In this paper, we present what we have attempted towards the digitalization of the Philippine languages and their respective applications, and what we intend to do in the future. We present the development of a multi-engine bi-directional English-Filipino Machine Translation (MT) system, and the building of various language resources and tools for this system. We also discuss our experiments on various technologies for automatic extraction of these resources to handle the intricacies of the Filipino language, designed with the intention of using them for the MT system. We explore here what will be our possible future directions in terms of connecting the Philippine islands through the digitalization of our Languages.

Key Words: language resources; language tools; lexicon; machine translation; morphology.

1. INTRODUCTION

In [16], it presented a vision towards the development of a multi-engine bi-directional machine translation (MT) system that is a computer software that performs machine (or automatic) translation of English texts to Philippine languages, and vice versa. For the past three years, the DLSU NLP research team is coming to the end of the first stage of providing a hybrid English-Filipino MT system which basically involves one Philippine language, Tagalog, as our Filipino language basis. We discuss in this paper what we have already done towards the fulfillment of this vision, and what can we do as NLP researchers and citizens of this nation to connect Luzon, Visayas and Mindanao through the digitalization of our languages.

2. THE MACHINE TRANSLATION ENGINES

Natural language translation is a very complex task. Its automation presents more issues and difficulties in addressing, in particular, translation quality. During the past 60 years of research on machine translation involving various languages, different paradigms and methods have been suggested and employed to improve translation quality. The issue is: how effective is an approach in capturing the features of natural languages and the translation phenomena between these two languages so that translation quality can be considered at par with the translation of human experts? It has been shown in previous researches that this can not be achieved by a single MT paradigm. Hence, the multi-engine MT research attempts to integrate the various MT paradigms so as to draw a synergy out of the strengths of these paradigms. The current MT system outputs the MT engines’ respective generated translations and presents them to the user.

Our multi-engine English-Filipino MT system considers both the rule-based and corpus-based approaches. Rule-based MT builds a database of rules for language representation and translation from linguists and other experts; while corpus-based MT automatically learns such information from sample text translations. Two corpus-based approaches are considered: example-based and template-based. An expansion slot allows for the integration of other MT engines into the system in the future. To implement these paradigms, language resources are built, such as the bilingual English-Filipino lexicon (or electronic dictionary), Filipino grammar, translation rules and annotated corpora. The corpus-based approaches require a part-of-speech tagged corpus. The MT paradigms will also use the morphological analyzer and generator to extract root words from texts. The next phase of the project is the language modeler that would choose the “best” translation, either as-is or integrating the outputs of the various engines.

One of the major challenges is in the building of language resources such as the lexical and morphological information, lexicon, grammar, corpora and translation rules from almost non-existent digital forms. Linguistic information on Philippine languages are available, but as of yet, the focus has been on theoretical linguistics and little is done on the computational aspects of these languages. Computational issues involving Philippine languages, in particular Filipino, are considered in this study. Some of these issues include the complex verbal morphology of the language, free word order of sentences, and focus of the sentence.

3. COMPUTATION LEVELS OF LANGUAGE REPRESENTATION

3.1. Morphological Processes

We consider rule-based and example-based approaches for both morphological analysis and generation. As with the MT systems, the rule-based approach requires the capturing of the morphological phenomena through the expertise of a human linguist, and transforming them into rules that can be represented in a computer, while the example-based approach automatically extracts morphological behavior from examples fed into the system. Since the Filipino language has very complex
word morphology, the rule-based approach can be complemented by the example-based approach.

Tagalog language exhibits complex morphological phenomena, which include concatenative behavior such as prefixation and suffixation, and non-concatenative behavior such as infixation and reduplication (either partial or full). Tagalog is also an agglutinative language, where for instance, prefixes may be successively combined. Research on computational morphology has been predominantly on concatenative morphology and on finite-state models of morphotactics [12]. Although attempts were made to handle non-concatenative phenomena, it has been on a limited capacity only [2; 3].

We experimented on a rule-based MA using Optimality Theory (OT) [9], since OT has been proven effective in generating non-concatenative phonology. Test results show a 96% accuracy when on 1,600 Tagalog verb forms (having 3 to 7 syllables) from 50 Tagalog roots which exhibit both concatenative and non-concatenative morphology. The 4% error is attributed to d-r alteration, an example of which is in the word *lakab* and suffix *an*, but *d* is changed to *r*. Unfortunately, time efficiency is affected by the exhaustive search performed since all candidates are generated, and erroneous ones are later eliminated through constraints and rules.

To augment the rule-based MA, we also experimented on an example-based MA by extending the WordFrame model [18] from a seven-way split representation of morphological re-write rules from word pairs of a morphed word and its corresponding root. Unfortunately, it is still not capable of fully modeling Filipino morphology since some occurrences of reduplication are still represented as point-of-suffixation for various locations of the longest common substring. There are also some problems in handling the occurrence of several partial or whole-word reduplications within a word. Despite these problems, the training of the algorithm that learns these re-write rules from 40,276 Filipino word pairs derived 90% accuracy when applied to an MA. The complexity of creating a better model would be costly but it would ensure an increase in performance and reduced number of rules [5].

### 3.2. The Filipino Grammar

The Filipino grammar was defined manually by formalizing observable conceptual rules and patterns of the language upon consultation with linguists, and specified using the lexical functional grammar (LFG) formalism. The main building block of LFG is the context free grammar which handles the syntax level of the language and the additional semantic actions that capture the semantic categories of words in the sentence such as subject and object.

One of the major challenges of the Filipino language is its free word order in sentence formation, wherein one sentence in English can be translated to various sentences in Filipino. For instance, the English sentence *The man bought an umbrella from the store* can be translated into several different Filipino sentences while maintaining the semantics of the original English sentence:

- *Ang lalaki ay bumili ng payong sa tindahan.*
- *Bumili ang lalaki ng payong sa tindahan.*
- *Bumili ng payong ang lalaki sa tindahan.*
- *Bumili ng payong sa tindahan ang lalaki.*

Because of this free-word order phenomenon in Filipino sentences, grammar representation and translation mapping from source to target language, that is English to Filipino, and vice versa, also pose problems of one-to-many and many-to-one relationships, respectively.

### 3.3. Translation Rules

Transfer (or translation) rules are based on the lexical functional grammar specification of the two languages and are bi-directional. Rules can be specified by hand (rule-based), or can be derived automatically from a set of examples.

For our example-based MT system, transfer rules are learned using the general framework used by [6; 19], which has three steps: seed rule generation, compositionality, and generalization. In seed rule generation, seed rules that define the token sequence, expressed as a combination of POS tags and possibly constant words, token constraints and alignment scheme of a translation pair are generated. Compositionality infers rules of higher syntactic structure, that is, constituent labels are deduced by using the longest adjacent POS tags found in both the English and Filipino rule. The system groups together similar rules and generalizes it to encompass a wider range of unseen examples. In general, we have achieved 68% accuracy of translated sentences when all other information in morphology, lexicon and tagging are correct. The 32% error can be attributed to lack of information in the lexicon for possible disambiguation by the semantic analyzer [1].

Another way to capture trends in the example translations is through templates and chunks, so instead of rules, templates and chunks are saved into the knowledge-base. Sentence templates are used for translation, and when no appropriate templates are available, phrase translation chunks are used for the construction of the translations. We adopted the machine learning techniques to implement the similarity template learning algorithm performed by [6], and introduced template refinement and derivation of templates from previously-learned chunks. Test results show that strict chunk alignment with splitting (SCAS) in training, instead of loose chunk alignment, wherein correspondences are not required, improves quality of learned input. Test results show an accuracy of up to 96% in the testing of a translation corpus. Our results show possible extensions on a more stringent match disambiguation procedure, an algorithm that accepts chunks with empty contents, a feasible chunk refinement method, more comprehensive linguistic resources, or an approach to Cicekli’s difference template learning algorithm [11]. This is used during the template
extraction phase, to ensure that the system does not learn templates whose fixed elements contain only common or noise words, e.g., The X ⇔ Ang X. This also prevents the system from learning long chunks for X, which would be very difficult to use during translation.

4. LANGUAGE RESOURCES

In this particular NL application, we require a lexicon and corpora. We describe here briefly these resources.

4.1. The Lexicon

The lexicon (or dictionary) is a collection of source words with the corresponding translation in the target language, and their features (such as part-of-speech tag, sample sentences, and semantic information). The base lexicon for the Multi-engine MT project is the dictionary of the Commission on the Filipino Language (CFL) of the Philippine government, which contains approximately 10,000 English source words with a total of about 25,000 Filipino meanings. Each English entry in the lexicon contains the POS tag of the word and the corresponding Filipino meaning, with co-occurring word information, and other English features or attributes (e.g., other acceptable spellings or synset id based from WordNet) for proper translation. Semantic features address the problem of ambiguity since many words have many-to-many relationships and determining the proper translation is vital. A study is currently being explored in building a Tagalog WordNet so that the base lexicon can be extended to handle semantic features as in the WordNet.

But since languages are in the process of evolution, it is imperative that the project provides some way to be able to determine and capture new words and probably new meanings of words in the languages considered in this study. New terms can be added into the base lexicon through automatic lexicon extraction from documents on English and Filipino. New lexicon terms are automatically learned from sample documents. Two approaches have been experimented on using parallel documents (translations of each other) and non-parallel documents (non-translations of each other). In using parallel English-Filipino corpora in the automatic lexicon extraction [13], a probabilistic approach is used to determine candidate translations of English and Filipino words through the principle of co-occurring words in parallel English-Filipino documents, since it is assumed that semantically related words usually co-occur (i.e., words that appear together) in documents in a language and would also co-occur in documents in a different language. Only 57% accuracy is derived from training using about 40,000 words, since there are other sources of errors such as tags which were not verified by linguists.

In the representation of assimilated words in the lexicon, there may be various entries of words in the lexicon when following various spelling conventions [7]. For instance, the word congregation can be spelled as konggregasyon or kongregasyon.

4.2. The Corpora

Our work requires both parallel English-Filipino corpora and monolingual corpora. The example-based MT engines require parallel English-Filipino corpora, while components of the system such as the part of speech Filipino tagger only requires Filipino corpora. We have built bilingual parallel English-Filipino corpora which consist of 207,000 words, and currently half of the Filipino documents are manually POS tagged and verified by linguists, and about 4,000 words in the monolingual Filipino corpus.

To address the need of building a reliable Filipino corpora and yet minimizing the need for manual checking, automatic tagging improved the precision of the system by 53% for Bicolano, 18% for Cebuano, and 20% for Tagalog for query lengths 4, 1 and 2, respectively.

We developed the Filipino tagset with 9 general POS tags, 60 specific POS tags and 5 other tags (for punctuation and currency symbols), based on the Penn Treebank tagset which contains 36 POS tags and 12 other tags [14]. We identified tags for nouns, pronouns, determiners, adjectives, conjunctions, verbs, adverbs, cardinal number, and punctuation marks. One of the main differences in the tagset of English is the tag for the word ay which is literally translated to is in English. In the sentence Ako ay kumakain (I am eating), the Tagalog ay is not considered as a verb but as a lexical marker since it only signifies that the sentence is in the form subject + ay + predicate and is not in the natural order. The natural rendition of this sentence is Kumakain ako which is of the form predicate + subject. Also, singular and plural personal pronouns have distinct tags, while gender in Tagalog pronouns such as siya (he or she) is not captured. Interjections are also tagged in Tagalog. The English if which is translated to kung in Tagalog is tagged as a conditional adverb rather than a conjunction.

While our linguists are manually tagging our corpora, we attempted to build automatic POS taggers as well. The example-based tagger requires a large corpus where words are associated with the corresponding part of speech tags, while the rule-based tagger based on Brill’s
tagger [4] also learns tagging rules from examples. Initial tagging can be done through our template-based Tagalog tagger [15] since accuracy of this tagger is only 83% for general tags and 77% for specific tags, and verification of tags are done manually so that tags can be confidently considered correct.

5. PALITO

Our English-Filipino machine translation system has given us a hands-on experience on an NLP application and the necessary linguistic resources for this particular application. We contend and propose that we come up with NL applications that will involve as many Philippine languages as possible and as many NL researchers all over the country, as well as being socially-relevant. We also explore ways in which we can use and re-use existing language resources and tools on Philippine languages that we have already built for the successful implementation of these NL applications.

We envision a system that includes an NL generation component that will output NL in various Philippine languages depending on the choice of the user. It could include a question-answering system and a report generation system. Natural language resources would include a lexicon and grammar knowledge-base, and NL tools would include an MA and an MG, a NL parser and sentence generator.

To be able to successfully undertake this project and other projects of this nature with the purpose of connecting the Philippine islands through language, we need the involvement of researchers from various parts of the country who are willing to work on automation of their respective local languages, and various applications that we intend to develop for these languages.