AUTOLEX: An Automatic Lexicon Builder for Minority Languages Using an Open Corpus

Evan Liz C. Buhay[a], Marie Joy P. Evardone[a], Hansel B. Nocon[a],
Davis Muhajereen D. Dimalen[a], Rachel Edita O. Roxas[b]

[a]Information Technology Department, School of Computer Studies
MSU Iligan Institute of Technology, Tibanga, Iligan City, Philippines 9200
ebuhay, mevardone, hnocon@yahoo.com, davis.dimalen@gmail.com
[b]College of Computer Studies, De La Salle University-Manila
rachel.roxas@delasalle.ph

Abstract: The aim of this study is to build natural language resources for languages with limited resources or minority languages. Manually building these resources is tedious and costly. These natural language resources such as a language corpora and lexicon will be used for natural language processing research and system development. Tagalog, a minority language was considered in this study as a test bed. This study exploited the use of the WWW to retrieve documents that are written in a minority language. We employed a frequency-based algorithm to build the lexicon. For our evaluation, we considered 260 Tagalog documents extracted from the web as our corpus. From the corpus, the system automatically selected 1,386 candidate unique words based on the threshold (with value of 10) as the lexical entries. Each lexical entry is validated by a language expert. Our evaluation shows an accuracy of 97.84% and only 2.16% error rate. The error was based on incorrectly spelled words or words that are not Tagalog.

Keywords: Automatic lexicon builder.

1 Introduction

Philippine natural language (NL) resources such as document corpora and lexicons are badly needed in natural language processing research and system development. Building these NL resources requires time and linguistic knowledge. These resources can be built automatically or manually. The manual process of building NL resources such as a lexicon is tedious and is prone to typographical errors and if copied one by one from an existing lexicon may create copyright issues. Thus, this study implemented an automated system that builds a language specific lexicon by using an open corpus or the World Wide Web.

In our study, lexical enries are determined based on the frequency of the words in the corpus. Words with higher frequency were selected and words that are below a certain threshold were not selected. This kind of lexicon can be used in variety of human language technolog systems, such as word database, word processesors, software for read back by speech synthesis in Text-to-Speech systems and dictation by automatic speech recognition systems (Al-shalabi and Kanaan, 2004).

Our approach in retrieving documents from the web was based on our previous study on a resource builder on a closed corpus (Dimalen, 2004). The resource builder is ideal since it is a system that automatically builds a minority language corpus. We extended the resource builder to retrieve documents in an open corpus by using google API which has access to the google search engine database.

Both the corpus and the lexicon that were automatically built by our system are natural language resources that are significant to Lexicology. Lexicology is the branch of descriptive linguistics concerned with the linguistic theory and the methodology for describing lexical information, often focusing specifically on issues of meaning (Al-shalabi and Kanaan, 2004).

Lexical information includes lexical semantics, and the study of the syntactic and morphological and phonological properties of words.

2 Automatic Lexicon Builders

This section will describe existing systems similar to our study. Table 1 shows a list of automatic lexicon builders including our system with features common to all. Features are listed to be able to compare each of the entries.

Table 1: Lexicon Builders

	Appended Linguistic Attributes	Automatic	Lexicon Selection Basis	Input or Resource	Input type
Arabic Automatic Lexicon Builder	morphological info, POS, etc.	yes	Pattern recognition	Text document	Domain and Language specific
Lexicon Extraction from a comparable corpora	POS, etc.	yes	Sense grouping	Parallel corpora (2 language specific corpus)	POS tagged Language and Domain specific
AutoLex	none	yes	Frequency based	Open or Closed Corpus	Domain and Language specific

2.1. Constructing an Automatic Lexicon for Arabic Language

Figure 1 shows the architecture of the research for constructing an automatic lexicon for Arabic language (Al-Shalabi and Kanaan, 2005). The objective of their research is to build a Lexicon for the Arabic language by automatic means. Unlike our research their study builds a lexicon that contains morphological information, part-of-speech tags, linguistic attributes, patterns and affixes for all lexicon entries.

The system is devided into 6 main stages namely tokenization, stemming, affix extraction, pattern recognition, part-of-speech tagging, and finally storing words in the lexicon.

The system starts by entering a vowelized or non-vowelized Arabic text document taken from the Holy Qur'an and from Arabic abstracts taken from the Proceedings of the Saudi Arabian National Computer Conference. After the document is fed to the system, the input document will undergo the 6 stages mentioned above and will output a lexicon for the Arabic Language.

According to the result of their study they achieved a 96% accuracy.

2.2. Lexicon Extraction from Comparable Corpora

Another lexicon builder shown in figure 2 automatically builds a lexicon from comparable corpora. It accepts as input a source and a target comparable corpora to build a lexicon. Both the source and the target corpora were assumed to be POS tagged. Word pairs were gathered from both corpora and aligned with the help of a bilingual lexicon. Correlation of the senses

that accompany the is done as a final process. A certain threshold was set for measuring the similarity of the senses (Tiu, 2003).

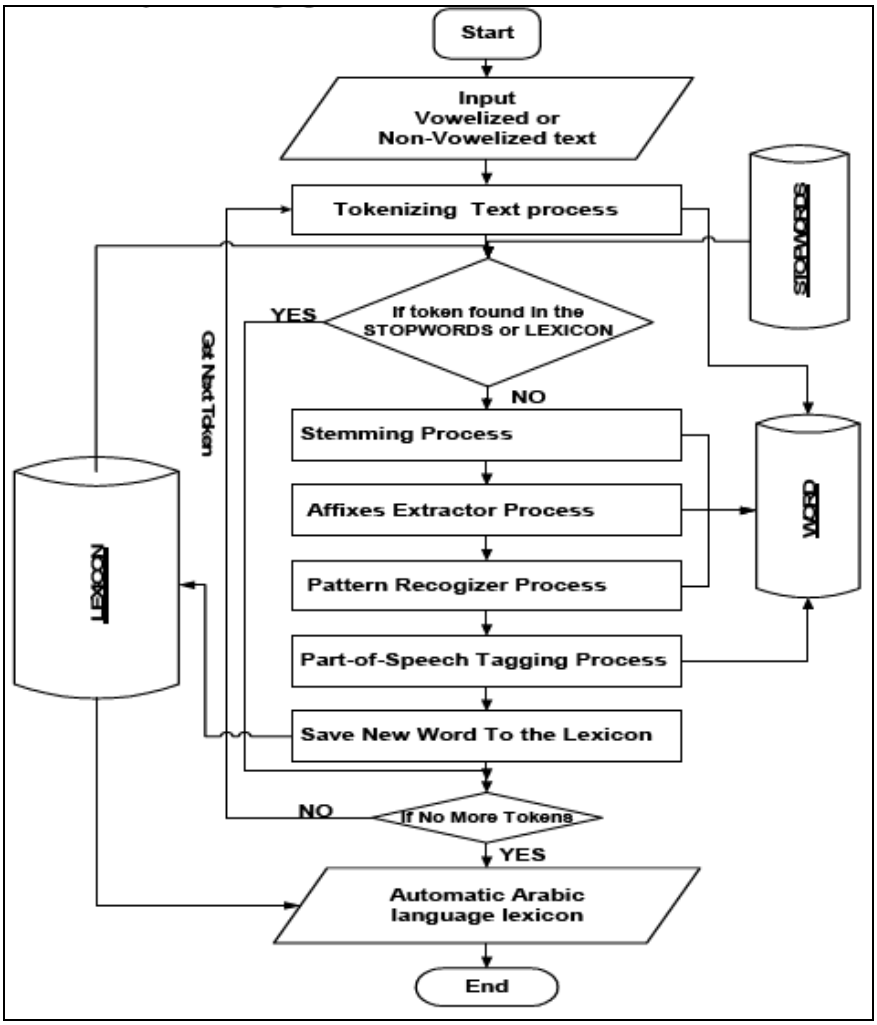

Figure 1 Automatic Lexicon Builder for Arabic Language

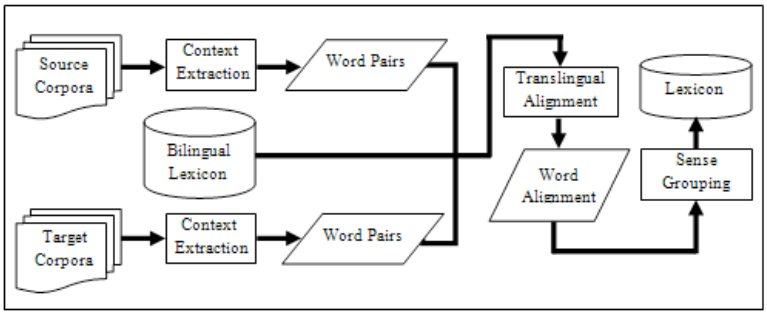

Figure 2 Lexicon Extraction From Comparable Corpora

3 Approach

This section presents the algorithm and steps in the development of the Autolex. Figure 3 below summarizes all the salient steps in the lexicon.

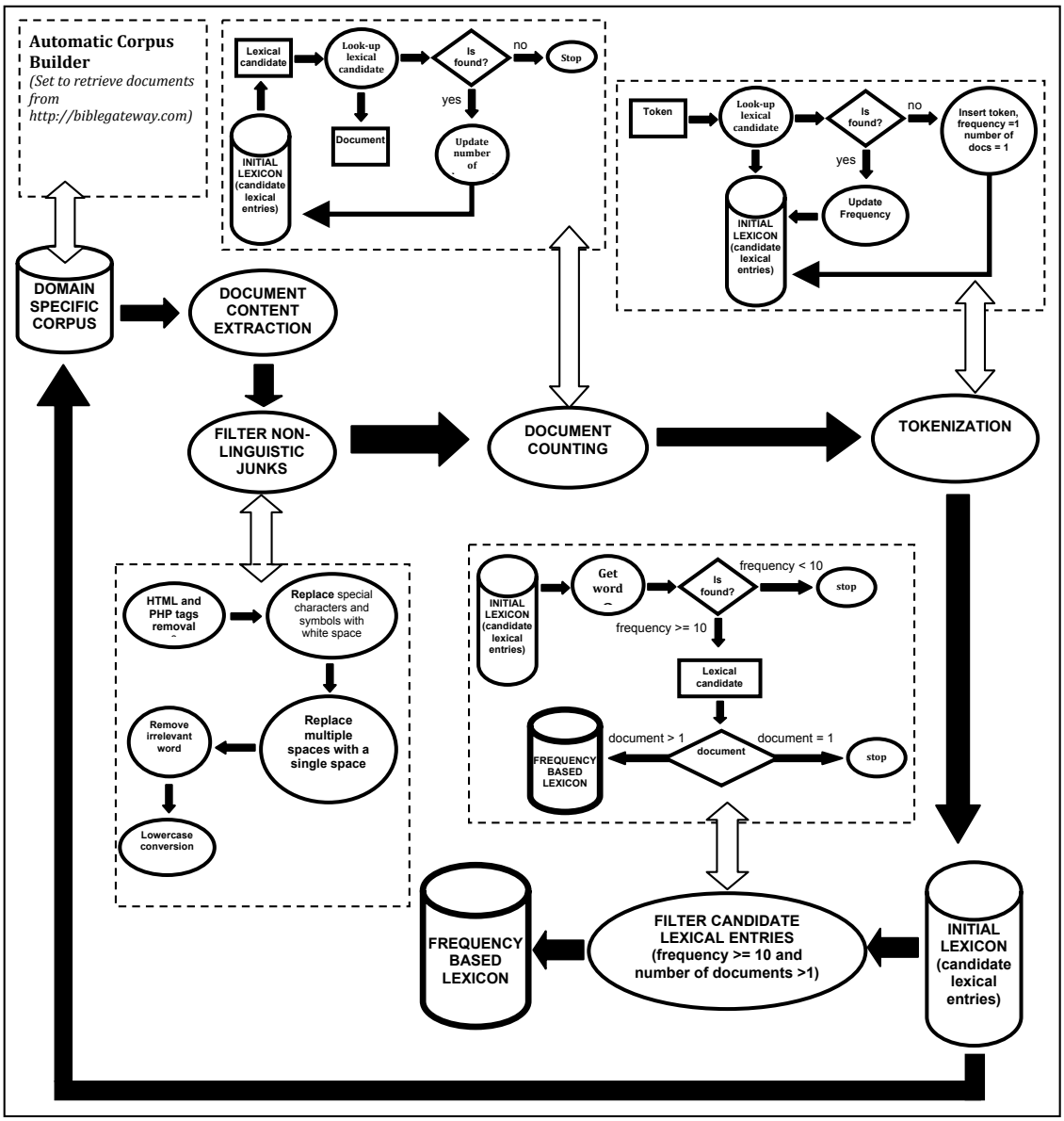

Figure 3: Autolex Architectural Diagram.

3.1. Corpus Builder

The corpus builder retrieves document from the internet via google API enabling it to access the google search engine database. The process of building the corpus includes query generation, document sampling via google API and document classifier. The query generation automatically gets a set of positive query terms and negative query terms from relevant document and non relevant document respectively. The query terms are fed to the sampling module wherein the terms will be processed by the google API. The google API will return search results wherein each of the URL will be extracted by the document extraction module. The extracted document will be process by the document classifier module by identifying language from which the document is written. The set of documents retrived by the corpus builder will be used by the next module.

The domain specific corpus contains 260 Tagalog documents. These documents were gathered automatically using the Corpus Builder. By setting the specific web domain to "http://biblegateway.com", AutoCor was able to retrieve domain specific documents. In this

case, “documents from the bible” is the domain being considered. AutoCor only retrieves from the site, documents written in Tagalog only.

3.2. Filtering non-linguistic junks

Non-linguistic junks are those characters and symbols that are unnecessary in the tokenization process (Sokolava, 2004). These junks can be html tags and other characters and symbols. The removal of these junks depends on the nature of the task to be performed (See Figure 4). Since the words are the only ones needed in this research, html and php tags are removed. Special characters and symbols are replaced with a white space. Tables 2 and 3 shows the list of the tags, symbols and characters removed. Extra spaces like tabs and new line are then converted to single space. Irrelevant words such as English words that are common to all of the documents in the corpora are removed. After all the non-linguistic junks are removed, all letters are converted to lowercase.

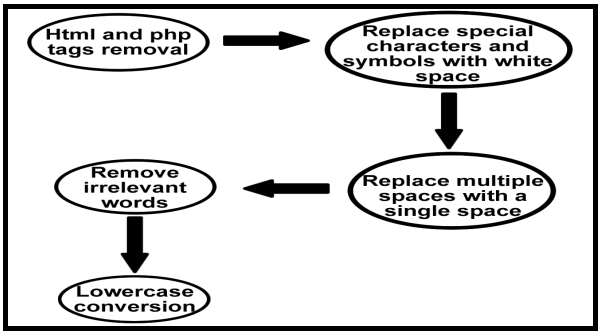

Figure 4: Filtering non-linguistic junk.

Table 2: List of tags removed.

Javascript	<script[^>]*?>.*?</script>
Style tags	<style[^>]*?>.*?</style>
Html tags	<[\/\!]*?[^<>]*?>
Multi-line comments	<![\s\S]*?—[\t\n\r]*>
Others	, &amp, &lt, &gt, &quot, &lsquo, &rsquo &bdquo, &copy, &permil,&raquo, &trade, &laquo &Aacute, &aacute&Oacute, &oacute, &reg

Table 3: List of special characters and symbols removed.

Numerical characters	1 2 3 4 5 6 7 8 9 0
Panctuation marks	, . ! ? ; : “ ‘
Special characters and symbols	( ) [ ] { } / \\ | @ # $ % ^ & ~
Mathematical symbols	+ - = \ *
Others	white space + - + white space

3.3. Document Counting

Knowing the number of documents wherein the word is found is important to avoid biases. There are words that appeared several times in a document making its frequency higher than the other words, thus to avoid biases in lexical entries selecting the number of documents wherein the word appears in is considered. Counting is done by searching in a document, a certain lexical candidate found in the initial lexicon, if the word is found the number of document is updated. Otherwise proceed to another lexical entry and vice versa. Figure 5 shows the document counting process.

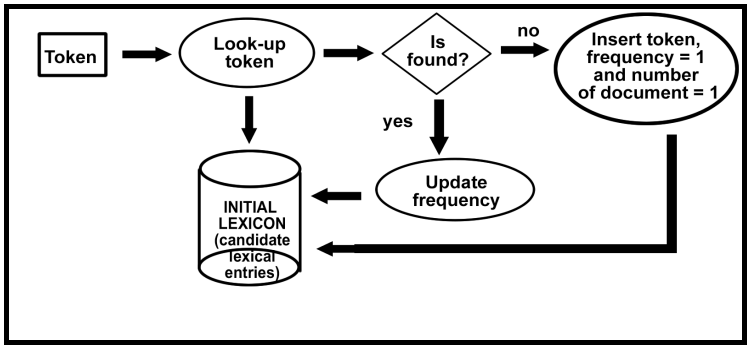

Figure 5: Document Counting Process.

3.4. Tokenization

The texts are tokenized by words. Space is the common separator of words. Thus, space marks are used as the explicit delimiters or token separator.

Every time a space is encountered, the words after the space become a token. The token is stored in an initial lexicon in the database setting its frequency initially to 1. Every time the same token is seen its frequency is updated. Otherwise if the token is not yet found in the initial lexicon, it is inserted to it. This process repeats until the last document is finished being tokenized. Figure 6 shows the diagram representation of the tokenization process (Zdziarski, 2005).

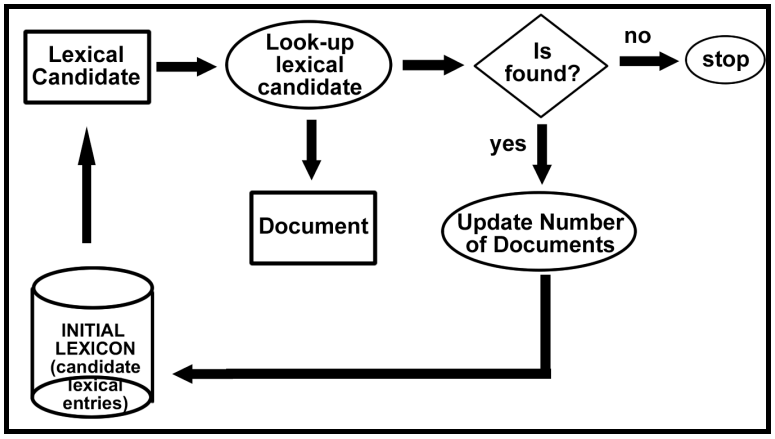

Figure 6: Tokenization Process.

3.5. Lexical Entries Selection

Lexical entries are selected based on their frequency and the number of documents it appeared. The threshold set is that words with frequency greater than or equal to 10 and appears in more than 1 document are chosen as the lexical entries. The threshold set is based on the research studies of Cynthia Whissell on her Whissell's Dictionary of Affect in Language and of Julio Gonzalo et.al on their study of Corpus-based Terminology Extraction Applied to Information Access.

The lexical entries are sorted based on their frequency. Words having the same frequency were arranged alphabetically. Sorting was done in PostgreSQL using SQL statements (See Figure 7).

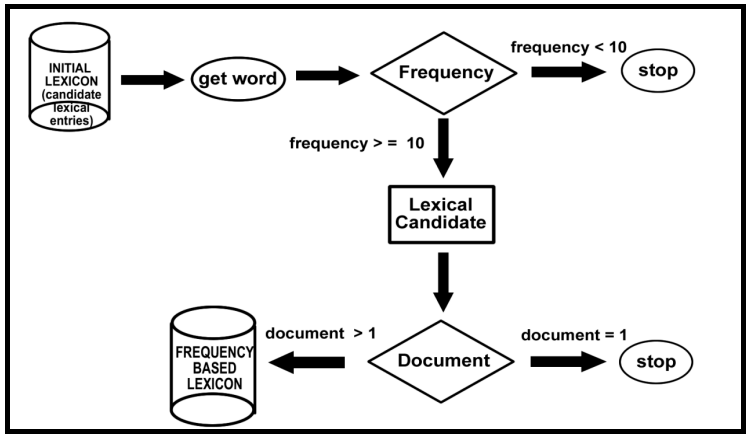

Figure 7: Lexical Entries Selection.

4 Evaluation

For evaluation purposes, a domain specific corpus containing 260 documents consisting of 201,076 tokens was considered. The corpus was processed such that only 9,965 unique words were left out of the total number of tokens.

The filtering criteria for the selection of words in the AutoLex was based on the research studies of Cynthia Whissell on her Whissell's Dictionary of Affect in Language and of Julio Gonzalo et.al on their study of Corpus-based Terminology Extraction Applied to Information Access. It adapts words with frequency greater than or equal to 10 and appearing in more than one document as the minimum requirement for filtering the lexical candidates.

In view of these criteria, only 1, 386 words out of 9,965 or 13.91% of the lexical candidates were chosen as the final lexical entries, 45 of which are proper nouns. Each of these entries was then evaluated manually by the language expert. A term was classified valid or invalid whether the word is correctly spelled and if it is found in the Tagalog Language.

Table 4: Evaluation Distribution of the Final Lexical Entries.

Word Frequency Interval	**Number of Lexical Entries (A + B)**		**No. of Valid Lexical Entries (A)**		**No. of Invalid Lexical Entries (B)**	
10-59	1053	75.97%	1027	97.53%	26	2.47%
60-109	134	9.67%	131	97.76%	3	2.24%
110-159	53	3.82%	52	98.11%	1	1.89%
160-up	146	10.53%	146	100%	0	0.00%
Total	1386	100%	1356	97.84%	30	2.16%

Table 4 shows the evaluation distribution of the lexicon. As shown in the said table that majority of the lexical entries resulted to about 75.97% have word frequency 10 to 59 and the least, with 3.82% of the entries have word frequency 110 to 159.

Among the 1053 entries in the first interval, 97.53% are valid while 2.47% are invalid, based on the experts evaluation. The percentage of invalid entries decreases as the word frequency interval increases, as shown in the same table. In the second category, percentage correct entries is 97.76% with an error of 2.24%, while for the third interval, percentage correct is 98.11% with an error of 1.89%. For the last interval, percentage correct is 100%. As expected more invalid terms are found in the interval with the least number of word frequencies.

A graphical representation of the percentage error is shown in Figure 8. The graph shows that as the word frequency increases the percentage of error decreases. This implies that an entry is highly reliable as its frequency of occurrence gets 160 or higher or as the entry appears

most often in documents the higher is its validity or correctness. By these, one can say that high frequency words have a greater possibility that it is correctly spelled.

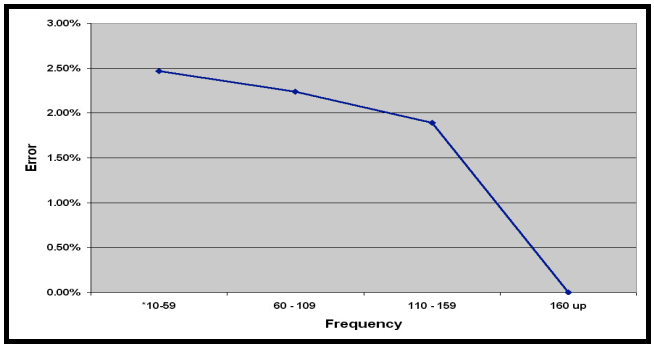

Figure 8: Percentage Error

4.1. Error Analysis

Table 5 shows a sample of the words that are considered invalid. Though the words reached the threshold set, they are still considered invalid since they are not correctly spelled and are not suitable Tagalog words as evaluated by the language expert. Like the word *punong-kahoy*, it is considered invalid because according to the language expert that should be spelled as *punongkahoy*.

Table 5: Sample of Invalid Lexical Entries.

Lexical Entries	Frequency	Number of Documents	Mark
Festo	14	3	invalid
Jesusnang	12	10	invalid
naala-ala	10	9	invalid
punong-kahoy	30	17	invalid
Tiro	11	8	invalid
Apollos	10	7	invalid
Niluwalhati	14	14	invalid

One factor that causes the inclusion of these erroneous words in the lexical entries is that, the source corpus was written by the same person. There is a tendency that a misspelled word is constantly used by that person. Another is due to typographical errors.

4.2. Overall Evaluation

Based on the result of the evaluation, the system is reliable since it was able to build a lexicon with 97.84% accuracy. Also, the words are validated by a language expert, hence, one is assured that the words marked as invalid are really unacceptable and words that are marked as valid are correct.

5 Conclusion and Recommendation

AutoLex is an automatic frequency-based Tagalog lexicon for spelling checkers. It does not provide a dictionary with meanings but merely a word list, gathered on the basis of their frequency. This research study can be a starting point for building dictionary in the future which is also a prerequisite in building natural language applications such as machine translators and many others.

The system was developed using an open source programming language and is not language specific, so that the system can be used to make a lexicon for other languages, not only Tagalog. One will just have to use language specific corpora to be able to acquire the desired word list for that language.

The system used a domain-specific corpus gathered using AutoCor, as the source of the lexical entries. The lexicon acquired 1,386 lexical entries, 45 of which are proper nouns. The system was able to build this in an approximate time of 4 hours unlike building it manually which will take a longer period of time.

Lexical entries were evaluated by a language expert. Based on the evaluation, the developed lexicon has an accuracy of 97.84% and only 2.16% error, which is considered highly reliable. The 2.16% error was caused by misspelled words. Being a benchmark study on NLP, the developed lexicon is useful for spellcheckers in Tagalog and for other users of Tagalog language and its application.

Consequently, AutoLex is capable of automatically building a lexicon for spelling checkers.

For interested researchers, this study offers a lot of avenues for further study. One may explore other criteria in the selection or filtering process. Or one may adopt some other procedures, say statistical methods, in the evaluation process. It is also recommended that the study be tested using non-domain specific corpus and a larger corpus to obtain a large list of Tagalog words. Finally, this study can be adopted in creating lexicon for other languages or for other areas of discipline.

6 References

Dimalen, D. (2004). AutoCor: Automatic Acquisition of Corpora of Closely-Related Languages. Masters Thesis, DLSU, September 2004.

Gonzalo, J., Peñas, A., Verdejo, F. (2001). Corpus-based Terminology Extraction Applied to Information Access. http://interneg.concordia.ca/interneg/research/papers/2004/01.pdf

Kuenning, G. (1996). Dictionaries for International Ispell. http://lasr.cs.ucla.edu.geoff/ispell.html

Monson, C., Levin, L., Vega, R., Brown, R., Llitjos, A. F., Lavie, A., Carbonell, J., Cañulef, E., Huisca, R., Data Collection and Analysis of Mapudungun Morphology for Spell Checkers. http://www.cs.cmu.edu/~alavie/papers/Mapudungun-LREC-04.pdf

Santos, Vito C. (1986). Pilipino-English Dictionary. Philippine Graphic Arts, Inc. 163 Tandang Sora, Samson Road, Caloocan City

Sokolava, M. (2004). Automatically Building a Lexicon from a Noisy Data in Closed Corpus. http://interneg.concordia.ca/views/bodyfiles/enegotiation/projects/users/03report1.html

Whissell, C. Whissell's Dictionary Affect in Language Technical Manual and User's Guide. http://www.hdcus.com/manuals/wdalman.pdf

Zdziarski, J. (2005). Tokenization the Building Blocks of SPAM. http://nostarch.com/download/endingspam_ch6.pdf