Lexicon Acquisition for the English and Filipino Language

Jason Oliver Lat
Spencer Troy Ng
Kenneth Sze
Gene Derrick Yu
Nathalie Rose T. Lim

De La Salle University
2401 Taft Avenue
1004 Manila, Philippines
(632) 524-0402

{ricochet_008, speng_14, rushbaal, doughboy_derrick}@yahoo.com, limn@dlsu.edu.ph

Abstract

A system was developed using the correlation formula in Kaji's[4] algorithm complemented with other components, such as a part of speech tagger and a stemmer, to extract bi-lingual terms from English and Filipino comparable corpora. The system was tested on two main corpora, The Alchemist[2], and Jose Rizal: Life, Works and Writings of a Genius, Writer, Scientist and National Hero[8] – both of which were tested independently; and were found to have results of at most 50% accuracy. This paper discusses the components and illustrates the issues with regards to bilingual lexicon extraction for the Filipino and English language.

1. Introduction

Recent researches and developments on Natural Language Processing for Machine Translation Systems have been successful in many different languages. These languages include English, Spanish, Arabic, Nihongo, Mandarin and many more; however, there has been little research with regards to the Filipino language. This paper discusses the Automatic English and Filipino Lexicon Builder (AEFLex) system, which is a lexicon extraction system designed for the English and Filipino language, as well as results of the extraction process.

The system adopted algorithms used in other systems where they underwent additional variations and components. An initial lexicon taken from the IsaWika![1], with an entry of 22,940 English to Filipino words and 19,980 Filipino to English words was used. Lists of function words were also used by the system. The list of English function words was taken from Cornell University[3], and the list of Filipino function words was lifted from IsaWika[1] where they were chosen based on their part of speech tags. A function word is a word that is not a noun, verb or adjective.

2. Architecture

The AEFLex System, shown in figure 1, has five main components. These components are Preprocessor, Co-occurrence Analyzer, Computation of Similarity/Correlation, Selection of highly Similar/Correlated words, and the Lexicon Editor.

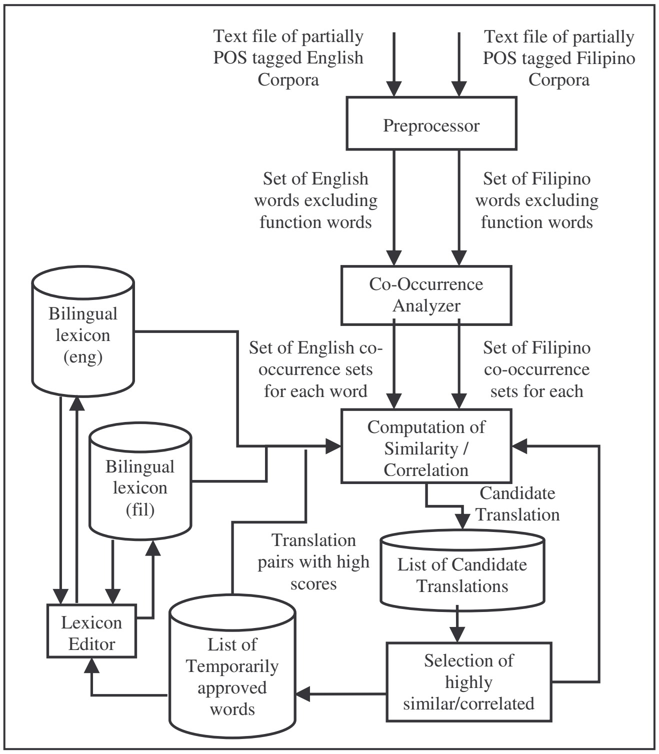
Text file of partially POS tagged English Corpora
Text file of partially POS tagged Filipino Corpora
Preprocessor
Set of English words excluding function words
Set of Filipino words excluding function words
Co-Occurrence Analyzer
Bilingual lexicon (eng)
Set of English co-occurrence sets for each word
Set of Filipino co-occurrence sets for each
Computation of Similarity / Correlation
Bilingual lexicon (fil)
Candidate Translation
Translation pairs with high scores
List of Candidate Translations
Lexicon Editor
List of Temporarily approved words
Selection of highly similar/correlated

Figure 1. The Architectural Design of AEFLex

The system starts by accepting the text files of the English and Filipino corpora. The corpora are then stripped off of function words and named entities. It then undergoes a series of preprocessing before applying the lexicon extraction algorithm. After the preprocessing, the words then go through the co-occurrence analyzer. This component would determine co-occurrence sets by examining the context of each word in the corpora. By examining the bilingual lexicon and co-occurrence sets of each term, the correlation or similarity scores of candidate translation pairs will be computed. Finally, when the scores are already completed, the resulting list is manually checked to see which words obtained the correct translation.

2.1. Preprocessor

In the preprocessor, the insignificant words in the input corpora are removed. These words include function words, words with apostrophe (‘) and named entities. They are removed because they do not contribute to the similarity measurement scores and can only decrease the accuracy. Each word can be stemmed and/or tagged with its part of speech (POS) using a look-up table based on the initial lexicon. A stemmer is also used to find the root word of each word. Figure 2 shows the procedure of the preprocessor.

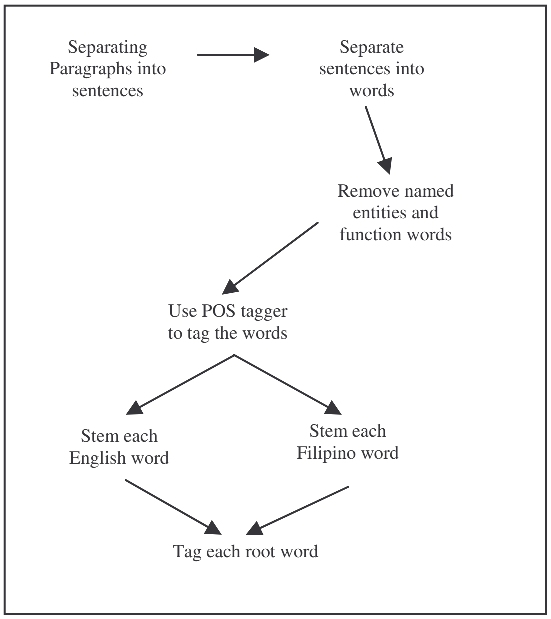
Separating Paragraphs into sentences
Separate sentences into words
Remove named entities and function words
Use POS tagger to tag the words
Stem each English word
Stem each Filipino word
Tag each root word

Figure 2. Preprocessor

The system begins by loading the text file of the English and Filipino corpora, where it will tag the corpora by referring to the initial lexicon database. The tagger used for the system simply uses the initial lexicon database as a lookup table. It repeatedly searches the lexicon for the words in the corpora, starting from the first word to the last, and returns a result each time. If the result returns exactly one match, then the part of speech tag attached to the word (in the lexicon) is assigned to the word. If, however, the word does not exist or that it has more than one part of speech tag, “nopos” is assigned.

After tagging each word in the corpora, they will undergo stemming. The process of stemming starts by getting each word in the corpora, and repeatedly removes prefixes and/or suffixes the word might have; resulting in having a corpus of mostly root words. The English stemmer used in the system is Porter’s stemming algorithm[6]. The Filipino stemmer used in the system was made by following the rules of Filipino word structures found in the English-Tagalog Vocabulary[5].

Finally, these root words will again undergo another tagging process. This second tagging process will try to tag words with “nopos” tags. The purpose of this second tagging process is to increase the accuracy of tags since a word may not exist in the lexicon but its root word might.

Example:

Corpora: The car was running fast on the street.

All of function words will be removed first. The words 'the', 'was' and 'on' will be removed. Leaving the words car, running, fast and street.

Assuming the word "fast" has two entries in the lexicon – "adjective" and "adverb"; "street" and "car" each have only one entry – "noun"; the word "running" does not exist in the lexicon.

After the first tagging process, the word "street" and "car" will have a "noun" tag, the words "fast" and "running" will have "nopos" as their tag. The stemming process will only accept words tagged as "nopos". The words running and fast will undergo stemming. After stemming the word running will be changed to run while the word "fast" will have no change. In the second tagging process, only the words run and fast will be tagged again. “fast” will still be tagged as “nopos”, since it has two tags; but the word “run” will be assigned a “verb” tag.

2.2. Co-occurrence Analyzer

After preprocessing, the corpora will be passed to the co-occurrence analyzer. This component determines the frequency of each word co-occurring with another word. The basis of the collocates (co-occurring words) is a window size. The system uses a default window size of 2 (meaning 2 words that come before it and 2 words that come after it). These words are passed on to the next process.

Given the example text from Section 2.1 and assuming “street” is the unknown word looking for its translation, and all other words are known or can be found in the initial lexicon, the unknown words will have the following co-occurrence sets:

English:

Content Word	POS	Co-occurrence set	Frequency
street	noun	running	8
		fast	5

Filipino:

Content Word	POS	Co-occurrence set	Frequency
kalsada	noun	nabangga	3
		poste	2

		mabilis	7
		tumatakbo	5

Content Word	POS	Co-occurrence set	Frequency
ambulansya	noun	mabilis	4
		tumatakbo	3
		papunta	8
		nabanggang	5
Content Word	**POS**	**Co-occurrence set**	**Frequency**
nabanggang	nopos	mabilis	1
		tumatakbo	2
		papunta	6
		ambulansya	2

The word "street" co-occurs 8 times with "running", and 5 times with "fast." Three candidate translations were chosen from the sample Filipino corpus. The first candidate translation is the word "kalsada." It co-occurs 3 times with "nabangga", 6 times with "poste", 7 times with "mabilis", and 5 times with "tumatakbo." The second candidate is the word "ambulansya." It co-occurs 4 times with "mabilis," 3 times with "tumatakbo", 8 times with "papunta", and 5 times with "nabanggang." Finally, the third candidate translation is "nabanggang." It is included in the candidate translation since it has no part of speech tag, and all words without a part of speech tag will be included in all candidate translations. It co-occurs 3 times with "mabilis", 3 times with "tumatakbo", 6 times with "papunta", and 7 times with "ambulansya."

2.3. Computation of Similarity/Correlation

This computes for the score of each word and its candidate translation. The system uses the formula for correlation used by Kaji[4].

After processing input corpora, the system eliminates insignificant terms based on a factor/value given by the user. Insignificant co-occurring terms are also eliminated based on a factor/value asked from the user. An insignificant term is identified based on the number of occurrence of the specific term.

Following the elimination of insignificant terms, the system then processes the co-occurring words for both corpora based on the window size specified. These co-occurring words are extracted and compared to the candidate translations.

From the example text in Section 2.2, all words in the co-occurrence set will be translated to its Filipino translation by consulting the bilingual lexicon.

"street"	"kalsada"	"ambulansya"	"nabanggang"
running → tumatakbo (8)	tumatakbo (5)	tumatakbo (3)	tumatakbo (3)
fast → mabilis (5)	mabilis (7)	mabilis (4)	mabilis (3)

Finally, the frequencies of the co-occurrence sets for both English and Filipino word would be compared and the similarity scores would be computed. The formula used for computing similarities between these terms is from the formula that Kaji used for lexicon extraction[4]:

$$R\ (sw, tw) = \frac{|\ C(sw) \cap C(tw)\ |}{|\ C(sw)\ | + |\ C(tw)\ | - |\ C(sw) \cap C(tw)\ |}$$

where sw is the source word,
tw is the target word,
C(X) is the co-occurring set of X,
R(X, Y) is the similarity between X and Y, and
| X | is the occurrence of X.

The intersection is computed by adding the lowest value on the two words. For example, the intersection (∩) of 'street' and 'kalsada' is computed as follows:

running -> tumatakbo(8) = tumatakbo(5)
fast -> mabilis(5) = mabilis(7)

Since the lower value for each match will be used, the word 'tumatakbo' will be using the value of 5 and 'mabilis' will also be using the value of 5, for a total of 10.

$$R(\text{'street'}, \text{'kalsada'}) = \frac{|\ C(\text{'street'}) \cap C(\text{'kalsada'})\ |}{|\ C(\text{'street'})\ | + |\ C(\text{'kalsada'})\ | - |\ C(\text{'street'}) \cap C(\text{'kalsada'})\ |}$$

$$= \frac{10}{(13 + 17 - 10)}$$

$$= 10 / 20 = 0.5$$

2.4. Selection of highly similar/correlated

The score computed from the previous process will range from 0.0 to 1.0. The AEFLex system currently has a default threshold of 0.4. It is determined through various testing of the two corpora (The Alchemist[2] and Jose Rizal[8]) that there is a small chance for incorrect translation candidates to obtain a score below 0.4. However, this is under the assumption that all function words and named entities have been properly identified and eliminated. There is also the chance for the scores of correct translation candidates to get scores lower than 0.4 during a two-way (English to Filipino and Filipino to English) extraction. In a two-way extraction, the average of the two scores of each translation candidate is computed. Knowing that there are more unknown and less known words for the Filipino corpus than in the English, the scores from extracting Filipino to English will be lesser than the score from extracting English to Filipino.

2.5. Lexicon Editor

The lexicon editor is a component that will allow the user to add, edit or delete entries from the lexicon. The editor will enable the user to add attributes to a word in the lexicon. Some examples of the attributes that the user can add are gender and phonology. The user can also note the variants in the spelling of a word.

3. Issues

Issues and difficulties encountered from testing the system are discussed in this section. Plans for future testing are also tackled in this section.

3.1. Noisy corpus

There have been instances where a known word appears as unknown. This is due to the abnormality in the corpus that is not clearly seen. An example is the word “the” which is a known function word; however, there were instances that it appeared as “the□” in the corpus. It had an extra character ‘□’, which is also equivalent to ‘\n’, which was not concealed. Due to this reason, some words with noise such as these appear as unknown words, and some of them even have high occurrence count.

3.2. Known word to unknown word ratio

It is determined, through testing, that many of the unknown words are actually known words that have a different morphology. In order to test this, the English stemmer was added in hopes of increasing accuracy and scores. The accuracy and scores increased but the co-occurrence factor remained the same. Since the English stemmer improved the results, a Filipino stemmer was added.

Another issue is with regards to the initial lexicon, which is too small that there are usually more unknown words than known words in an input corpus. This cause the words extracted to have incorrect or low scores.

3.3. Part-of-Speech Tagger

Available POS taggers for English have an accuracy of 93%, while the available Tagalog (not Filipino) tagger has an estimated maximum accuracy of 80 to 90%. Inaccuracies of the POS tagger could also possibly decrease accuracy. One instance would be if a word is matched with a wrong candidate translation due to its tag.

Having a database look-up as the POS tagger does not decrease processing time; but instead causes the processing to take longer. Not only does it increase the total time of extraction, it doesn’t increase accuracy. This is because the words tagged will always be the known words from the database, while the untagged will always be the unknown words, not found in the database.

3.4. Multiword term

The system cannot handle multiword terms because the system separates the input corpora into single word terms. If the input text is already tagged, then the system can support or recognize multiword terms if the words were enclosed in curly braces.

3.5. Assimilated English in the Filipino Language

The Filipino language is dynamic. There are several English words that have been assimilated to be used as a normal Filipino word. Some examples of these are “bus”, “truck” or “colgate”. Assimilated terms are used with Filipino or Tagalog prefixes, suffixes and infixes. Having these words in the corpora might provide additional difficulty in extracting terms.

4. Testing

The system was tested on three different corpora. The testing for The Alchemist[2] and Jose Rizal’s Life Works[8] were tested with parallel and non-parallel comparable. The news articles were only tested as non-parallel comparable corpora. The Lexicon initially contains 22,697 words.

- **Test #** – the *i*th testing of the indicated corpora.
- **Tweak** – The change or different factor from each *i*th test
- **WS** – The window size which contains the number of collocates taken for a word
- **Thres** – The threshold of the test run. The score greater than or equal to the threshold is considered as a candidate translation.
- **Stm** – 0 indicates no stemmer, 1 English stemmer only, 2 both English and Filipino stemmer.
- **TC** – Total Correct Number of English to Filipino translation candidates extracted
- **TN** – Total Number of English to Filipino translation candidates extracted.
- **Acc** – Accuracy of the test run. ((TC/TN)*100)
- **IAcc**– Expected accuracy if all translation candidates are evaluated, even if it did not obtain the highest score for each set of candidate word translation.
- **Lex #** - The total number of words and its translations in the lexicon.

Table 1. Rizal Corpus parallel test results

Test #	Tweak	Lex #	WS	Thres	Stm	TC	TN	Acc
1	Addition of English stemmer to increase scores and accuracy	22,697	2	Highest value	1	2	15	**13%**
2	All of the extracted words are found in the lexicon.	22,699	2	Highest value	1	0	15	0%
3	Removed 2 words from the lexicon	22,697	2	Highest value	1	3	14	**21.42%**
4	Removed 3 more words from the lexicon	22,694	2	Highest value	1	6	14	**42.85%**
5	Removed all the other words without translations from the lexicon	22,685	2	Highest value	1	8	14	**57.14%**

4.1. Jose Rizal Corpus

The Jose Rizal's Life Works[8] corpus was tested mostly on parallel corpora. However, if the corpus was chopped like The Alchemist[2], then the corpora can no longer be considered as comparable because each chapter of the book does not necessarily talk about the same topic. Please refer to table 1 for the test results.

Generally, the test was run with a window size of 2 and only the English stemmer was used for stemming. The threshold for the test is based on getting the highest scoring word among all of the candidate translations of a word. The 1st test was used to see the accuracy of the results if there was an English stemmer. In the next test (test # 2), all of the extracted words together with its correct translation from the 1st test were added as new words to the lexicon. The system was unable to extract any other new known words. Since all of the words and their correct translations extracted from the first test are already in the lexicon, the system can no longer extract words because all of these words are already considered to be found or existing in the lexicon. In test # 3, the system will attempt to retrieve words that were removed from the lexicon. By removing 2 words from the lexicon, the test (test # 3) retrieved 3 words from the corpus. In the 4th test, another 3 words were removed from the lexicon. These words were also retrieved. For the final test, all of the 14 found words were removed from the lexicon. The system was able to extract a total of 8 words out of 14. The accuracy of the extraction reached 57.14%. If most of the words in the input corpora are already found in the lexicon, then the system can extract more words.

The Rizal Corpus was also tested as a non-parallel corpus. For the English part, the chapters 1-11 were used the Filipino part was composed of chapters 12-22. It was tested with a window size of 4 because there were no correct translations for lower window sizes. The known to unknown word ratio of the corpus is not high but it was able to extract 3 correct translations out of the 90 words extracted. Its accuracy is 3%.

4.2. The Alchemist

The Alchemist[2] had an English version and a translated Filipino version. This corpus was tested in both parallel and non-parallel comparable versions. The non-parallel comparable versions were obtained by chopping the English and Filipino version into 2 parts. The English part begins with the prologue and was cut from the book's 2nd part on line number 209 and the Filipino part begins with line number 210 in the 2nd part until the end. Since the electronic version does not indicate chapters, the division of the English and Filipino corpus is based only on an estimate of the corpus. These corpora were run as non-parallel texts. For this section, the results will focus on test results for the non-parallel corpus. Please refer to Table 2 for the results.

The test runs were run under different conditions to see the differences in the improvements done to the system. The first test was to see the effect of an English stemmer to the results. Due to the stemmer, the system was able to extract 3 new words from this test run. The 2nd test run implements the removal of words with apostrophe (') As observed from testing, words with apostrophe, with the exception of ('s) are normally named entities or function words. Therefore by removing the words with apostrophe, the accuracy increased. It also made the scores of the words increase. The 3rd test was run with a different window size. By increasing the window size to 3, the system will check for the 3 words before and 3 words after a word. This will increase the number of co-occurring words and also increase the number of words extracted. However, the runtime of the system increased by 3 times. The accuracy also improved for this test run. The 4th run included a Filipino stemmer. This test run increased the results in the expected accuracy but the actual accuracy only reached 7.9%.

Table 2. The Alchemist parallel test results

Test #	Remark	WS	Thres	Stm	TC	TN	Acc	IAcc
1	Addition of English stemmer to increase scores and accuracy	2	0	1	11	135	**8.1%**	16%
2	Removal of all words with apostrophe on the assumption that most of them are function words	2	0	1	12	137	**8.8%**	13.9%
3	Runtime increased by a factor of 3.	3	0	1	22	215	**10.2%**	14.4%
4	Addition of Filipino stemmer to increase scores and accuracy	2	0	2	10	126	**7.9%**	19.7%
5	Addition of three high frequency Filipino words as function words. To increase chances of correct Filipino candidate translations to be selected	2	0.4	2	14	105	**13.3%**	
6	Removed 9 words from the lexicon English and Filipino Stemmer is used.	2	0.4	2	18	95	**18.2%**	
7	Look-up POS tagger is disabled and proven to be ineffective and increases time needed to complete.	2	0.4	2	18	95	**18.2%**	
8	English and Filipino Stemmer is disabled in order to prove its effect to the results	2	0.3	0	18	104	**17.3%**	

9	Look-up POS tagger and Stemmer is disabled. No difference to Test Number 8 except less time required to complete.	2	0.3	0	18	104	**17.3%**	

However, by refining the next test, the accuracy can increase. The 5th test was run to see if the results will improve by adding some high frequency words which appear as the highest scoring translation for each source word to the list of function words. The accuracy improved to 13.3%. To see the effect of the English and Filipino stemmer, 9 words were removed from the lexicon. All these 9 words were extracted and the accuracy improved for the 6th run compared to the previous test runs. The 7th test is similar to the previous (6th run) test. The lookup POS tagger is ineffective for the system because it increases the runtime if it is used but it does not improve scores or the accuracy of the test. For the 8th and 9th test, the system was run without the stemmer. The 9th run was run similar to the 8th but the POS tagger was disabled. The stemmer improves the scores of each word extracted. When the stemmer was disabled in the test run, the scores for several words no longer reached the original threshold (threshold is 0.4 by default). The threshold needed to be set to 0.3 to extract correct results.

The Alchemist was also tested as a non-parallel corpus. However, it did not get good results. It only got less than 1% accuracy. This result was caused by the Filipino corpus having a very low known word to unknown word ratio. (2:5)

4.3. News Articles

The system was also tested on news articles collected from The Malaya and the Abante website. The news for the English corpus were 2 news articles that dated from April 1, 2005 while 16 Filipino news articles were used to make the size of both corpora similar. There were no correct translations found. The corpora had very low (1:3) known word to unknown word ratio making it difficult to find collocates for a word.

5. Conclusions and Recommendations

The system can extract unknown words from corpora; however, there are still several factors that decrease accuracy or produce incorrect results. Below are some possible solutions that can be tried to improve the results and the accuracy of the system.

First, having efficient morphological analyzers can greatly increase the accuracy of the system. By determining the root words accurately, there will be fewer words and less candidate translations to compare with. At the same time, it will also increase the count of the words linking the unknown word and the candidate translation. These two events can increase the scores for each candidate translation, as well as provide fewer candidates for each word.

The system could also improve by having an increased list of function words and a better named entity recognition component. By successfully removing insignificant words from the system, only the relevant words will remain and the number of candidate translations and words linking the unknown word and the candidate translation will lessen. The system may then be able to achieve better results.

If the known to unknown word ratio of a corpus is high, then there will be more words linking an unknown word to its candidate translation, thereby significantly improving the results. The known to unknown word ratio can be increased either by having an initial lexicon with more words or by using corpora with a lot of words found in the lexicon.

By tweaking the system and the lexicon, the AEFLex system can extract correct translations from English and Filipino corpora. The system has several components that enable it to extract words. It contains a lookup POS tagger, a stemmer and a lexicon editor. The accuracy of the system can reach at most 57%.

6. References

[1] Borra, A. et al. (1997). IsaWika!. Philippines: University of the Philippines.

[2] Coelho, P. (1988) The Alchemist. United States.

[3] Cornell University. Ithaca, New York, USA. [online]. Available: ftp://ftp.cs.cornell.edu/pub/smart/english.stop

[4] Kaji, H. et al. (1996). Extracting Word Correspondences from Bilingual Corpora Based on Word Co-occurrence Information. Tokyo, Japan.

[5] Panginiban, J. (1946). English-Tagalog Vocabulary.

[6] Porter's Stemming Algorithm. [online]. Available: http://www.tartarus.org/~martin/PorterStemmer/

[7] Sadat, F. et al. (2003). Learning Bilingual Translations from Comparable Corpora to Cross-Language Information Retrieval: Hybrid Statistics-based and Linguistics-based Approach. Nara, Japan: Nara Institute of Science and Technology.

[8] Zaide, G. (1999) Jose Rizal: Life, Works and Writings of a Genius, Writer, Scientist and National Hero. Quezon City: All-Nations Publishing Co.