	60
Machine part	50
Headline	41
Hair	37
Early hours	41
Chief	30
Emotion	13
Heading	7
Talent	5
Head of coin	6

Table 5.2: Sense Distribution of เก็บ /kep1/

Sense of เก็บ /kep1/	**No. of Senses**
To keep	832
To charge	627
To take	322
To gather	295
To hide	61
To arrange	41
To purchase	20
To kill	10
To pick up	7

English corpus contains 2,369 sentences which have *interest* word. Testing data contains 600 samples which were randomly selected. Table 5.3 presents sense distribution of the training data of *interest* word.

Table 5.3: Sense Distribution of *interest*

Senses	No. of Senses
readiness to give attention	361
quality of causing attention to be given	11
activity, subject, etc., which one gives time and attention to	66
advantage, advancement, or favor	178
a share (in company, business, etc.)	500
money paid for the use of money	1,253

As we describe the details of methodology which is used in this thesis in Section 4.5 of Chapter 4, we can summary the step of our experiments as follows:

Training Phase

Step 1

1.1 Find feature vector as *word vector.* A vector for word *i* is derived from the close neighbors of *i* in the corpus. Close neighbors are all words that co-occur with *i* in a sentence or a larger context.

Example 1.1: Suppose that we have the following sentences which have ambiguated word หัว /hua4/ in the training data. Each word in each sentence is already segmented. Assumed that word หัว /hua4/ has 2 senses. The first sense is *head* and the second sense is *bulb*.

(i) ลักษณะ เป็น โครงหลัก โปร่ง สูง ระดับ **หัว** คน มี ช่อง ให้ เสียบ หนังสือ ซ้อน

(ii) เพิ่มขึ้น เนื่องจาก หอม จะ ลง **หัว** ช้า เพราะ ต้อง หา อาหาร ไป เลี้ยง ดอก หอม

In sentence (i), คน /khon/ (word *i*) co-occurs with มี /mii/ (word *j*). In sentence (ii),หอม /hom/ (word *i*) co-occurs with ช้า /cha/ (word *j*).

Examples of เก็บ /kep1/ which co-occur other words are followed:

(iii) กด /kot1/ มี ปัญหา เลย เกิด ความ **เก็บ** กด เมื่อ มี การ ใช้งาน

(iv) ตัว /tuua/ ยโส หยิ่ง ทะนง ใน ตนเอง ค่อนข้าง **เก็บ** ตัว และ างเหิน จาก คน อื่นๆ

(v) ถูก /thuuk1/ อบต. ห้วยไผ่ ถูก **เก็บ** อีก 1 ขณะที่ บางน้ำเปรี้ยว สุด พิลึก

(vi) ไล่ /lai2/ และ มี การ ไล่ **เก็บ** หุ้น ตัว นี้ จน ดัน ราคา หุ้น

(vii) หุ้น /hun2/ บริษัท ก็ ยัง ได้ **เก็บ** หุ้น ที่ จะ ขายให้ กับ กลุ่ม ทุน

An example of sentence (iii), เก็บ /kep1/ (word *i*) co-occurs with กด /kot1/ (word *j*).

1.2 Form all feature vectors in matrix representation. The feature vector is co-occurrence of a word within windows size ±2. Word vectors are formed by collecting words *i* and words *j* co-occur in a window size. Words that are represented as word vectors are also formed as the dimensions space which represented by the matrix form. This matrix is called *co-occurrence matrix* whose rows and columns represent the words and element entries indicate the number of times of the corresponding word pairs.

With regards to example 1.1, in sentence (i), we have word *i* is *คน* /khon/ and word *j* is *มี* /mii/. In sentence (ii), word *i* is *หอม* /hom and word *j* is *ช้า* /cha/. We can represent two above sentences in co-occurrence matrix form as shown in Table 5.4.

1.3 Represent feature value in term of measures scores. A real valued feature can represent the scores of measures of association which is the log–likelihood ratio. From Table 5.4, each entry of co-occurrence matrix can be computed by the formula of log-likelihood ratio which is described in Sub-section 4.3.1 of Chapter 4.

Table 5.4: Co-occurrence Matrix

	คน	มี	หอม	ช้า	อาหาร
คน	0	1	0	0	0
มี	1	0	0	0	0
หอม	0	0	0	1	1
ช้า	0	0	1	0	1
อาหาร	0	0	1	1	0

1.4 Reduce the dimension of word vectors. We use Singular Value Decomposition (SVD) to reduce the dimension of word vectors. We reduce the matrix to 10% of its original number of columns, or 300 columns.

Step 2

2.1 Create context vector. The context vectors are derived from word vectors. A context vector is the centroid (or sum) of the vectors of the words occurring in the context. After we use SVD reduce the dimension of matrix, we can compute the context vector from reduced matrix. For example, suppose the co-occurrence matrix in Table 5.4 is reduced the dimension by SVD and had word vectors คน, มี, ช้า, อาหาร represented by v(คน), v(มี), v(ช้า), v(อาหาร) respectively. The context vector of sentence (i) is summation of v(คน) and v(มี) and the context vector of sentence (ii) is summation of v(ช้า) and v(อาหาร).

Step 3

3.1 Create sense vector. The sense cluster can be created by grouping similar contexts. The centroid of cluster is the representation of a sense. Sense representations are computed as groups of similar contexts. This set of context vectors is then clustered into a predefined number of clusters or context groups. An example is shown in Figure 5.1, the clustering step has grouped context vectors. In this thesis, the number of clusters is from manual

inspections of cluster results and is not fully automatic. This thesis approach takes explicitly specified to create the number of clusters which are 20 clusters of หัว /hua4/, 9 clusters of เก็บ /kep1/ and 6 clusters of *interest*. These predefined cluster numbers are from the number of the training data senses distribution which is the result of word sense analysis.

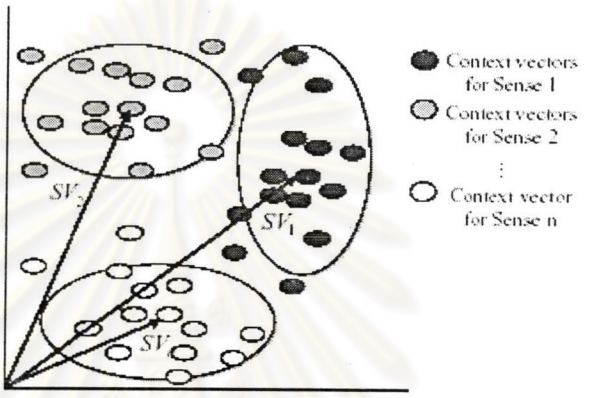

Figure 5.1: Sense Vector

Test Phase

Compute similarity measure. Having created the feature vectors for each of the selected feature word and get sense vector in the training data, we will build a context vector for each target word in the test data. The similarity measure between context vector of test data and sense vector which is created from training data is computed via cosine similarity measurement. The context vector which is closest to the sense vector is disambiguated and is assigned to belong to that sense cluster as it is illustrated in Figure 5.2. The predefined threshold value is defined to measure the similarity between the context vector of test data and sense vector. We use arbitrarily a cosine threshold of 0.5. The performance of a clustering algorithm can be evaluated using sense-tagged data in which context vector of test data being clustered are compared with their true sense tags.

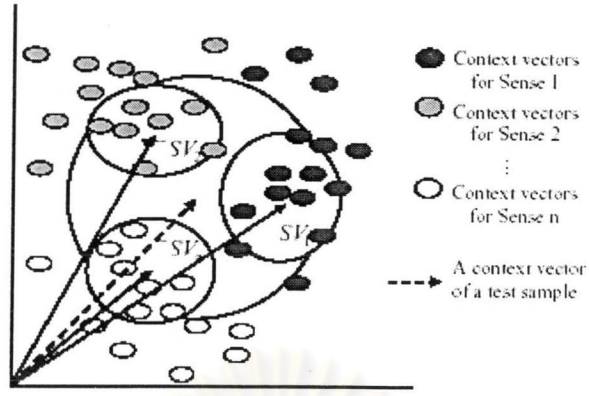

Figure 5.2: Compare context vector of test data with sense vector

5.3 Experimental Results

Table 5.5 and Table 5.6 show experimental results compared with the baseline system for the disambiguation of หัว /hua4/ and เก็บ /kep1/ respectively. Table 5.7 shows experimental results compared with the baseline system for the disambiguation of *interest* word. The first column of Table 5.5, Table 5.6 and Table 5.7 shows sense definitions. The precision and recall are shown in the second column and the third column of Table 5.5, Table 5.6 and Table 5.7 respectively. The final column of Table 5.5, Table 5.6 and Table 5.7 shows F measure. In Table 5.5, considering the average F measure, it is showed that the polysemous word หัว /hua4/ has 70.96% correctness. When the average precision 71.27% is compared with the baseline that is 23.00% correctness, our approach can outperform the majority baseline system which is the highest frequency sense of word หัว /hua4/. Likewise, the average F measure in Table 5.6 shows that the polysemous word เก็บ /kep1/ has 74.36%. correctness. The average precision 75.58% against with the baseline that is 37.81%. Our approach can outperform the majority baseline system. With regards to an English ambiguous word, the average F measure in Table 5.7 is 70.71% correctness. Our approach can outperform the majority baseline system since the average precision 67.05% against with the baseline that is 52.89%.

Table 5.5: Accuracy of disambiguation of หัว /hua4/.

Sense Definitions	Precision (%)	Recall (%)	F-Measure (%)
Brain	77.5	89.5	83.07
Bulb	75.9	67.8	71.62
Chief	76.5	70.4	73.32
Concentrate	76.5	69.6	72.89
Early hours	70.4	81.5	75.54
Emotion	72.8	71.2	71.99
Entity	61.2	70.8	65.65
Front	69.8	65.8	67.74
Hair	78.7	78.6	78.65
Head	62.4	58.4	60.33
Head of coin	60.1	74.1	66.37
Heading	70.2	56.3	62.49
Headline	74.5	85.7	79.71
Intelligence	73.2	49.5	59.06
Machine part	77.2	57.6	65.98
Talent	63.7	74.9	68.85
Titles or names	70.9	62.7	66.55
Top	71.3	85.4	77.72
Topics	77.2	67.8	72.20
View point	65.4	75.6	70.13
Average	**71.27**	**70.66**	**70.96**

Baseline= 23.00 %

Table 5.6: Accuracy of disambiguation of เก็บ /kep1/.

Sense Definitions	Precision (%)	Recall (%)	F-Measure (%)
To arrange	80.7	78.1	79.38
To charge	72.1	85.9	78.40
To gather	79.8	65.7	72.07
To hide	76.5	83.2	79.71
To keep	70.2	63.7	66.79
To kill	75.8	66.1	70.62
To pick up	79.8	84.9	82.27
To purchase	73.5	60.7	66.49
To take	71.8	70.4	71.09
Average	**75.58**	**73.19**	**74.36**

Baseline = 37.81 %

Table 5.7: Accuracy of disambiguation of *interest*

Sense Definitions	Precision (%)	Recall (%)	F-Measure (%)
readiness to give attention	74.3	85.7	79.59
quality of causing attention to be given	71.2	74.5	72.81
activity, subject, etc., which one gives time and attention to	65.2	82.7	72.91
advantage, advancement, or favor	65.7	79.5	71.94
a share (in company, business, etc.)	50.7	65.7	57.23
money paid for the use of money	75.2	60.7	67.18
Average	**67.05**	**74.8**	**70.71**

Baseline = 52.89%

5.4 Discussions

As a preliminary result, it can be observed if the algorithm is tested with polysemous words which have fewer senses, its performance should be better. Words หัว /hua4/ and เก็บ /kep1/ are more polysemous words since they have 20 senses and 9 senses respectively. By testing with polysemous words which has fewer senses, the algorithm will have fewer problems with word form. Firstly, words which have fewer senses have clearer sense indicators, as their senses are not closely related, so different senses occur with totally different context. Secondly, words which have fewer senses require fewer training data. This is because words which have fewer senses have fewer numbers of senses, so it is easier to find many samples of all senses in a small size of data.

5.5 Result Comparison with Other Thai Word Sense Disambiguation Methodology

Since there are very few research works which pays attention to Thai word sense disambiguation and it lacks of Thai repository of word sense corpus, this thesis will give an example of research work which worked on Thai word sense disambiguation. According to (Kanokrattanukul, 2001), this research work aims to

develop a prototype of word sense disambiguation program in Thai (หัว /hua4/, เก็บ /kep1/) by using the decision list algorithm. The methodology is based on supervised learning technique. The precision rate of หัว /hua4/ is 87% while the precision rate of เก็บ /kep1/ is 80.25%. Table 5.8 shows the comparison the average precision percentage of sense disambiguation of หัว /hua4/ and เก็บ /kep1/ between (Kanokrattanukul, 2001)'s work and this thesis work, although the methodology approaches are different and it actually cannot compare between two methods.

Table 5.8: Comparison Results of Average Precision rate of disambiguation of หัว /hua4/ and เก็บ /kep1/ between Decision List and our work.

	Precision Rate (%)	
	Decision List (Kanokrattanukul, 2001)	**Our Work**
Polysemous word หัว /hua4/	87.00	71.27
Polysemous word เก็บ /kep1/	80.25	75.58

5.6 Further Investigated Parameters

However, there are other parameters that are out of thesis scope and have not been extensively studied but these parameters effect to the performance of algorithm. They parameters should be conducted in study detail as the future works. These parameters can be discussed below.

The Window size

Since this thesis uses feature as it is co-occurrence within a small window ±2. Co-occurrence feature is a feature vector with small size of experimental data, these parameters can effect to these experimental results of word หัว /hua4/, เก็บ /kep1/ and *interest* word. Other parameters such as other features, vary windows sizes and larger data size can have some effect on the algorithm performance.

Sample Size of Test Dataset Selection

The accuracy of disambiguation of หัว /hua4/, เก็บ /kep1/ and *interest* word are 71.27%, 75.58% and 67.05% respectively. These experimental results are based on the experimental setting of sample size of test dataset. The test data size is 30% of training size as we select arbitrary the number of test data size. If the test data size is larger size, the test data will have sentences which cover every possible sense to be tested. It will yield better result.

Numbers of Clusters

The number of clusters can be defined in any numbers but the numbers of clusters can effect to the result. If the given number of clusters is more or less than true sense cluster, the accuracy of disambiguation will be less. This is a significant challenge in any clustering task is to determine how many optimal number of clusters should be created automatically for the given

This thesis approach takes explicitly specified to create the number of clusters which are 20 clusters of หัว /hua4/ 9 clusters of เก็บ /kep1/ and 6 clusters of *interest* respectively and they are from manual inspections of cluster results and are not fully automatic. These cluster numbers are inspected from sense distribution of หัว /hua4/, เก็บ /kep1/ *interest* respectively which is shown in Table 5.4, Table 5.5 and Table 5.6. It is possible that a context vector of test data can be assigned multiple possible clusters. However, we used K-Means which is the hard clustering in our experiments and hence do not classify any instance into more than one clusters. Moreover, every target word in our evaluation data is tagged with only one sense.

Threshold Similarity

The predefined threshold value is defined to measure the cosine similarity between the context vector of test data and sense vector. The similarity measure will determine context vector of test data should be assigned to which sense cluster. If the threshold value is higher, the more overlap between context vector of test data and sense vector. This will effect that a context vector of test data can be assigned multiple possible clusters. If the threshold value is lower, there is less

overlap between context vector of test data and sense vector. This will effect that a context vector of test data cannot be assigned to any clusters. We use arbitrarily a cosine threshold of 0.5 since we assume that it will not effect to any cluster assignment as we count this threshold of 0.5 is the average of cosine value. More experiments with varying threshold values should be further conducted to exam which threshold value yields the best result.