A Hybrid Agent for Automatically Determining and Extracting the 5Ws of Filipino News Articles

Evan Dennison S. Livelo
evan_dennison_livelo@dlsu.edu.ph

Andrea Nicole O. Ver
andrea_nicole_ver@dlsu.edu.ph

Jedrick L. Chua
jedrick_chua@dlsu.edu.ph

John Paul S. Yao
john_paul_yao@dlsu.edu.ph

Charibeth K. Cheng
charibeth.cheng@dlsu.edu.ph

De La Salle University - Manila

Abstract

As the number of sources of unstructured data continues to grow exponentially, manually reading through all this data becomes notoriously time consuming. Thus, there is a need for faster understanding and processing of this data. This can be achieved by automating the task through the use of information extraction. In this paper, we present an agent that automatically detects and extracts the 5Ws, namely the *who*, *when*, *where*, *what*, and *why* from Filipino news articles using a hybrid of machine learning and linguistic rules. The agent caters specifically to the Filipino language by working with its unique features such as ambiguous prepositions and markers, focus instead of subject and predicate, dialect influences, and others. In order to be able to maximize machine learning algorithms, techniques such as linguistic tagging and weighted decision trees are used to preprocess and filter the data as well as refine the final results. The results show that the agent achieved an accuracy of 63.33% for *who*, 71.38% for *when*, 58.25% for *where*, 89.20% for *what*, and 50.00% for *why*.

1 Introduction

Information can be found in various types of media and documents such as news [Cheng *et al.*, 2016] and legal documents [De Araujo *et al.*, 2013]. These documents provide different types of data beneficial to people ranging from field-specific professionals to the everyday newspaper readers. Thus, from the seemingly endless sea of unstructured data, it is important to be able to determine the appropriate information needed quickly and efficiently.

The process of automatically identifying and retrieving information from unstructured sources and structuring the information in a usable format is called Information Extraction. This task involves the use of natural language processing in the analysis of unstructured sources to identify relevant data such as named entities and word phrases through operations including tokenization, sentence segmentation, named-entity recognition (NER), part-of-speech (POS) tagging, and word scoring. This system is applied to various fields such as legal documents [De Araujo *et al.*, 2013], work e-mails [Xubu and Guo, 2014], and news articles [Cheng *et al.*, 2016].

Our information extraction agent automatically extracts the *who*, *when*, *where*, *what*, and *why* of Filipino news articles. *Who* pertains to people, groups, or organizations involved in the main event of the news articles. *When* refers to the date and time that the main event of the news article occurred. *Where* refers to the location *where* the main event took place. There can be one or more *who*, *when*, and *where* features in an article. On the other hand, *what* is the main event that took place while *why* is the reason the main event happened. There can only be one *what* and *why* for each article. Moreover, it is possible that there are no *who*, *when*, *where*, *what* or *why* features in an article if one does not exist. Figure 1 shows a sample article translated in English with the corresponding 5Ws.

However, the grammar of English and Filipino are not the same. Some of the nuances encountered in the latter are the differences in focus-subject order (i.e.

In: Proceedings of IJCAI Workshop on Semantic Machine Learning (SML 2017), Aug 19-25 2017, Melbourne, Australia.

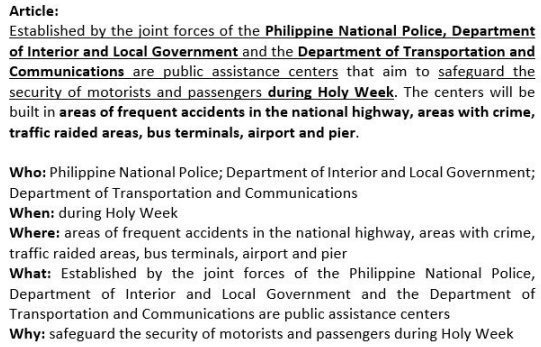

Figure 1: Sample Article Translated to English

verb first before performer) as well as the presence of ambiguous prepositions (i.e. "sa" can be applied to either a location or a date). Moreover, due to this, automatic translation of large data from Filipino to English is not feasible. Thus, our agent was designed to recognize and handle these linguistic features through a combination of machine-learned models and rule-based algorithms.

The results of this research can greatly benefit individuals and organizations reliant on Filipino newspapers such that they will be able to determine and aggregate essential information based on main events (as compared to mere presence) quickly and efficiently. Moreover, the research contributes an advancement in the field of natural language processing and semantic machine learning for the Filipino language.

2 Related Works

Information extraction has been performed in several previous studies dealing with a variety of languages and retrieving different kinds of information.

In a study by [De Araujo *et al.*, 2013], 200 legal documents written in Portuguese concerning cases that transpired in the RS State Superior Court were analyzed in order to determine the events that occurred. The events examined in these documents included formal charges, acquittal, conviction, and questioning. In addition, the study discussed how they put the legal documents through a deep linguistic parser and then represented the tokens in a web ontology language or OWL using a linguistic data model. Moreover, they described how after running documents through a deep linguistic parser and converting to OWL format, they formulated linguistic rules using morphological, syntactical, and part-of-speech (POS) information and integrated these to domain knowledge in order to produce a generally accurate information extraction system. Likewise, the study of [Xubu and Guo, 2014] described how they extracted information from descriptive text involving enterprise strategies such as e-mail, personal communication, and management documents through manual information extraction rule definitions in order to determine the efficiency of strategic execution.

Our agent also utilizes various rules and grammatical information such as POS and text markers for linguistic tagging. Similarly, [Das *et al.*, 2010] also adopted a rule-based information extraction in order to improve the overall accuracy of their information extraction system. However, unlike [De Araujo *et al.*, 2013] and [Xubu and Guo, 2014], they also used Machine Learning. They applied machine learning to their information extraction system through the use of a gold standard created by the matching answers of two annotators.

In 2012, [Dieb *et al.*, 2012] discussed how they used part-of-speech (POS) tagging as well as regular expressions to parse texts and determine orthogonal features within the considered nanodevice research documents. In addition, they discussed how after tokenizing and parsing the research papers, they made use of YamCha, a text chunk annotator, for machine learning in order to determine each of the parsed data category or tag (e.g. Source Material, Experiment Parameter) within an annotation automatically. Our agent also learns by example through several machine-learned classification algorithms derived from annotated Filipino news articles.

Furthermore, in the field of Filipino news, the research of [Cheng *et al.*, 2016] in 2016 extracted the 5Ws from Filipino editorials through a rule-based system in order to determine the possible candidates for each W and uses weight to choose among the list of candidates. They reported a performance of 6.06% accuracy for *who*, 84.39% for *when*, 19.51% for *where*, 0.00% for *what*, and 50.00% for *why*. However, the test corpus is composed of mostly true negatives and thus, there are only few examples as basis for implementation. Moreover, the candidates were subjected to minimal processing and filtering. Therefore, problems such as difficulty identifying correct candidates and low precision are present.

3 Information Extraction Agent Implementation

Figure 2 shows the architecture of the hybrid information extraction agent. A hybrid approach was implemented by means of utilizing a combination of machine learning techniques and rule-based algorithms.

A file containing a corpus of Filipino news articles acts as the agent's environment. The agent scans through the environment and gets all the Filipino news articles. Each article is then parsed and stored internally as a word table, which contains tokens with the

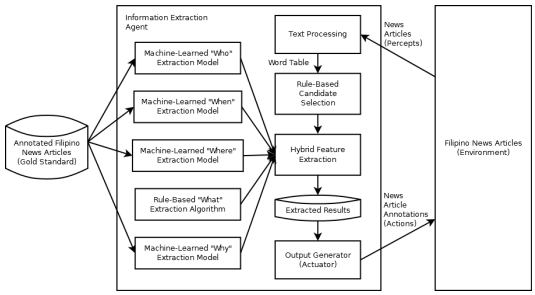

Figure 2: Hybrid Information Extraction Agent Architecture

corresponding position, POS and NER tags, and word score. The word table is passed to the candidate selection and feature extraction module to get the final who, *when*, *where*, *what*, and *why* for each article. The results are passed to the actuator that writes the corresponding annotations to the environment, which in turn saves the file and generates an inverted index file (Figure 3).

```
<data>
    <who>
        <entry>
            <text>Juan Ponce Enrile</text>
            <articleIndex>0</articleIndex>
            <articleIndex>155</articleIndex>
            <articleIndex>194</articleIndex>
            <text>Joseph Estrada</text>
            <articleIndex>1</articleIndex>
            <articleIndex>12</articleIndex>
            ...
        </entry>
        ...
    </who>
    <when> ... </when>
    <where> ... </where>
    <what> ... </what>
    <why> ... </why>
</data>
```

Figure 3: Sample Inverted Index File

3.1 Linguistic Tagging

Linguistic tagging is first applied to each news article and the parsed data is stored in a word table. The body of the article is initially segmented into its composite sentences and then individually tokenized. Each token is processed in order to determine the following information:

1. Part-of-Speech tag; e.g. proper noun (NNP), preposition (IN), determiner (DT)
2. Named-Entity tag, which includes person (PER), location (LOC), date (DATE), and organization (ORG)
3. Word score or frequency count

In order to assign each token its corresponding part-of-speech tag, a tagger was implemented using a model trained on news-relevant datasets from TPOST, a Tagalog Part-of-Speech Tagger [Rabo, 2004].

For named-entity recognition, each token is evaluated and assigned (if applicable) as a PER, LOC, DATE, or ORG. This process utilizes a Stanford NER model trained on 200 Filipino news articles.

Lastly, under linguistic tagging, word scoring is performed. Word scoring utilizes term frequency and counts how many times a token or word is encountered in an article.

3.2 Candidate Selection

Even though the articles have the named-entity tags assigned to particular words, these are not enough indicators of candidates. This is because named-entity tags do not consider grammatical information and, consequently, common nouns. Moreover, *what* and *why* candidates are sentence fragments that are composed of a variety of word tokens with different part-of-speech and named-entity tags, further indicating the need for the agent to perform candidate selection.

To select candidates, we use a rule-based approach to select possible candidates for the final *who*, *when*, *where*, *what* and *why* of each article.

A word or phrase is a *who*, *when* and *where* candidate when:

1. It is a noun or noun phrase
2. The word or phrase acts as a subject within the article
3. For proper nouns, it has a PER or ORG named-entity tag for *who*, DATE or TIME named-entity tag for *when* and LOC named-entity tag for *where*.
4. For common nouns, it is encapsulated by neighbouring markers including Filipino determiners, conjunctions, adverbs, and punctuations.

On the other hand, for the *what*, the agent simply chooses the first two sentences of the article's body as candidates. Lastly, for the *why*, the agent runs through the first six sentences of the article's body. Sentences where *why* feature makers are found are considered as the *why* candidates of the article.

3.3 Feature Extraction

Feature extraction is then performed to narrow down the candidate pool of the *who*, *when*, *where*, *what* and *why* in order to get the final results. A machine-learned model was trained and used for the *who*, *when*, *where* and *why* while a rule-based algorithm was developed for the *what*. Among the machine-learning algorithms tested include J48, Naive Bayes, and Support Vector Machine. Variations were also tested using boosting, bagging, and stacking. Moreover, several iterations involving feature engineering and parameter fine tuning

were done to get the optimal results for each algorithm based on true positive and accuracy rate among others.

Each of the *who*, *when*, *where* and *why* candidates pass through a machine-learned model which determines whether or not it is a final result. The models were generated using a gold standard composed of annotated Filipino news articles. Before being fed to the machine learning algorithm, however, the gold standard articles are pre-processed and filtered into candidates as discussed previously in order to better represent the data in a way such that the model can establish patterns better.

In order to do this, the gold standard articles were put through the same candidate selection module discussed previously and corresponding linguistic features were assigned to each candidate. The list of features that were tested include the following:

1. The candidate string
2. The number of words in the candidate
3. The sentence which the candidate belongs to
4. The numeric position of the candidate in the article
5. The distance of the candidate from the beginning of the sentence it belongs to
6. The frequency count of the candidate
7. 10 neighbouring word strings before and after the candidate
8. The part-of-speech tags of the aforementioned neighbouring words

In order to determine the class attribute (whether or not it is a final W), the candidate was matched against the annotations found in the gold standard to see if it matches. If it does, the class attribute is set to yes. Otherwise, it is set to no. These candidates, and their corresponding features, were used to train several models using different algorithms for testing. The features to be considered varied among the Ws, since not all of the listed features were proven useful in choosing the *who*, *when* and *where* results.

Furthermore, the algorithm that showed the best true positive and accuracy rate is J48 with boosting for *who* and J48 with bagging for *when* and *where*.

The model evaluates each candidate by assigning it an acceptance probability as well as a rejection probability. If a candidate's acceptance probability is higher than its rejection probability, it is added to the final *who*, *when* and *where* results.

Table 1: Final feature sets for *who*, *when* and *where*

	Who	When	Where
Candidate String	✓	✓	✓
No. of Words	✓	✓	✓
Sentence No.	✓	✓	✓
Position	✓		
Proximity	✓		
Word Score	✓	✓	✓
No. of Neighboring Words and their POS Tags	10	3	10

On the other hand, for *what*, a weighting scheme was implemented in order to determine the best candidate. This was done since we found that determining the *what* is more straightforward than the other Ws. Thus, feature engineering and fine tuning a machine learned model for this W is unnecessary and may even cause unnecessary complexities.

The implementation firstly determines the presence of the extracted *who*, *when*, and *where* and adds 0.3, 0.3, and 0.2 respectively to a candidate's score. The weights were chosen after several experimental iterations starting with neutral arbitrary weights of all 0.5. The *when* and *where* are extracted in a similar way to the *who* except for a few differences in parameters, values, and implementation. Secondly, the sentence number is considered. The formula for computing the additional weight based on sentence number is given below.

$$weight = 1 - (0.2 * sentenceNumber) \quad (1)$$

If the extracted *who*, *when*, and *where* found in the candidate is present in the title, an additional 0.1 is added to the candidate score.

The candidates are then trimmed based on the presence of a list of markers composed of Filipino adverbs and conjunctions that denote cause and effect. If one of the markers are found within the candidate, the candidate is trimmed. If the marker found is a beginning marker, all words before the marker including the marker itself are removed. On the other hand, if the marker is an ending marker, all words after the marker including the marker itself are removed.

The candidate with the highest weight is then chosen as the final *what* result for that article.

Lastly, for *why*, the candidates first undergo trimming and weighting. This is done since the machine-learned models are limited to the data that is fed to them. Thus, they require an associated rule-based al-

gorithm to pre-process the data before it is used for training or classification.

Words that come before starting markers and after ending markers are removed from the candidate. The presence of the extracted *what* and the markers were also given additional weights. The final feature set for in feature extraction of the *why* included the following:

1. The candidate string
2. The number of words in the candidate
3. The sentence which the candidate belongs to
4. The candidate's weighted score
5. The number of *who* features are in the candidate
6. The number of *when* features are in the candidate
7. The number of *where* features are in the candidate
8. 10 neighbouring word strings before and after the candidate
9. The part-of-speech tags of the aforementioned neighbouring words

Furthermore, the algorithm that showed the best true positive and accuracy rate is J48.

4 Results and Observations

4.1 Gold Standard

In order to train and evaluate the agent, a gold standard was created. This gold standard is composed of 250 Filipino news articles retrieved from the study of [Regalado *et al.*, 2013]. Each article was manually annotated with 5Ws by four annotators. For each disagreement where only two or less annotators agree, the four annotators deliberated the best annotation. In the case that the decision is split, the annotation is discarded and left blank, denoting ambiguity. The resulting annotated corpus was then qualitatively evaluated by a literary expert.

Table 2: Inter-annotator agreement for the *who*, *when*, *where*, *what* and *why*

Feature	Value
Who	59.35%
When	61.25%
Where	71.00%
What	74.40%
Why	70.40%

Based also on inter-annotator agreement, the *who* and *when* proved to be more ambiguous than the rest. Since, based on the observations of the annotations, the *what* can be found in the first two sentences, the annotators found it easier to choose the annotation for this and thus there was more agreement. On the other hand, because there are many possible *who* and *when* in an article, the annotators may have had a harder time choosing all the relevant *who* and *when* in an article thus leading to more disagreement. There is also a possibility of finding more than one possible *where* in an article, but based on the results it was easier for the annotators to identify the *where* in a given article.

4.2 Evaluation

After implementing the agent, the agent's results were compared against the gold standard comprising of 250 articles. For the true positive value, complete matches, under-extracted, and over-extracted annotations were included. The results can be seen in Table 3[1].

Table 3: Statistics for the *who*, *when*, *where*, *what* and *why*

	Who	When	Where	What	Why
CM	63.46%	67.53%	53.82%	40.4%	39.2%
UE	2.41%	4.43%	4.86%	12%	9.6%
OE	0.92%	0.74%	1.39%	36.8%	1.2%
CMM	33.17%	27.31%	39.93%	10.8%	50%
TPCM	59.23%	35.51%	11.11%	40.4%	10.8%
TPPM	3.19%	5.07%	6.06%	48.8%	10.8%
FP	4.78%	5.80%	21.89%	10.8%	10%
TN	0.91%	30.80%	41.08%	0%	28.4%
FN	31.89%	22.83%	19.87%	0%	40%
P	92.88%	87.50%	43.97%	89.2%	68.35%
R	66.18%	64.00%	46.36%	100%	35.06%
A	63.33%	71.38%	58.25%	89.2%	50%
F	77.29%	73.93%	45.13%	94.29%	46.35%

Based on the statistics shown, the *when* was able to obtain the highest complete match rate, while the *why* has the lowest. This was possibly because the *when* had only a limited number of frequent candidates that could be seen across the news articles (i.e. seven days in a week, twelve months, holidays, relative days), making it easier to identify the candidates.

For the *who* and *where*, both had slightly lower complete match rates compared to that of the *when*. The candidates produced seemed to be greater in number because of the many different possible *who* and *where* across articles. The reason is that people and places

[1]CM - Complete Match Rate; UE - Under-extracted Rate; OE - Over-extracted Rate; CMM - Complete Mismatch Rate; TPCM - True Positive for Complete Match; TPPM - True Positive for Partial Match; FP - False Positive; TN - True Negative; FN - False Negative; P - Precision; R - Recall; A - Accuracy; F - F-Score

of significance can change over time unlike the more constant *when* candidates. Thus, the candidate selection and feature extraction had a more difficult time in identifying the correct *who* and *where* candidates for the article.

On the other hand, the *what* has less than half complete matches. However, the combined number of complete matches and partial matches still greatly outnumber the number of complete mismatches. This is because during the implementation of the agent, it has been observed that most of the *what* can be found in the first two sentences of the article with 94.00% of the instances in the first sentence and 4.40% in the second. Thus, the primary problem for the *what* is the trimming of candidates in order to completely match what is needed (and annotated) based on the gold standard. In part, it is because the linguistic structure of Filipino makes it so that sometimes, adjectives and other descriptors become too lengthy that some important details may be considered insignificant by the agent and are thus trimmed off. On the other hand, some phrases are not trimmed because of the presence of details that may be unnecessary but are considered linguistically significant by the agent possibly because of misleading markers.

Moreover, the reason why the recall of the *what* is 100% is because the agent always extracts a *what* feature for each article. Since partial matches are also considered as true positives, all the gold standard annotations for *what* were considered extracted.

Lastly, for the *why*, it could be observed that it obtained a high amount of false negatives, which shows that the agent fails to detect the *why* in the article even if one is present in the article. The agent also has difficulty in identifying the correct *why* from the candidates. This could probably be caused by the lack of relations between the *why* and *what* candidates. The linguistic structure of some articles prove to be difficult because of the interchangeability of the potential *what* and *why*. Thus, the agent could get confused when a supposed *what* is actually a *why* which came ahead of a *what* candidate. Moreover, text markers denoting reason could be misleading the agent to deciding that the phrase that follows the aforementioned text markers is the *why*, which matches the extracted *what* when in reality, they are only related by proximity.

Furthermore, the *who* performed well using a machine learning approach for its feature extraction. An experiment supporting this was performed. The experiment involved comparing the final *who* results of two different evaluation runs wherein the first run utilized the machine-learned model while the second only relied on the candidate selection module. The results of the experiment show that the accuracy was 63.33% for the first run while it was 38.27% for the second run.

We did the same experiment for the *when* and *where*. For the *when*, the agent was able to achieve an accuracy of 63.35% on the first run compared to 16.17% it got from the second run. For the *where*, the first run with machine learning achieved an accuracy rate of 58.25% in comparison to the second run with an accuracy of 13.33%.

For the *why*, experiment results show that the accuracy of the *why* feature when run with machine learning algorithms went up to 50%, compared to the 47.60% accuracy it got with a rule-based feature extraction.

Table 4: Comparison between our hybrid approach and a rule-based approach using the data of the latter

Evaluation Metric	Complete Match	Under-Extracted
Hybrid Who	43.84%	2.46%
RB Who	6.06%	8.08%
Hybrid When	59.1743%	7.7981%
RB When	84.39%	0%
Hybrid Where	56.4593%	1.4354%
RB Where	19.51%	1.22%
Hybrid What	28.0%	31.5%
RB What	0.00%	5.88%
Hybrid Why	11%	7.5%
RB Why	50%	3.13%

Table 4 shows a comparison between the performance of our hybrid extraction agent and an existing rule-based extraction system [Cheng *et al.*, 2016], using the same test data. Based on the results above, our agent proved to be better than the previous system for the *who*, *when* and *what*.

For the *who* and *where*, in terms of candidate selection, the rule-based system only uses markers. On the other hand, our agent uses NER and POS tagging in addition to markers. Furthermore, for feature extraction, our agent uses a machine-learned model as compared to a weighting system to better filter out candidates.

For the *what*, instead of immediately constricting candidates in the candidate selection stage using markers (as done in the rule-based system), our agent retrieves entire sentences and trims the markers out during the feature extraction stage. Moreover, our agent utilizes other extracted features including the *who*, *when*, *where* and title presence as additional weights to better determine the final *what*.

For the *when* and the *why*, the results show that the existing rule-based feature extraction performed better than the machine learning. However, if the data

used to train the *when* was increased, it is possible to improve the results of the machine learning feature extraction.

5 Conclusion and Future Work

This paper presents a hybrid information extraction agent for automatically determining the 5Ws of Filipino news articles.

In conclusion, performing machine learning on *who*, *when*, *where*, and *why* was beneficial since the agent allows the models to choose which candidates are correct. The performance is also further supported by the associated pre-processing, filtering, and refining rule-based algorithms. Thus, if the model is iterated upon, the results may improve. On the other hand, using purely rule-based selection on *what* is beneficial since, based on the structure of most Filipino news articles, the *what* can be found in the first two sentences and there are common markers that can easily denote the feature.

The framework used in this study can be applied in extracting other information and features such as perpetrator-victim, crime-ridden areas, businesses or companies involved in a main event, among others from news articles. However, the agent's models and algorithms would need to be modified for the information. Specifically, rule-based algorithms may have a different set of parameters and values while machine-learned models would have to be re-trained on the domain corpus of the new data. Thus, the linguistic tagging, candidate selection, and feature extraction would need to be tested and modified based on the aforementioned corpus.

Future work for the study include integrating anaphora resolution in order to maximize the power of pronouns and other referential linguistic information. Moreover, an ontology consisting of known figures, locations, positions, and organizations in the Philippines can be incorporated to possibly improve the extracted information. Lastly, a larger and more diverse corpus of news articles can serve as examples and aid in training better models and for more exhaustive evaluation.

Acknowledgements

The authors gratefully acknowledge the Department of Science and Technology for the support under the Engineering Research and Development for Technology scholarship.

References

[Cheng *et al.*, 2016] Charibeth Cheng, Bernadyn Cagampan, and Christine Diane Lim. Organizing news articles and editorials through information extraction and sentiment analysis. In *20th Pacific Asia Conference on Information Systems, PACIS 2016, Chiayi, Taiwan, June 27 - July 1, 2016*, page 258, 2016.

[Das *et al.*, 2010] A. Das, A. Ghosh, and S. Bandyopadhyay. Semantic role labeling for bengali using 5ws. In *Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on*, pages 1–8, Aug 2010.

[De Araujo *et al.*, 2013] D.A. De Araujo, S.J. Rigo, C. Muller, and R. Chishman. Automatic information extraction from texts with inference and linguistic knowledge acquisition rules. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, volume 3, pages 151–154, Nov 2013.

[Dieb *et al.*, 2012] T.M. Dieb, M. Yoshioka, and S. Hara. Automatic information extraction of experiments from nanodevices development papers. In *Advanced Applied Informatics (IIAIAAI), 2012 IIAI International Conference on*, pages 42–47, Sept 2012.

[Rabo, 2004] V. Rabo. Tpost: A template-based, n-gram part-of-speech tagger for tagalog. Master's thesis, De La Salle University, 2004.

[Regalado *et al.*, 2013] R.V.J. Regalado, J.L. Chua, J.L. Co, and T.J.Z. Tiam-Lee. Subjectivity classification of filipino text with features based on term frequency – inverse document frequency. In *Asian Language Processing (IALP), 2013 International Conference on*, pages 113–116, Aug 2013.

[Xubu and Guo, 2014] M. Xubu and J.E. Guo. Information extraction of strategic activities based on semi-structured text. In *Computational Sciences and Optimization (CSO), 2014 Seventh International Joint Conference on*, pages 579–583, July 2014.