FILIET: An Information Extraction System For Filipino Disaster-Related Tweets

A Thesis Document
Presented to
the Faculty of the College of Computer Studies
De La Salle University – Manila

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Science in Computer Science

by
DELA CRUZ, Kyle Mc Hale B.
GARCIA, John Paul F.
KALAW, Kristine Ma. Dominique F.
LU, Vilson E.

REGALADO, Ralph Vincent
Adviser

March 10, 2015

Abstract

The Philippines is considered the social media capital of the world, and the role of social media has become even more pronounced in the country during disasters. Twitter is among the many forms of social media. As experienced, information and data shared through Twitter have helped individuals, institutions, and organizations (government, public, and private) during emergency response, in making decisions, conducting relief efforts, and practically mobilizing people to humanitarian causes. However, extracting the most relevant information from Twitter is a challenge because natural languages do not have a particular structure immediately useful when programming. Another problem that information extraction faces is that some languages, like Filipino, is morphologically rich, making it even more difficult to extract information. Therefore, the goal of this research is to create Filipino Information Extraction Tool for Twitter (FILIET), a system that extracts relevant information from disaster-related tweets composed in Filipino.

Keywords: information extraction, disaster management, Twitter

Table of Contents

List of Tables

List of Figures

List of Code Listings

1.0 Research Description

This chapter introduces the research undertaken in the field of Text Classification (TC) and Information Extraction (IE) in Natural Language Processing (NLP) for disaster management. This chapter is divided into four sections. The first section talks about the motivations and the problem that needs to be addressed. The second section discusses the objectives of the research. The third section details the scope and limitations of the study. Lastly, the fourth section tackles the significance of the research and its benefits to Philippine society.

1.1 Overview of the Current State of Technology

According to a report of the United Nations International Strategy for Disaster Reduction (UNISDR) Scientific and Technical Advisory Group, disasters have destroyed lives, properties, and livelihood across the world. Just between 2000 and 2012, about 2 million people have died during disasters and an estimated US$ 1.7 trillion in damages have been recorded. In the same report, the UNISDR posits the use and research of new scientific and technological advancements in disaster management (Southgate et al., 2013).

Social media are online applications and platforms that aim to facilitate interaction, collaboration, and sharing of content. Social media can be accessed by computers or by smart phones. In a study of Universal McCann and an analysis of 24/7 Wall St., LLC about social media, the Philippines received a high rank in most of the categories. This led to the country being dubbed as the "Social Media Capital of the World" (Universal McCann, 2008; Stockdale & McIntyre, 2011).

Social media play a vital role in disaster management. For example, after the Haiti earthquake in 2010, numerous posts and photos were published in various social media sites. Just 48 hours later, the Red Cross has raised US$8 million. Social media have also enabled the generation of community crisis maps and interagency maps. They are maps that work as intermediaries between the public and relief organizations (Gao, Barbier & Goolsby, 2011). Patrick Meier, a crisis mapper, makes use of social media to improve the efficiency of relief efforts. He launched the website MicroMappers[1], that quickly sorts through online data, from tweets to uploaded photos, and then displays the information on satellite maps, to assist in relief efforts during the disaster of Super Typhoon Haiyan (also called Yolanda) in the Philippines (Howard, 2013). To illustrate further how social media are significantly regarded, in a study commissioned by the American Red Cross[2], it was revealed that 74% of the respondents expect response agencies to answer social media calls for help within an hour.

Twitter is a social media microblogging platform where users can post statuses in real-time. In times of disaster, Twitter is used to share information regarding the disaster including updates on response efforts. As part of the Philippine disaster management for natural calamities, the government has released an official newsletter detailing the official social media accounts and hashtags[3]. Filipino Twitter users tend to post tweets about requests for help and prayer. Other tweets pertain to traffic and weather updates, related observations, and class suspensions. While some users prefer to post messages in English, a large number of users also communicate with their native language when tweeting during disasters (Lee et al., 2013).

Knowing that various emergency response organizations aim to, as much as possible, attend to all requests for help, it would be very important and beneficial to have a system that is capable of extracting relevant disaster relief operation information from the contents that are posted by Filipino

---

[1] MicroMappers digital disaster response system. http://micromappers.com/

[2] The American Red Cross, *Web Users Increasingly Rely on Social Media to Seek Help in a Disaster*, Press Release, Washington, DC, August 9, 2010. http://newsroom.redcross.org/2010/08/09/press-release-web-users-increasingly-rely-on-social-media-to-seek-help-in-a-disaster/

[3] Official Gazette of the Republic of the Philippines, *Prepare for natural calamities: Information and resources from the government*, July 21, 2012. http://www.gov.ph/crisis-response/government-information-during-natural-disasters/

netizens in Twitter. Furthermore, it would be very helpful to have an information extraction system that is able to mine relevant information from the language that is dominant in the disaster-stricken areas, which, in the case of the Philippines, is the Filipino language and, at the same time, support the way how content is posted in Twitter like having certain formats (having #tags), writing style (TXTSPK and code-switched styles), etc. In general, having this system can open up opportunities for improving how disaster relief operations are planned and conducted in the Philippines, and eventually, can help save lives.

1.2 Research Objectives

This section presents the general and specific objectives of the proposed research.

1.2.1 General Objective

To develop an information extraction system that extracts relevant relief effort information from disaster-related tweets.

1.2.2 Specific Objectives

The following are the specific objectives of the research:

1. To review different information extraction systems;
2. To identify the different types of disaster-related tweets and the relevant information needed in relief operations;
3. To review different NLP techniques that are applicable in pre-processing Twitter data;
4. To analyze different approaches used in implementing an information extraction system;
5. To evaluate existing tools and resources that could be incorporated in the information extraction components of the system;
6. To determine the metrics for assessing the performance or effectiveness of the information extraction system.

1.3 Scope and Limitations of the Research

The research is about the design of an information extraction system for the Filipino language. Review of various information extraction systems to know the different approaches to implementation was covered. Different existing domain-independent, domain-dependent information extraction systems were also analyzed to understand their components, architectures, and implementation. Additionally, this study examined information extraction for MRL to grasp the techniques used to extract from MRL given that the Filipino language is considered an MRL.

For the system to extract relevant information, the research determined which information details are deemed relevant in times of disaster, especially in relief operations. The research also identified the different types of disaster-related Tweets to support the task of discerning relevant information from the given tweets. Other researches on the use of Twitter in disaster management were also evaluated to aid in the formulation of ontologies of the information extraction system developed in this study.

In terms of system performance, the research looked into different natural language processing techniques used for data preprocessing before feeding them into the information extraction system. Examples of the NLP techniques are text classification and text normalization. Text classification is the process of automatically assigning a text or document into a predefined category based on its content (Özsu & Liu, 2009). Texts may need to be classified according to categories so that the system can use appropriate algorithm to extract the information. Text normalization is the transforming of ill-formed words into their canonical forms (Han & Baldwin, 2011). The information extraction system will need a text normalizer as data coming from Twitter are noisy. Most of the text has no structure, incorrectly spelled words, and invented terms.

Different information extraction techniques were also examined. Some of these are Named Entity Recognition (NER), lexical analysis, and conference analysis. Lexical analysis involves splitting up sentences into words and performing Part-Of-Speech tagging to each word (Grishman, 1997). NER is the classification of each word into a category (Zhou & Su, 2002). Co-reference analysis is the resolving of references for the pronouns (Grishman, 1997).

Existing NLP tools for building information extraction systems were also reviewed. Examples of these tools are OpenNLP and Lingpipe. OpenNLP is a machine learning based toolkit for the processing of natural language text that can support a number of common NLP tasks like tokenization, sentence segmentation, part-of-speech tagging, named entity extraction, chunking, parsing, and co-reference resolution (Apache Software Foundation, 2010). On the other hand, Lingpipe is a toolkit for processing text using computational linguistics that can perform certain tasks like finding names of people/organizations/event, classify Twitter data, and check spellings (Alias-I, 2011).

Lastly, metrics were determined to measure system performance.

1.4 Significance of the Research

Being the social media capital of the world, the Philippines generates a lot of diversified information that cannot be easily tapped because of the limited capabilities and tools that are available in processing the language unto which these information are written in, the Filipino language. With Twitter being one of the most commonly used social media platforms in the country, a new level of information dissemination has been established. With an information extraction system that is built for the Filipino language and at the same time for supporting texts that are found in Twitter, respective stakeholders can explore more possibilities and opportunities with regard to effectively utilizing such information from the web and use them for disaster management purposes.

From a disaster management standpoint, there are a number of advantages to having an information extraction system that is specifically made to work with Twitter texts that are written in the Filipino language.

First, respective stakeholders can collect disaster-related information in a way that is less strict because with an information extraction system built for the two languages, stakeholders can effortlessly accept and process information that are written in a much more natural and open way. With this, they can reach out to more people and to more places because they can have a system that can extract information from how Filipinos speak and communicate through the different social media platforms available, and to be specific, in Twitter.

Second, with an information extraction system, respective stakeholders can easily make use of the information that are written in the format of the different variations of the languages like the 'TXTSPK' and 'Code Switching'. With a custom-built information extraction algorithm, the information extraction system will be able to increase the probability of accurately and precisely extracting relevant information.

Third, the information that can be extracted from Twitter can be further utilized to help in disaster relief efforts. With a system that can further categorize tweets automatically can help in extracting more straightforward and meaningful information about the current state of disasters. Certain types of tweets can indicate a specific set of relevant information that can be extracted. Take, for instance, Disaster Information Tweets. Information that can be extracted from this kind of tweets can include, but not limited to, the type of disaster, location of disaster and etc. Or take, for instance, Casualty Report Tweets. Information like the number of casualties or the names of missing people can be extracted from this type of tweets.

Lastly, with can information extraction system that can organize the extracted relevant information, respective stakeholders can now expedite the process of conducting relief operations since they can be presented with information that has already been processed to be easily read and understood by the normal people. With this information extraction system, the process of consolidating necessary relevant disaster-related information can be more intuitive and faster.

1.5 Research Methodology

Scrum-based methodology, an iterative software development life cycle, was applied in the course of this research.

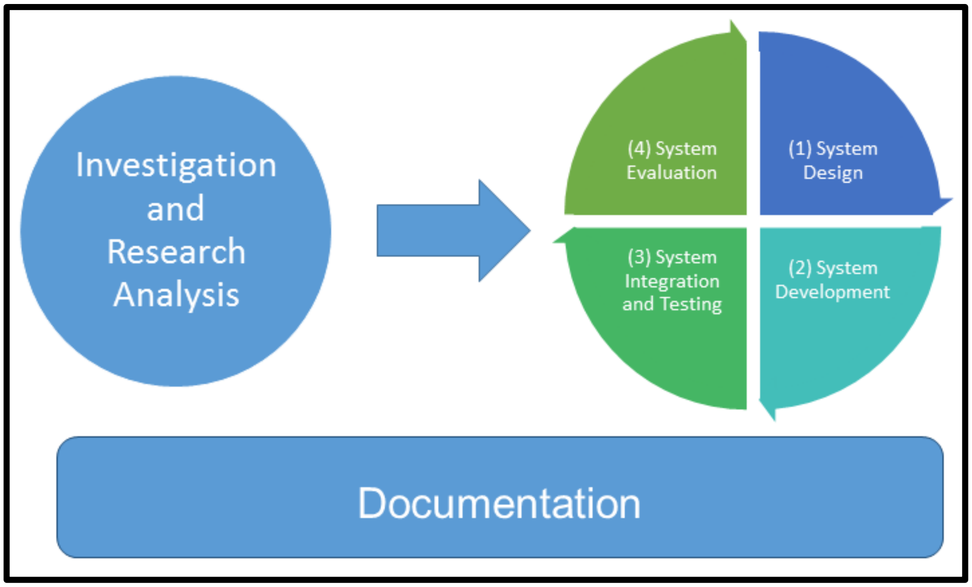

Figure 1-1. Research Methodology Phases

Figure 1-1 shows a diagram of the phases the research will undergo. The phases are as follows: investigation and research analysis, system design, system development, system integration and testing, system evaluation, and documentation. Regular consultation with the thesis adviser will also be conducted in order to keep the research on track for the whole duration of the thesis.

1.6 Investigation and Research Analysis

This phase involves the study and understanding of the fundamental knowledge of the concepts, algorithms, techniques, and tools that can be used to implement the system as well as identifying the modules and requirements of the system to be developed. The key activity involved in this phase is literature reviews of related works. From the review process, the techniques, tools, and metrics were then compared to verify which can be adapted to the system.

1.6.1 System Design

With the information takeaways from the earlier phase, system design commences. In this phase, appropriate architectures, algorithms, information extraction techniques, and other necessary tools are identified so that they can be effectively utilized in system creation. In addition, in this phase, necessary modules for the system are also identified based on the different processes and features that will be built into the system. The designs of the User Interfaces and the basic architecture for the databases are also covered in this stage. Finally, this phase also addressed data source identification for use and processing by the system. Once the data sources have been identified, data collection immediately commences.

1.6.2 Sprints

There is a two-week timeframe for each sprint to ensure that there is progress in the research. Each member is expected to produce a working output based on the tasks assigned to him during the sprint planning meetings. The tasks may vary from developing a part of the system to conducting further study regarding a certain concept.

1.6.3 Sprint Planning Meetings

At the beginning of each sprint, a sprint-planning meeting is conducted. Tasks that must be accomplished for the current sprint are discussed here. Included in these meetings is the assignment and division of the tasks among the members of the team. The evaluation of tasks from the previous sprint is also done here. If there are any unmet tasks, these will be carried over to the next sprint.

1.6.4 Scrum Meetings

Scrum meetings of 10-15 minutes in duration are conducted daily. The purpose is to update each member about what has or has not been accomplished yet in the assigned tasks. This ensures that there is daily progress and if there are issues that hinder members from accomplishing their assigned tasks so that they can be assisted.

1.6.5 System Development

From the design phase, system development follows. Data collection will also be done in this phase. Each team member is assigned to modules. The development of the system follows a scrum-based methodology wherein the system is developed in an iterative manner. Daily and weekly meetings, as well as regular consultations with the adviser, are conducted to assess the progress of the thesis and to plan the succeeding tasks.

1.6.6 System Integration and Testing

All the modules that have been developed are integrated into one system. This phase is also about unit testing processes for each module to ensure that there will be no significant bugs that can be found after integration processes are completed. After finishing integration, the system is then subjected to another round of tests to check again for any faulty integration and bugs.

1.6.7 System Evaluation

This phase is system performance evaluation following the metrics selected and reviewed. The following metrics have been identified so far: Precision, Recall, and F-measure results of the information extracted by the system. A number of tests of information extracted manually and those from the training set are undertaken to compare and validate results. The metrics can also be modified as needed depending on additional tests and findings in the future.

1.6.8 Documentation

Documentation of activities, methodologies, and of the system developed is important for monitoring and modification or improvement purposes. It will also be used for further reference, in case there is a need to validate or cross-reference any future work.

1.6.9 Calendar of Activities

Table 1-1shows a Gantt chart of the activities for the thesis period. Each bullet represents one week worth of activities.

Table 1-1. Timetable of Activities (April 2014 - April 2015)

Activities	Apr (2014)	May	Jun	Jul	Aug	Sept	Oct	Nov	Dec	Jan	Feb	Mar	Apr (2015)
Investigation and Research	_ * _ *	_ _ * *	_ * * *	* * * *	* * * *								
System Design					_ _ * *	_ * * *	* * * *	* * * *	* * _ _	_ * * *	* * _ _		
System Development					_ _ * *	_ * * *	* * * *	* * * *	* * _ _	_ * * *	* * _ _		
System Integration and Testing					_ _ * *	_ * * *	* * * *	* * * *	* * _ _	_ * * *	* * _ _		
System Evaluation					_ _ * *	_ * * *	* * * *	* * * *	* * _ _	_ * * *	* * _ _	* * * *	* _ _ _
Documentation	_ * _ *	_ _ * *	_ * * *	* * * *	* * * *	* * * *	* * * *	* * * *	* * _ _	_ * * *	* * * *	* * * *	* _ _ _

2.0 Review of Related Works

This chapter discusses the features capabilities, and limitations of existing research, algorithms, or software applications that are related or similar to this research.

2.1 Machine Learning-Based Information Extraction Systems

This part discusses information extraction systems that use machine learning-based techniques.

Machine Learning for Information Extraction in Informal Domains (Freitag, 2000)

The researchers of the paper explored one variation of the slot-filling problem, specifically how to find the best unbroken fragment of text to fill a given slot in the answer template. A definite template is given to an IE task. The template consists of fields that need to be filled with instances from the text source. The researchers set two ways of simplifying how to study the behavior of the algorithms to be developed: to isolate each field, learn the problem and focus on fields that are not instantiated or have a unique instance in a text source. With this, they found two primary aspects: multi-strategy learning and feature engineering. Multi-strategy learning because they believed that there is no single representation for all IE problems. Feature engineering because the ML of a feature set is needed to help adapt to domains containing novel structures because they will target informal domains. The researchers used four ML components: rote learning, term-space learning, learning abstract structure with grammatical inference, and relational learning for information extraction. They conducted experiments to gauge the performance of the four components.

In summary, the researchers found that it is possible to perform IE from informal domains found in the internet. Also, they stated that ML is a rich source of ideas for different algorithms that can be trained to perform IE. They have shown that with the right ML techniques, training effective extractors with very simple document representations is feasible.

TOPO - Information **Extraction System for Natural Disaster Reports from Spanish Newspaper Article (Téllez-Valero, 2005)**

This information extraction system extracts information related to natural disasters from newspaper articles written in Spanish. The system extracts the following information: (1) information related to the disaster itself (date, place, and magnitude), (2) information related to buildings (number of destroyed buildings, affected houses), (3) information related to people (number of casualties, missing or wounded), (4) information related to infrastructure (number of affected hectares, economic losses. It is able to extract information on natural disasters like hurricanes, forest fires, inundations, droughts, and earthquake.

The system uses general information-extraction system architecture. First, the document is turned into Boolean vectors representing the presence and absence of certain words. This stage is the document feature extraction. To limit the dimension, they used information gain technique. After conversion to a Boolean vector, classification follows. They used Support Vector Machine (SVM), Naïve Bayes (NB), C4.5, and k-Nearest Neighbors (kNN). After classification, text that might contain relevant information is selected. This stage is the candidate text selection. This process uses grammar to select the text and a dictionary of names and number to treat grammar exceptions. Then the output becomes candidates of relevant information. The system will then select which of the information will be used. This system uses the same algorithms in the text classification stage. They used different classifiers for different outputs.

This architecture boasts of its portability because it is language independent and domain adaptive. It is language independent because its training features and candidate text segments are based on simple lexical rules. It is domain adaptive because it only needs to change the training corpus.

In this work, the text filtering stage was evaluated on 134 news reports on the metrics of precision, recall, and F-measure. The algorithm that produced the best result was the SVM. They obtained an F-measure from 72% to 88% on the classification of news reports. The information extraction stage was evaluated on 1353 text segments that consist of names, dates, and quantities randomly taken from 365 news reports. The best classifier for the name and quantities was SVM, while it was kNN for dates. The overall system obtained an average of 72% on the F-Measure.

EVIUS (Turmo & Rodriguez, 2000)

EVIUS is a multi-concept learning system for free text that follows a multi-strategy constructive learning (MCL) approach. The system also supports insufficient amounts of training corpora. M-TURBIO is the multilingual IE system where EVIUS is its component. The system's input is both a partially parsed semantically tagged training corpus and a description of the desired target structure. The system's approach to learn is by using MCL with constructive learning, closed-loop learning, and deductive restructuring (Ko, 1998). EVIUS decides which concepts to learn and updates the IE rule sets continuously. The system uses FOIL (First-Order Induction Learning) (Quinlan, 1990) to create an initial rule set from a set of positive and negative examples. Positive examples can be selected using a friendly environment either as text and ontology relations. Negative examples are automatically selected. If any uncovered positive examples remain after using FOIL, this is because there are insufficient examples. The system tries to develop recall by growing the positive examples with artificial examples (pseudo-examples). Combining the uncovered example vector and a randomly selected covered vector makes a pseudo-example. This is done as follows: For each dimension, one of both possible values is randomly selected as the value for the pseudo-example. The new set of positive examples is now executed again using FOIL, the resulting set will be combined with the first rule set.

2.2 Rule-Based Information Extraction Systems

This part discusses information extraction systems that use rule-based techniques.

Vietnamese Real Estate (VRE) Information Extraction (Pham & Pham, 2012)

The Vietnamese Real Estate (VRE) Information Extraction system extracts information from Vietnamese real estate advertisements. It collects information like the type of estate, category of the estate, area, zone, price, name of the author, and contact details. The system uses the GATE framework for its architecture.

For its data, it has to pass specific criteria before it is fed into the system. First, input must be news articles related to real estate advertisements. Second, only one advertisement is allowed from each input data file. Lastly, it must be stripped off of all its HTML tags. After the data have met all the criteria, it will now go to data normalization. This process helps reduce ambiguity and assists in the annotation process. The necessary punctuation at the end of each sentence is also added. Second, it merges multiple paragraphs into one. Third, punctuations are normalized, redundant spaces are removed, and the first character after each punctuation is capitalized. Lastly, the telephone, price, area, and zone details are normalized to a common pattern. Upon completion, the data will now be manually annotated using Callisto, an annotation software.

After annotation, data are now ready to go to the information extraction system. It will go first through the tokenizer. The tokenizer will output two types of annotations, Word and Split. The Word annotation contains the part-of-speech, the word; it also checks if the first letter is capitalized, and has other features (kind and nation). This will be used to create the Java Annotation Pattern Engine (JAPE) rules. The Split annotation contains the delimiter. The next process is through the Gazetteer. Gazetteers are dictionaries that are created during system development and they include potential named entities (person, location) or categories, phrases used in contextual rules (name prefix or verbs that are likely to follow a person's name), and potential ambiguous entities. The output of the gazetteer is a lookup annotation covering the specific semantics. After this process, the text or data will now be passed on to the JAPE transducer. The JAPE transducer is responsible for extracting the information. It uses JAPE rules to recognize the entities that will need to be extracted. The annotated documents are the output.

The system has been tested using a lenient criterion and a strict criterion. An entity that is recognized correctly when the type is correct but the span overlaps in the annotated corpus is called the lenient criterion. On the other hand, an entity that is recognized correctly when the type and span are the same in the annotated corpus is called strict criterion. On the lenient criterion on test data, it registered 96% on the F-measure. While on the strict criterion, it registered 91% on the F-measure. The problem is on the data. The writing styles of the people are very diverse. The system has a problem in recognizing some of the entities like the zone entity because some of the zone entities are very long and do not observe capitalization.

Business Specific Online Information Extraction from German Websites (Lee & Geierhos, 2009)

The Business Specific Online Information Extraction System is a system that extracts information from the information pages of a German business website like its company profile, contact page, and imprint, and then identifies relevant business specific information. The system concentrates on the extraction of specific business information like company names, addresses, contact details, names of CEOs, etc. With regard to how the researchers pre-process their chosen input data, they interpret the HTML structure of documents and analyze some contextual facts to transform the unstructured web pages into structured forms. The approach applied by the researchers is quite robust in the variability of the DOM (for the web pages); it is also upgradeable and keeps data up-to-date. The evaluation metrics showed high efficiency of information access to the generated data. In conclusion, they stated that the developed technique is also adaptive to non-German websites with slight language-specific modifications, and experimental results from real-life websites confirm the feasibility of their approach.

In their proposed system, the researchers had two main modules for processing and extracting information from the German Information Web Pages: one for establishing a relational database storing company information and the other is for providing a query module. Within these two modules are three sub-processes that are done to further process the input data: (A) Localization of the Information Pages on the Web; (B) Document Analysis and Information Extraction; lastly, (C) Query Processing. In sub-process A (Localization of the Information Page), a web crawler is fed with the URLs of the web pages that are stored in the specialized database and then it fetches them from the web. Afterwards, the proposed system will then retrieve the document by following the anchor tags that lead to the information pages. On the other hand, in sub-process B (Document Analysis and Information Extraction), the fetched Information Pages are sent to an 'info analyzer' module which examines the HTML content of the page and then extracts the needed information bits. Here, the system exploits the internal structure of the named entities and uses sublanguage-specific contexts or attribute classes to identify the attribute-value pairs. Lastly, in sub-process C, the user of the system is given the right to query the database for information bits that he/she needs and then add these bits to the index.

For the Information Page Analyser (info analyser) in sub process B, the input data has to go through a number of processes to finally extract the information needed by the user. When given an Information Page, the analyser starts by pre-processing the frame structure and existing JavaScript of the page. Before creating the expressive DOM Tree, the HTML file of the page has to be validated and corrected, if needed, by using a special tool called 'tidy'. After doing so, the system will now be able to locate the minimal data region (or the data region of the information bit searched for) surrounded by a number of HTML tags which contain the information record being searched. By doing a depth- first traversal of the expressive DOM tree, the desired subtree can be isolated based on the headings of the data record like the following: “Herausgeber” (publisher), “Betreiber” (operator), “Anbieter” (provider), etc. The system was programmed to disregard domain-name irrelevant information; thus, the analyser will work further with a pruned DOM tree. After identifying the minimal data region, all information bits that are relevant to the domain name are extracted by using the Named-Entity Recognition technique and the attribute-value process (each attribute has a corresponding value that is indicated by the structure of the HTML file it is in) with respect to its external contexts and internal features. The system’s analyser module considers about 20 attribute classes and searches their corresponding values on the information page of business websites. The following are some of the attribute classes that are considered by the analyser: company name, address, phone and fax number, e-mail, CEO, management board, domain owner, contact person, register court, financial office, register number, value added tax number (VAT ID), and etc. After extracting the information bits needed from the pruned DOM trees, the information bits are then normalized to make sure that all information are consistent. The following are the classes that are affected by the normalization process: company names, legal form, register number, address (street, zip code, city), contact (phone and fax number, email), person name, and legal notification (tax number, VAT ID).

To conclude, the system performed surprisingly accurate with an average precision score of 99.1% and a recall score of 91.3% from a small test corpus that was composed of approximately 150 business web pages. The only encountered problem by the system was when value for certain attributes were erroneously represented like text in phone numbers, among others.

2.3 Template-Based Architecture

A template-based information extraction system uses templates to extract information. A template-based information extraction will only be able to extract information that is deemed important by the user. Its performance is based on how the user created the templates (Corney et al., 2008).

An Open Architecture for Multi-Domain Information Extraction (Poibeau, 2001)

Thierry Poibeau has provided a general architecture for developing information extraction systems regardless of its domain (Poibeau, 2001). In his paper, he proposed an information extraction architecture that takes advantage of the capabilities of machine learning to help researchers define new templates (this is where the extracted information is being filled in) with respect to the IE system’s domain.

Poibeau’s architecture is divided into 5 main modules: (1) the module for extracting information from the structure of the text; (2) the module for named entity recognition which is responsible for recognizing places/dates/etc.; (3) the module for the semantic filters; (4) the module for the extraction of specific domain-dependent information; and lastly, (5) the module for filling in a result template.

In module 1, a number of information are extracted from the structure of the input text. It is in this module where information that is embedded in the structure of the text is extracted such as those that are written in HTML or XML formats. On the other hand, in module 2, relevant information are extracted/recognized through linguistic analysis. This module is responsible for recognizing the different named entities present in the input text like names, places, and dates. Poibeau made use of the finite-state tool *Intex* to develop this module. Furthermore, in module 3, text categorization is performed on the set of so-called “semantic signatures” that were produced from a semantic analysis of the input text. Poibeau made use of the French system Intuition™ to develop this module. In addition, in module 4, specific information like the specific relationships between named entities are extracted by applying a grammar of transducers or extraction patterns on the input text. Lastly, in module 5, all the information extracted from the input text are linked together to fill in a specific result template(s) that present(s) a summarized view of the extracted information. Figure 2-1 illustrates the general architecture proposed by Poibeau.

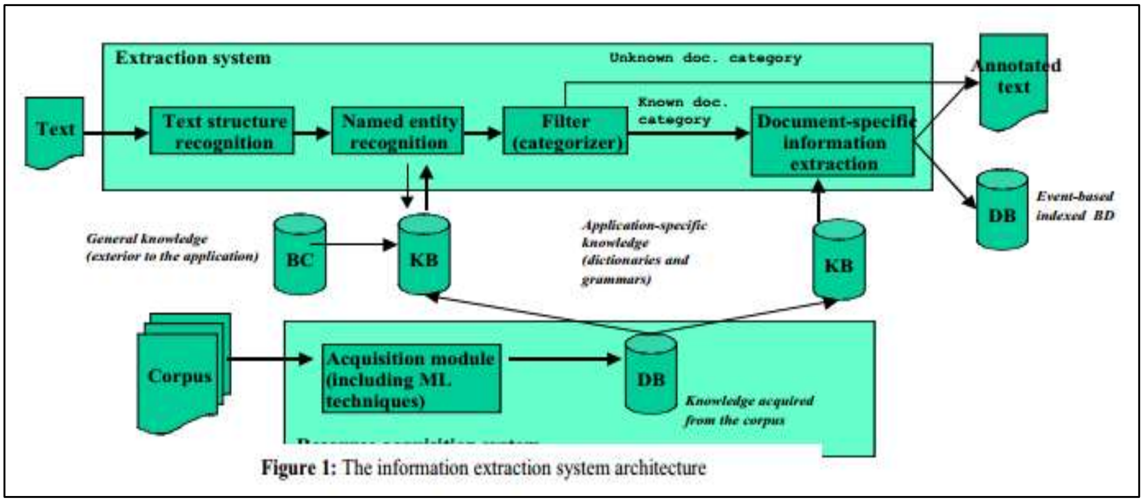

Figure 2-1. Poibeau's General Architecture

2.4 Ontology-Based Information Extraction Systems

This part discusses information extraction systems that use ontology-based techniques.

Ontology-Based Information Extraction (OBIE) System for French Newspaper Articles (Nebhi, 2012)

As most information extraction systems are based on the English language, it poses a problem for other languages in terms of limited tools available. To address this problem, the system maps the extracted entities to the ontology.

This system extracts names of persons, locations, and organizations from French newspaper articles. It collects data from LeMonde.fr. The system uses the GATE framework for annotation of entities in text and maps them to the ontology. It uses DBpedia databank that is based on Wikipedia projects. It contains 3,220,000 instances and is organized into a hierarchy of 320 classes and 1650 different properties. The system consists of 4 parts: pre-processing, gazetteer, rule-based semantic annotation, and the output. First, the system will pre-process the text. It will perform tokenizer, sentence splitter and POS tagger using the GATE application. After it is pre-processed, it will now go to the gazetteer. It will perform a lookup for the named entity recognition. After it passes through the gazetteer, grammar rules will be applied to create semantic annotation. The rules are written in JAPE which is part of the GATE framework. The system contains approximately 100 rules.

The system is evaluated using the Balance Distance Metrics (BDM) to consider ontological similarity. They manually annotated the documents using concepts on DBpedia ontology, and then compared it with the gold standard. They only evaluated person-, organization-, and location-named entities. The system scored an average of 0.94 on the BDM and achieved a 91% F-Measure.

2.5 Other Information Extraction Systems

This part discusses information extraction systems that use other techniques.

SOMIDIA - Social Monitoring for Disaster Management (Cheng et al., 2013)

SOMIDIA is a crisis-mapping system that focuses on plotting disaster on an interactive map in near real time. SOMIDIA collects data from different sources like news feeds, posts, SMS, blogs, and microblogs. One of the main components of SOMIDIA is its information extraction module. It extracts from both Filipino and English texts.

For the information extraction module, first, documents go through a tokenizer. They use OpenNLP to tokenize the document, then it will go to a sentence splitter. The sentence splitter accepts a list of tokens and an annotation list. It has a list of abbreviations so that the system can distinguish abbreviation periods from a period. The goal of the sentence splitter is to separate sentences by adding appropriate ending markers (period). The system uses OpenNLP's sentence splitter for its sentence detection. After the document has been split into sentences, it will go through a language guesser. They needed to differentiate English text from Filipino text because the language has different extraction techniques. They used frequency distribution of the words to detect the language. The output of the language guesser is the document with added metadata of the language. If the text is in English, the language guesser will pass the document to the POS tagger. Otherwise, it would be passed on to a Filipino NER.

For the English information extraction module, first it will go through the POS tagger. It uses the OpenNLP's POS tagger function. The output is a list of tokens with its corresponding POS tags. After the POS tagger, it will go through a 'chunker'. The chunker groups the tokens into their corresponding part-of-speech tag. This will be used to determine noun and verb phrases. It uses OpenNLP's noun and verb chunker. After chunking, it will pass through the English NER. The NER only focuses on proper nouns. It uses LingPipe because of its flexibility. LingPipe's NER uses three types of approaches, dictionary-based, rule-based, and statistic-based approaches. After the NER, it will go through co-reference resolution. The co-reference resolution will find the noun counterpart of the pronouns. It uses the Russian Mitkov algorithm for the resolution and WordNet for the lexicon. The normalization (standardizing data, collapsing of same sentences) will be done in this phase. The last step is the information extraction phase. It uses JAPE rules to extract the information, and the rules are paired with the two-tiered bootstrapping algorithm. The first tier bootstrapping algorithm starts with a small seed of words or rules. Then from the seed, it will try to learn the extraction pattern. The learned pattern will be used to generate a new extraction pattern. The process will then be repeated. The second-tiered bootstrap is responsible for keeping the most relevant extraction pattern.

For the Filipino extraction module, the document will go through the Filipino NER. They created their own NER because there is no existing Filipino NER tool. It uses dictionary-based and rule-based approaches for their NER. After tagging, it will now go through the Filipino extractor; the Filipino extractor has pre-defined rules (e.g. <event> sa <location>) that will extract the needed information.

The system is evaluated using precision, recall, and F-measure. They evaluated it on Tweets and news feeds. For English tweets, it scored a 75.17% F-measure on extracting disaster and 62.83% on extracting location. For Filipino Tweets, it scored 82.13% F-measure on disaster and 56.32% on extracting location. For news feeds, it scored 45.40% F-measure on English news feeds, while 38.82% on Filipino news feeds. The tweets scored higher because it is much easier to extract patterns on shorter text. The needed information will most likely be located near the text. On longer texts, the information needed might be located far away.

Table 2-1 shows a summary of all the reviewed information extraction systems. The table lists the system name, the language, and type of data it can extract, the domain, NLP pre-processing techniques, information extraction techniques, and evaluation metrics used by the system.

Table 2-1. Summary of Reviewed Information Extraction Systems

System	Language	Type of Data	Domain	Pre-processing Techniques	Information Extraction Techniques	Evaluation Metrics
Machine Learning for Information Extraction in Informal Domains (Freitag, 2000)	N/A	Documents (i.e. email)	Informal Domain	Not mentioned	Machine Learning-Based	Precision, Recall
TOPO - Information Extraction System for Natural Disaster Reports From Spanish Newspaper Article (Téllez-Valero, 2005)	Spanish	Free-text	Natural Disasters	Text Classification, Document Feature Extraction	Machine Learning-Based	Precision, Recall, F-measure
VRE Information Extraction System (Pham & Pham, 2012)	Vietnamese	Free text	Real Estate Advertisement	Text Normalization	Rule-Based	Precision, Recall, F-measure
Business Specific Online Information Extraction from German Websites (Lee & Geierhos, 2009)	German	Structured Text	Business Specific Information	Named Entity Recognition, Text Normalization, Attribute-Value Process	Rule-Based	Precision, Recall
Ontology-Based Information Extraction (OBIE) System (Nebhi, 2012)	French	Free text	News article	Tokenization, POS Tagging, Sentence Splitter	Rule-Based, Ontology	Precision, Recall, F-measure, BDM
Social Monitoring for Disaster Management (Cheng et al., 2011)	English, Filipino	Free text	News article, tweets	Tokenization, Sentence Splitter, Language Guesser	Machine-Learning Based	Precision, Recall, F-measure

2.6 Disaster Management – Relief Operations

This part discusses more about Relief Operations and the different information that are essential to this aspect of Disaster Management.

Humanitarian Knowledge Management (King, 2005)

This paper discusses the complexities and numerous challenges that many humanitarian organizations face whenever complex international humanitarian emergencies occur and how certain critical information in relation to disaster management activities, such as humanitarian assistance or relief operations can be utilized to help facilitate needed actions. King mentioned that the problem lies on the management of the data needed about these emergencies. In his paper, King stated that data management includes identifying, presenting, and disseminating critical information about the situation although such critical information, in itself, present a serious problem that could greatly affect data management. The problem lies in how this critical information is gathered: what information should be gathered and where should these be taken from? Upon efficiently identifying this in the early stages of these kinds of activities, as King mentioned, humanitarian organizations can more effectively make contingency plans and respond to natural disasters and complex emergencies and at the same time, potentially save a significant number of lives.

In the paper, a specific section was made to discuss what information are essential and crucial to different humanitarian organizations whenever they would conduct relief operations as a response to international complex emergencies like natural disasters and etc. According to King, humanitarian organizations like NGOs, UN agencies, local and national government, etc. need two specific types of information: (1) background and (2) situational information. Furthermore, information that is not within these types is more pertinent, relevant and critical to various specific personnel that are also within the said organizations. To support this claim, King gave an example through a scenario. He mentioned, "*policy makers want "big picture snapshot" analysis in order to understand the issues, to make decisions on providing assistance, and to be alerted to problems and obstacles*…*field personnel and project and desk officers in aid organizations, on the other hand, need more detailed operational and programmatic information in order to plan and implement humanitarian assistance and reconstruction programs.*"

With all of these, King listed down four main categories for the different vital information that is needed by organizations whenever they would conduct relief operations. Table 2-2 lists the four categories as well as their description and guide question that helps in determining which category the information belongs to.

2.7 Twitter and Disaster

This part discusses the uses of Twitter in times of disaster, the information that are useful during disasters, the information that can be extracted from disaster-related tweets and lastly, systems that make use of Twitter for disaster management procedures.

Extracting Relevant Information Nuggets from Disaster-Related Messages in Social Media (Imran et al., 2013)

This paper focuses on the extraction of relevant information from disaster-related tweets. The data set the authors worked with are Twitter data during hurricane Joplin on May 22, 2011 with #joplin. Their approach includes text classification and information extraction.

First, the tweets were classified into their respective categories. Table 2-3 lists the categories for the tweets. After filtering the tweets, only those of the Informative category were used. The informative tweets were further categorized into the information types they contained, which is listed in Table 2-4. The basis for the categories was from the ontology by Vieweg et al. (2010).

Table 2-2. Four Main Categories of Vital Information

Category	Description	Guide Questions
Situational Awareness	Information about the latest situation on the ground and information about the conditions, needs, and locations of affected populations	• What is the latest/current humanitarian situation in the country? • What are the most recent severity indicators? (Death tolls, mortality rates, malnutrition rates, economic impact, infrastructure damage, etc.) • Who are the affected populations (refugees, IDPs, children and other vulnerable groups, resident populations, etc.); how many are there, and where are they located? • What are the conditions and humanitarian needs of the affected populations? • What is the assessment of damage to infrastructure? (Transport, buildings, housing, communications, etc.) • What is the latest/current security situation in the affected areas of the country?
Operational / Programmatic	Information necessary to plan and implement humanitarian assistance programs	• Where are and what are the conditions of the logistical access routes for delivering humanitarian assistance? • Who's doing what where? What humanitarian organizations are working in the country, what are their programs, what are their capacities, and where are they working? • How is the host country/government responding and can it provide more? • What are the programmatic/financial needs of the humanitarian organizations? • What and how much are being provided to the humanitarian response organizations and who are the donors?
Background	Information about the unique history, geography, population, political and economic structure, infrastructure and culture of the country to be able to compare the emergency situation and conditions to previous normal conditions; and lastly	• What are the country's population (national, province/state, city/town) and composition (ethnicity, religion, age cohorts, urban/rural, political, etc.)? • What is the geography of the country? • What are the country's past disasters and natural hazards? • What are the most recent annual baseline health indicators for the population? (crude mortality rate, infant/child mortality rates, HIV adult prevalence, malnutrition, etc.) • What are the annual economic indicators? (GDP, GNP, agricultural/food production, staple food prices, etc.)
Analysis	Humanitarian information needs to be interpreted in context and related to other thematic information. Analysis can include evaluations of issues and responses, projections about the future, and recommendations for policies and actions	• What are the causes and contributing factors of the emergency? • What are the constraints to providing humanitarian assistance? (Insecurity, inaccessibility, government, interference, etc.) • How effective are humanitarian assistance programs and responses? • What are the future impacts of the emergency? • What are the options and recommendations for action?

Table 2-3. Tweet Categories

Category	Description
Personal Only	If a message is only of interest to its author and his/her immediate circle of family/friends and does not convey any useful information to other people who do not know the author.
Informative (Direct)	If the message is of interest to other people beyond the author's immediate circle, and seems to be written by a person who is a direct eyewitness of what is taking place.
Informative (Indirect)	If the message is of interest to other people beyond the author's immediate circle, and seems to have been seen/heard by the person on the radio, TV, newspaper, or other source. The message must specify the source.
Informative (Direct or Indirect)	If the message is of interest to other people beyond the author's immediate circle, but there is not enough information to tell if it is a direct report or a repetition of something from another source.
Others	If the message is not in English, or if it cannot be classified.

To classify the tweets into the categories mentioned, Naïve Bayesian classifiers were trained and implemented using Weka. Their features include binary features (if the tweet contains the '@' symbol, hashtags, emoticons, links or URLs, and numbers), scalar features (the length of the tweet), and text features (unigrams, bigrams, POS tags, POS tag-bigrams, and VerbNet classes).

For each informative tweet category, various types of information, referred to as information nuggets, were extracted. Table 2-5 shows the extractable information nugget per informative tweet category as well as that category's type subsets. The location references, time references, and number of casualties were extracted using the Stanford Named Entity Recognizer. All the Twitter Handlers (i.e. all words starting with the '@' symbol and URLs) were extracted from the tweet for the sources. Caution/Advice and Damaged Object were extracted using the Stanford Part of Speech Tagger and WordNet. For the intention of the tweet, another classifier was trained to determine if the tweet is a donation effort or a request for help. Lastly, the type information nugget pertains to the Type Subset column. For each informative tweet category, another classifier was trained to classify the category into its corresponding subset.

Table 2-4. Informative Tweet Categories

Category	Description
Caution and advice	If a message conveys/reports information about some warning or a piece of advice about a possible hazard of an incident. Example*: "Alerto sa Mayon Volcano, itinaas ng Phivolcs sa level 2"*
Casualties and damage	If a message reports the information about casualties or damage done by an incident. Example*: "Bush fires destroy 50 hectares in Baler, Aurora – NDRRMC http://t.co/Oc70OMeung49"*
Donations of money, goods or services	If a message speaks about money raised, donation offers, goods/services offered or asked by the victims of an incident Example*: "Repacking of Mineral waters! (@ Dano Residenza) http://t.co/iHUn4XA7jb"*

People missing, found, or seen	If a message reports about the missing or found person affected by an incident or a celebrity seen visiting ground zero Example: *"@philredcross missing joahnna nicole juliana ortiz sn isidro sulat eastern samar maytigbao church evacuation http://t.co/PGLnSEOtmY"*
Information source	If a message conveys/contains some information sources like photo, footage, video, or mentions other sources like TV, radio related to an incident. Example: *"VIDEO: Alert level 2, itinaas sa Mayon Volcano http://t.co/g6U5AziDFt"*

Table 2-5. Extractable Information Nugget per Informative Tweet Category

Informative Tweet Category	Information Nugget	Type Subsets
Caution and advice	Location references Time references Caution/Advice Source Type	Warning issued or lifted Siren heard Shelter open or available Disaster sighting or touchdown
Casualties and damage	Location references Time references Number of Casualties Damaged Object Source Type	Infrastructure Death Injury Unspecified No Damage Both Infrastructure and People
Donations of money, goods or services	Location references Time references Intention of Tweet Source Type	Money Blood Voluntary Work Food Equipment Shelter Discounts Other
Information source	Location references Time references Source Type	Photo Video Website TV Channel Radio Station Unspecified

Practical Extraction of Disaster-Relevant Information from Social Media (Imran et al., 2013)

Based on their previous paper Extracting Relevant Information Nuggets from Disaster-Related Messages in Social Media, after classifying the tweets into the informative tweet category, they extracted the information by employing a different approach. This time, they used two datasets: (1) tweets during hurricane Joplin on May 22, 2010 with #joplin and (2) tweets during hurricane Sandy on October 29, 2012 with #sandy #nyc.

To detect class-relevant information, they treated it as a sequence-labeling task. For each token in the tweet, they labeled it as either part of the relevant information or not. The (+) label indicates that the token is part of the relevant information while the (-) label indicates that it is not. After labeling, they applied Conditional Random Fields (CRF) to extract the information. A tool they also used in this paper is ArkNLP, a Twitter-specific POS tagger.

Safety Information Mining - What can NLP do in a disaster (Neubig et al., 2011)

In the article presented by Neubig and his team of researchers, they described the efforts of researchers in the field of Natural Language Processing in creating an information extraction system that aided in the relief operations during the 2011 East Japan Earthquake. The system that was described was primarily built to ease the mining of information regarding the safety of those affected by the earthquake from one of the most prevalent information source during that time, that is, Twitter. The system included subsystems that work for the following NLP and IE techniques like word segmentation, named entity recognition, and tweet classification.

The development cycle of the IE system has two phases: (1) resource-building phase and the (2) actual IE system development phase. To begin the development of the information extraction system, the researchers first started out by making the prerequisite resources for the system (or the resource-building phase). The researchers first focused on developing the different Language Resources and Tweet Corpus of the system. These language resources included dictionaries (used to improve the performance of the different text analyzers and classifiers in the system) and a labeled corpus of tweets (this contains safety information about the disaster and was used for the extraction from unlabeled tweets).

For the creation of the dictionaries, the researchers made use of the “Balanced Corpus of Contemporary Written Japanese” and the “UniDic dictionary” for general domain languages while the “Mozc Japanese Input Method Dictionary” and other publicly available resources like the last names specific to northeast Japan and the database of postal code were used for the domain-specific language. An additional list containing station names and locations, landmarks, etc. were made to aid in the extraction process.

For the creation of the Tweet corpus, the researchers collected tweets that contain the word ‘earthquake’, and those that contains the following hashtags: #anpi (safety information), #hinan (evacuation), #j_j_helpme (help request) and #save_<location>. To complete the corpus, the researchers tried to recognize the topic of the tweet (tweet classification) and the people mentioned in the tweet (named-entity recognition). To do so, the researchers defined nine classifications for the labels/topic of the tweets and are (1) I - Himself/Herself is alive; (2) L - Alive; (3) P - Passed away; (4) M - Missing; (5) H - Help request; (6) S - Information request; (7) O - Not safety information; (8) R - External link; and lastly, (9) U - Unknown.

After developing the prerequisite resources, the researchers proceeded with the actual development of the information extraction system. According to Neubig et al., the first step in IE for the Japanese language is Morphological Analysis. The MA is responsible for the tokenization and POS tagging of the tweets and for this, they made use of an open-source tool called KyTea. To accommodate the proper named-entity recognition in the Japanese language, the researchers trained the POS tagging model and replaced all proper nouns with subcategory tags (e.g. “first name”, “last name”, “place name”, and etc.) together with the introduction of a Conversational & News Text Corpus (containing a large list of Japanese first and last names). However, even though the POS tagging has been polished, the NER still failed to detect named entities that are grouped (NER still works on a word-by-word basis) that’s why the researchers made a simple rule-based system to accommodate the grouping of the Japanese named entities.

With all these, the researchers finally combined the two developed systems (the language resources and the MA system) to make the final information extraction system. The combination of the language resources with the MA system tends to increase the performance (accuracy) of the developed information extraction system by being able to accommodate the variations in the different styles in the different datasets that were used in this research.

3.0 Theoretical Framework

This chapter presents a discussion on the different theoretical concepts associated to information extraction systems, and as well as common architectures, approaches, modules, and resources needed in developing such systems.

3.1 Information Extraction

There is already huge amount information freely available in the internet. The problem is that people could not process these information easily because of the huge volume. It becomes more difficult as the information are written in natural language, which can be ambiguous. However, using an information extraction system, it can now automatically collect information from different sources like news, papers, and journals. Information extraction is the identification of the class of events or relationship and the extraction of relevant arguments of the event or relationship inside a natural language. It involves the creation of a structured representation of the facts that will be extracted. An information extraction system can only extract those facts that are represented (Grisham, 1997).

An information extraction system is divided into two parts, local text analysis and discourse analysis. Local text analysis is responsible for extracting the information from a text document. It consists of lexical analysis, name recognition, partial syntactic analysis, and scenario pattern analysis. Lexical analysis is responsible for splitting up the text into tokens. After splitting the text, it looks up a dictionary to fill out the part of speech and features of each token. After lexical analysis, it goes through name recognition. Name recognition is responsible for identifying proper nouns, aliases, and other special forms (dates and currency). It uses regular expressions that are stated in the POS, syntactic features, and orthogonal features to identify names. It also uses a dictionary that contains the list of proper nouns such as company names to easier identification. After going through name recognition, it passes through a partial syntactic analysis to identify some of the syntax of the text. It is responsible for identifying some of the like noun groups and verb groups. However, some systems do not implement a syntactic analysis. After syntactic analysis, it goes through scenario pattern matching. Scenario pattern matching is the extraction of related events or relationship relevant to the scenario. The outputs of the scenario pattern matching are two clauses. The first clause is a reference to an event structure while the second clause is a reference to a created entity (Grisham, 1997).

After going through the phases of local text analysis, it can now pass through the discourse analysis. Discourse analysis is the combination of all the information extracted during the local text analysis, and the formatting of the information. Under the discourse analysis are co-reference analysis and inference. Co-reference analysis attempts to resolve anaphoric references (pronouns and definite noun phrases). To determine which entity is referenced, the most recent previous mention of the entity is the anaphoric reference. After the co-reference analysis, it will undergo inference and event merging. Inference is responsible for making implicit information explicit. It uses system production rules to implement the inference module. After the inference, it can now be place in the data representation. Figure 3-1 shows the general flow of an information extraction system (Grisham, 1997).

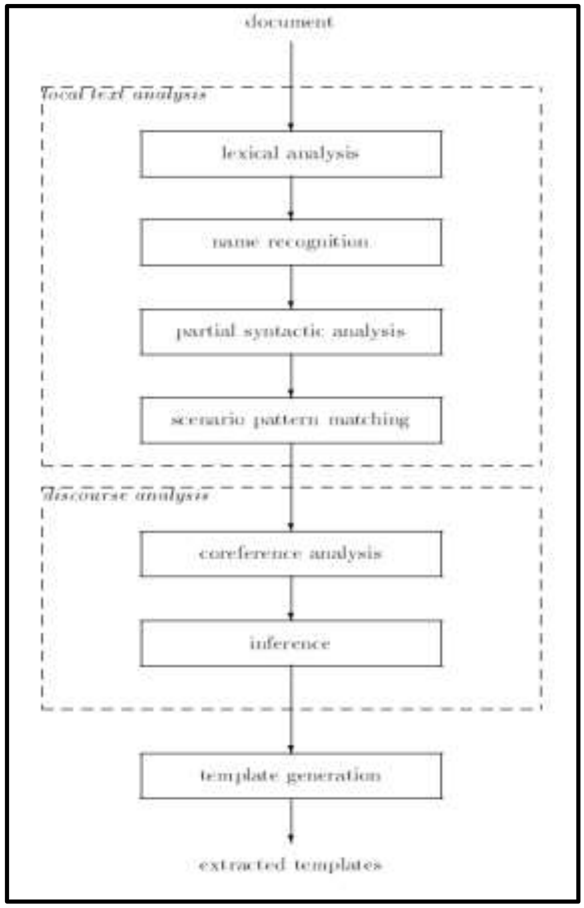

Figure 3-1. Structure of an Information Extraction System

3.1.1 Information Extraction Modules

This section explains the different modules that are commonly used in information extraction systems.

3.1.1.1 Tokenizer

Tokenizer is the module that segments a given text into tokens for further use in the natural language process. Tokens are usually the elements between spaces in the given input string. This module of natural language processing encounters several difficulties that need to be addressed such as tokenizing, email addresses, and uniform resource locators (URLs). Tokenizers today can identify that "15MB" is interpreted as "15 megabytes" if even there is no space between '15' and 'MB', and words with punctuation marks are also read correctly if tokenized. However, these tokenizers face two major problems, first is that the tokenizer performs its task independent of any knowledge, contained in the system. Another problem is that tokenizers are hard coded in the system. Thus, systems using these tokenizers end up tokenizing the input text without even caring whether the output of the tokenization made sense.

The researchers invented a tokenizer that validates the proposed output of the tokenization in a linguistic knowledge component, and this proposal validation repeats until there is no more possible segmentation or the text is validated. Lastly, the invented tokenizer also includes a language-specific data that contain a precedence hierarchy for punctuation (Bradlee et. al., 2001).

3.1.1.2 Sentence Splitter

The sentence splitter is a cascade of finite-state transducers that segments the text into sentences, and this module is used for the POS tagger (Cunningham et al., 2002). This module uses the set of regular expression-based rules that define sentence breaks like using periods, exclamation marks, and question marks (Zeng et al., 2006).

3.1.1.3 Normalizer

The presence of text speaks, slangs, and lingos is very high in SMS, social networks, and microblog sites. This presence makes it difficult for information extraction. In Aw and colleague's work (2006), they viewed text normalization as a specialized machine translation problem, called SMS Normalization. They see that text speaks, slangs, and lingos are just a variant of the English language. However, applying general machine translation will not work against SMS Machine Translation. General machine translation is based on non-standard words that have been well studied. However, with SMS, most of the lingos, for example "b4" (before) and "bf" (boyfriend) are not formally defined by linguistics yet. These words can still evolve as time passes by and more new text speaks, slangs, and lingos might be created by the younger generation.

There are two types of approaches used in Aw and colleague's paper (2006): basic word-based model and phrase-based model. In basic word model, an SMS word will be mapped to exactly one word. In phrase-based model, the SMS text will be split into k-phrases and the English words will also be split into k-phrases. Then, it will map the SMS phrase to an English phrase.

3.1.1.4 POS Tagger

The tagger produces a part-of-speech tag as an annotation on every word or symbol. These annotations produced can be used by a grammar checking tool to increase its power and coverage (Cunningham et al., 2002).

3.1.1.5 Gazetteer

The gazetteer contains lists of cities, organizations, days of the week, etc. It does not only contain entities, but also names of useful indicators, such as typical company designators (e.g. 'Ltd.'), titles, etc. The gazetteer lists are collected into finite state machines, which can match tokens (Cunningham et al., 2002).

3.1.1.6 Lemmatizer

Lemmatization is the reduction of inflectional forms and sometimes derivationally related forms of a word to a common base form. It uses vocabulary and morphological analysis to remove inflectional ending and return the root word (Manning et al., 2008). The traditional method of lemmatizing is to use morphological rules and dictionaries. However, with the presence of new words, it will be very difficult for the lemmatizer. Statistical method needs a large training corpus. StaLe is a lightweight statistical lemmatizer. In StaLe, the system produces result tokens based on the rules. Figure 3-2 shows StaLe's lemmatization process. Each token will be ranked according to its confidence factor and then pruned according to its candidate check-up phase. Those who pass will be the lemma of that word. However, if no token passed the candidate check-up phase, the input word will be the lemma. The problem with StaLe is that it sometimes produces a nonsense word resulting to a poorer outcome than a traditional dictionary-based lemmatizer.

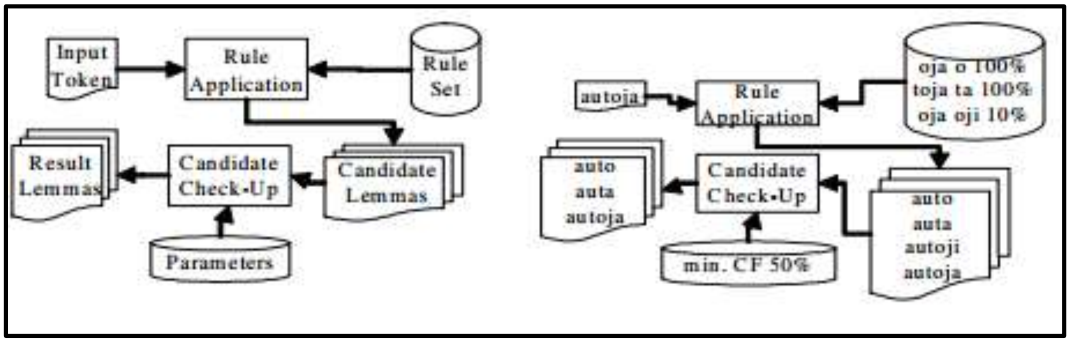

Figure 3-2. StaLe Lemmatization Process

3.1.1.7 Co-reference Resolution

This module consists of a main module and a set of submodules. The main module is responsible for initializing the submodules, and executes them in a particular order, then combines the results generated from the submodules, and eventually performs some post-processing over the result. There are three submodules in the main module: quoted-text module, pleonastic-it module, and pronoun-resolution module. The quoted text submodule recognizes the quoted fragments inside the text. The identified fragments are used by the pronoun-resolution submodule. The next module is the pleonastic-it submodule; it is responsible for finding pleonastic occurrences of “it”. The last and the main function of the co-reference resolution module is in the pronoun-resolution submodule. This submodule uses the results of the other two submodules after execution. The module works following an algorithm; first, it inspects the appropriate context for all candidate antecedents for this kind of pronoun and then chooses the best antecedent, if there is any. Then it creates the co-reference chains from the individual anaphor/antecedent (this step is performed from the main co-reference module) (Dimitrov, 2005).

3.1.1.8 Named-Entity Recognition

Named-entity recognition (NER) involves the automatic or semi-automatic processing of a series of words and then extracting or recognizing tokens in the text that refer to named entities (Lim et al., 2007). Named entities are phrases that contain the names of persons, organizations, and locations.

3.2 Information Classification

Text classification or information classification is the automatic classification of text into different categories based on their content. It consists of several important components: document representation, dimensionality reduction, classification algorithm, and performance evaluations (Shen, 2010). This will be useful as different types of text may need different types of extraction techniques.

3.2.1 Document Representation

Classification algorithms cannot understand texts directly. The text must be converted into some form that can be easily understood by the algorithm. There are different methods that could be used to represent documents. The traditional representation of documents is the Bag-of-Words (BOW) representation, which is based on the Vector Space Model. The use of BOW may vary as it can have different representations (Shafiei et al., 2007), one of which is word representation. In word representation, each word in the document is considered as a feature.

The problem with word representation is the 'curse' of dimensionality because text documents have a lot of unique words (Shafiei et al., 2007).

Another representation is term representation. Here, it uses multi-words or phrases as its feature. This drastically reduces the number of features. However, there has been mixed results on experimental results (Shafiei et al., 2007).

Character N-gram is another feature representation that could be used. Character N-gram takes n characters as a feature. Instead of focusing on the word, the character n-gram uses the characters. This makes model language independent. It is less susceptible to typographical errors and grammatical errors. It also does not require any linguistic preprocessing (Shafiei et al., 2007).

3.2.2 Dimensionality Reduction (Feature Selection)

The problem with text classification is the huge number of features present in the vector space. This huge number of features could drastically reduce the performance of the algorithm. It is important that when a number of features are reduced, accuracy is not sacrificed. The reduction of feature is called feature selection. There are different methods that could be used in feature selection.

Document Thresholding (DT) counts all the occurrences of each word in the document, then all the words that did not reach the specified threshold will be removed. The rationale behind this is that those words that have few occurrences are irrelevant (Wei et al., 2010).

Information Gain (IG) measures bits of information that could be gained in a document. The information gain of a word (w) is defined as:

$$IG(w) = -\sum_{j=1}^{K} P(c_j) log P(c_j) + P(w) \sum_{j=1}^{K} P(c_j|w) \log P(c_j|w) + P(w') \sum_{j=1}^{K} P(c_j|w') \log P(c_j|w')$$

where c_k is the set of all possible categories and $P(c_j)$ is the probability of a document classified into a category. This will be computed for all the words in the documents. Then, the words that did not reach the specified threshold are removed (Wei et al., 2010).

Mutual Information (MI) is the modeling of the word to a category. The mutual information criterion between term *t* and category *c* is defined as:

$$I(t,c) = log \frac{P_r(t \wedge c)}{P_r(t) P_r(c)}$$

and is estimated using

$$I(t,c) = log \frac{A \times N}{(A+C)(A+B)}$$

where,

A = number of times t and c co-occurs

B = number of times t occurs without c

$\mathrm{C} =$ number of time c occurs without t

$\mathrm{N} =$ number of documents

3.2.3 Classification

There are different classification algorithms that could be used in classifying text. One of which is the Bag-of-Word technique. In the work of Sriram et al., (2010), they classified short-text messages (Tweets) into news (N), events (E), opinions (O), deals (D), and private messages (PM). They used Bag-Of-Words to classify the tweets. First, they were able to extract 8 features: author, presence of shortening of words, slangs, time-event phrases, opinion words, emphasis on words, currency, and percentages. They used the author feature to determine the type of user. Corporate tweeters composed their message in a professional way. It uses less slangs, emotions, and shortening because they wanted to convey their message clearly. On the other hand, personal tweets contain usage of slangs, emotions, and shortening. These features can be used to distinguish corporate tweeters from personal tweeters. They collected 5407 English tweets, broken down into N = 2107, O = 625, D = 1100, E = 1057, and PM = 518. They also contained 6747 unique words. For the classification, they tried different setups: BOW, BOW and author feature (BOW-A), BOW and the seven features (BOW-7F), the 8 features (8F), and BOW and the 8 features (BOW-8F).

Another type of classification that could be used is the k-nearest neighbor (k-NN). k-NN is an instance-based lazy learner. It means it only trains when a new instance comes in. k-NN computes for the k nearest instances (neighbors). Then, k-NN will use the neighbors' categories to determine the class of the unknown instance. There are several ways to compute for the distance between the neighbors and the instances, Euclidean distance and Manhattan distance are some examples (Wajeed & Adilakshmi, 2011).

3.2.4 Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency – Inverse Document Frequency (TF-IDF) is a term weighting scheme that uses term frequency (TF) on the given document and its importance in relation to the whole collection or inverse document frequency (IDF) (Sammut, 2011; Zsu, 2009).

$$w_{ij} = f_{ij}[\log_2 N - \log_2 d_j]$$

where,

N: number of documents in the collection

d_j: number of documents containing term j

f_{ij}: frequency of term j in document i

W_{ij}: is the weight of term j in document i

3.3 Information Extraction Architecture

This section discusses the different architectures that can be applied in an information extraction system.

3.3.1 Adaptive Architecture

The problem with some information extraction systems (knowledge-based systems) is that they are not portable and are highly dependent to the domain. With sources rapidly growing and becoming more diverse, it will be very hard for an information extraction system to extract as these text are unstructured, especially given the natural language used. Another problem is that an error may propagate as it goes through each module, as the modules in information extraction architecture are cascaded. The use of machine-learning techniques tries to solve these problems. Adaptive Information Extraction systems use machine-learning techniques to automatically learn rules that will extract certain information (Turmo et al., 2006).

3.3.1.1 LearningPinocchio (Ciravegna & Lavelli, 2004)

LearningPinocchio is an adaptive information extraction system that uses induction rules to extract information. Machine-learning techniques are used to learn the rules over the training examples marked by XML tags. LearningPinocchio has two parts, preprocessor and modules. The preprocessor performs tokenization, lemmatization, POS tagging, and Gazetteer lookup. After doing the preprocessing, information can proceed to the modules. This is where the tags will be annotated. The modules may consist of NER, text zonings, and other IE tasks. Figure 3-3 illustrates the architecture used by LearningPinocchio.

Each module has three modes: training, testing, and production. Training mode is responsible for inducing the rules and learning how to apply IE rules in a specific scenario. The training mode accepts two inputs. First, it needs the module definition that includes a set of system parameters. Second is the preprocessed training corpus that has been tagged with XML. The output of the training mode is a set of rules that will be used in the testing and production modes. The testing mode is for testing on unseen tagged corpus. This mode tells how well the module performed in a certain application. The input for the testing mode is a module with induced rules and a test corpus that has been tagged with information that needed to be extracted. In this mode, it is still possible to retrain the model by adjusting the parameters to

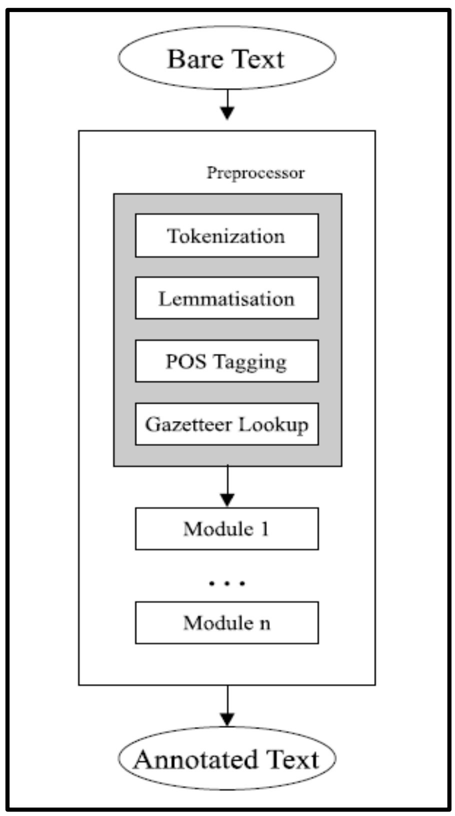

Figure 3-3. Architecture of LearningPinocchio

improve performance. The output is corpus tagged with XML and statistics on the performance of the module, and details of the mistakes as well. The production mode handles receiving the tagged corpus and the XML tags to the corpus.

For inducing rules, LearningPinocchio uses (LP)2 a covering algorithm especially for user-defined IE, to learn from training corpus marked with XML tags. It is a two-step process to induce the rules that will add XML tags to the text. Figure 3-4 shows the process of the inducing process of (LP)2. First, it induces tagging rules that will add preliminary tags. Second, it improves on the tagged rules by inducing correction rules.

A tagging rule consists of a left-hand side, which is the pattern of conditions of a sequence of words, and a right-hand side, which is the action that will insert the tags in the text. The rule-induction algorithm uses positive examples to learn the rules. Positive examples are instances that have been manually tagged by a specialist. For each positive example, the algorithm first initializes rules. Then, it will generalize the rules. Lastly, it will keep the best rules. The algorithm will repeat for each positive example. Information, like word window, lexical items, lemma, lexical category, lexical case, and user-defined semantic classes could be used as a condition in the initial rules. After getting the generalizations, they will be tested on training corpus to see if they will be accepted as best rules or contextual rules.

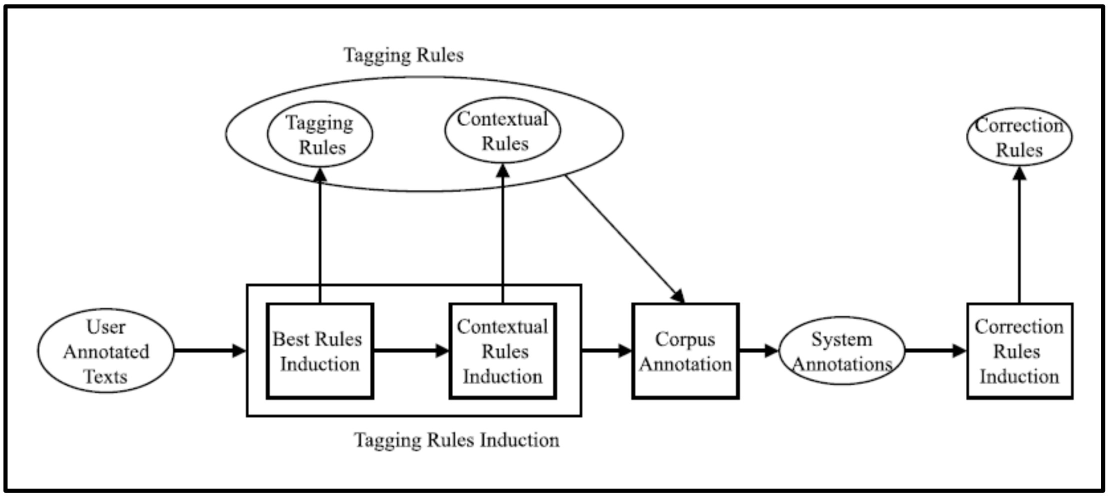

Figure 3-4. Rule-Induction Step

Best rules are rules that are highly dependable because they are able to cover most of the cases and their error rate is less than the threshold. These rules are sorted in decreasing number of covered cases. If the rules have the same number of matches, they are sorted according to their error rate. However, if they have the same number of matches and error rate, the one with the generic condition is preferred. The algorithm only keeps k generalizations. Although best rules can correctly tag the information, the problem is the low recall. The role of the contextual rules is to increase the recall without sacrificing precision. Contextual rules are additional rules that will correct the problem.

Correction-inducing rules are almost the same as the inducing rules. The difference is that the left-hand side of the correction-inducing rules contains the text and the tags and the right hand side, instead of adding tags, is shifting the misplaced tags. To select and apply the correction rules, the same algorithm as the inducing rules are used. Figure 3-5 illustrates the algorithm used by LearningPinocchio for choosing the best rules.

The information extraction process of LearningPinocchio consists of four (4) steps: initial tagging, contextual tagging, correction, and validation. The initial tagging will first tag the text. Next, the contextual tagging will further tag those that are missed during initial tagging, until no more tags can be placed. The third step will correct the errors. The last step will validate the tags. Figure 3-6 shows the process of the information extraction.

LearningPinocchio was tested in two languages, English and Italian. They trained the system on a corpus and tested the induced rules on unseen texts. The system was tested in two tasks: CMU Seminar announcements and Austin job announcements.On CMU Seminar announcements, tokenization and POS tagging were performed. A gazetteer was not done for a fair comparison. The IE must be able to extract the speaker's name, start time, end time, and location. They compared it to Rapier, symbolic-based (Califf, 1998), BWI, symbolic based, (Freitag & Kushmerick, 2000), SRV, WHISK (Soderland, 1999), and HMM, statistic-based (Freitag & McCallum, 1999). Based on the results, (LP)2 was able to achieve the highest score among the IE systems. (LP)2 was able to accurately extract the start time and end time, with 99.0% and 95% F-measures, respectively. However, it had difficulty in extracting the location and speaker's name with F-measures 77.6% and 75.1%, respectively. Overall, (LP)2 has the highest performance in All Slots with a score of 86.0%.

On Austin job announcements, the IE systems must be able to extract message ID, job title, salary offered, company offering the job, recruiter, state, city, and country where the job is offered, programming language, platform, application area, required and desired years of

```
method SelectRule(rule, currentBestPool)
  if (rule.matches≤MinimumMatchesThreshold)
     then return currentBestPool // i.e. reject(rule)
  if (rule.errorRate≥ErrorRateThreshold)
     then return currentBestPool // i.e. reject(rule)
  insert (rule, currentBestPool)
  sort(currentBestPool)
  removeSubsumedRules(currentBestPool)
  cutRuleListToSize(currentBestPool, k)
  return currentBestPool

method sort(ruleList)
  sort by decreasing number of matches
  if two rules have equal number of matches
     then sort by increasing error rate
  if two rules have same error rate and number of matches:
     then if one rule has more matches than a threshold
            then prefer the one with more generic conditions
          else prefer the other one
  return ruleList

method removeSubsumedRules(ruleList)
  loop for index1 from 0 to ruleList.size-1
      rule1=ruleList(index1)
      loop for index2 from index1+1 to ruleList.size
          rule2=ruleList(index2)
          if (subsumes(rule1, rule2))
            then remove (rule2, ruleList)
  return ruleList

method subsumes(rule1, rule2)
  return (rule2.matches is a subset of rule1.matches)

method cutRuleListToSize(list, size)
  return subseq(list, 0, size)
```

Figure 3-5. Algorithm for Choosing the Best Rules

experience, required and desired degree and posting date. The same preprocessing as with the CMU Seminar announcements was done. Based on the results, (LP)2 outperformed Rapier in almost all the aspects. Rapier was able to outperform (LP)2 in salary, desired year, and desired degree. However, in the overall performance, (LP)2 has a higher performance in All Slots with a score of 84.1%.

3.3.1.2 IE2 (Aone et al., 1998)

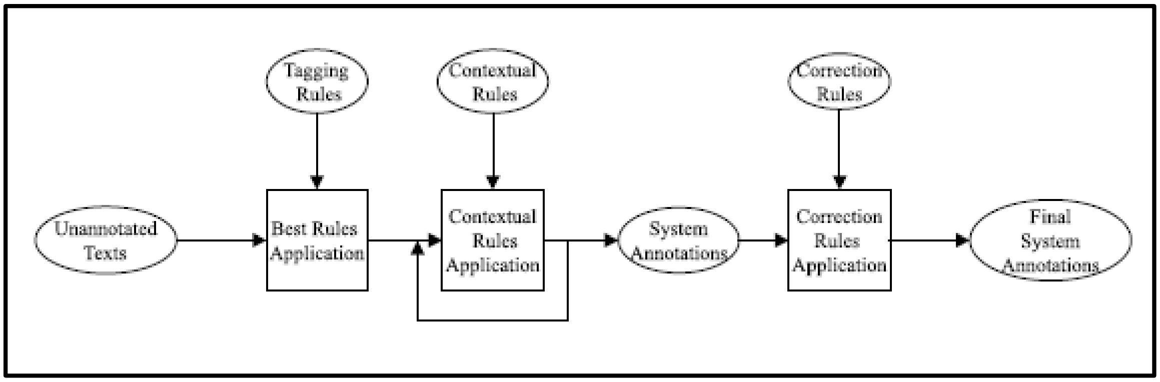

Figure 3-6. Information Extraction Process of LearningPinocchio

Aone and his team of researchers (1998) have presented an adaptive Information Extraction system that can be used to extract information from different types of texts like unstructured, structured, and semi-structured texts. In their article, they presented the architecture they used in building the system. Aone's IE system has six main modules in its architecture.

Module 1 is responsible for the named-entity recognition part of the IE system. For this module, they used a commercial tool called NetOwl Extractor 3.0 to recognize general named-entity types. It is in this module where time/numerical expressions, names (persons, places, organizations), acronyms (organization names, locations), and semantic subtypes (country, city) are being recognized and extracted. Module 2 or the Custom NameTag module is responsible for the recognition of restricted-domain named-entities by using pattern matching. The output phrases for this module are SGML-tagged (Standardized Generalized Markup Language) into the same input document. On the other hand, Modules 3 and 4 are responsible for SGML-tagging the phrases in the sentences that are considered to be values for the slots defined in the templates and they work hand-in-hand. Module 3 or the PhraseTag module works by applying syntactico-semantic rules to identify the noun phrases in the previously recognized/extracted named-entities. Module 4 or the EventTag module works by applying a set of custom-built syntactico-semantic multi-slot rules to recognize/extract events from the input sentence. Module 5 or the Discourse Analysis Module is responsible for co-reference resolution or the merging of the previously extracted noun phrases. This module is implemented using three different strategies so that it can be modified to reach optimal performance regardless of the extraction scenario. Strategy A or the Rule-Based Strategy uses a set of custom-built rules to resolve definite noun phrases and singular personal pronoun co-reference. Strategy B or the Machine Learning-Based Strategy uses a decision tree that has been formed from learning a corpus tagged with co-references. Strategy C or the Hybrid Strategy uses Strategy A to filter false antecedents and then uses Strategy B to rank the

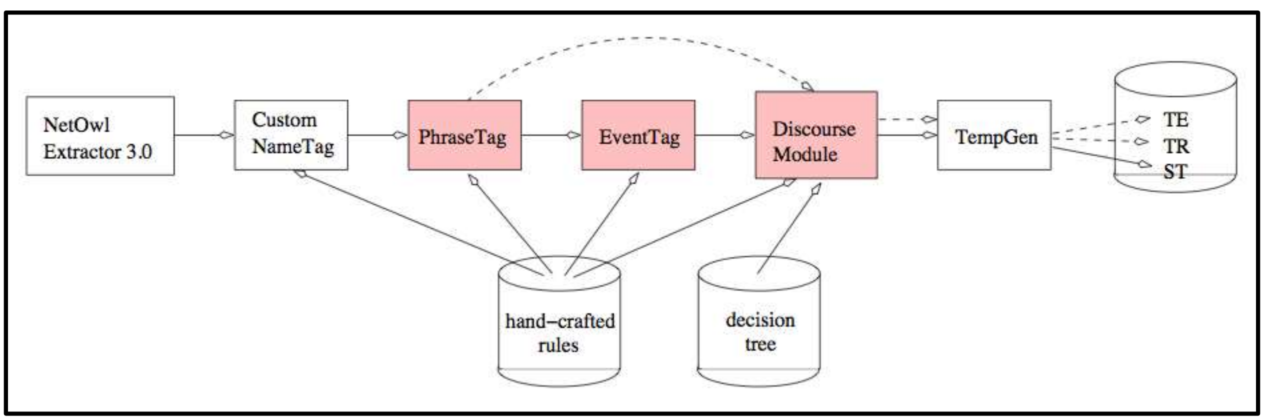

Figure 3-7. Architecture of IE^2 Adaptive Information Extraction System

remaining antecedents. In general, Module 5 is just merging the partial templates formed by the previous module. Lastly, Module 6 or the TempGen Module is responsible for the completion of the templates generated from the previous module by considering the consistency of the values in the slots of the event templates after resolving the noun phrase co-references and the generation of the output in the desired format. Figure 3-7 illustrates the architecture of the system proposed by Aone et al.

3.3.1.3 SOMIDIA (Cheng et al., 2013)

SOMIDIA uses an adaptive information extraction system that extracts relevant information (English and Filipino) from different sources (i.e. blogs, social media sites, news articles). After crawling the internet for documents, the documents are fed to the information extraction system. First, it performs a tokenizer. They used OpenNLP to do the tokenization (OpenNLP, 2013). Then, it goes through the sentence splitter. It accepts a tokenized document. The system will now split the document into sentences. They use OpenNLP for the sentence detection (OpenNLP, 2013). After the sentence splitter, the document will be classified into English documents or Filipino documents. This is done because different information extraction modules will be applied for English and Filipino. For English, they used POS Tagger, Chunker, English NER, Co-reference Resolution and English Extractor. For Filipino, they used Filipino NER and Filipino Extractor. The English information extraction process has POS Tagger, Chunker, English NER, Co-reference Resolution, and English Extractor. The Filipino information extraction process has Filipino NER and Filipino Extractor. For the Filpino NER, they build their own gazetteer for there is no existing gazetteer for Filipino. They used dictionary-based and rule-based approach in implementing the NER. Figure 3-8 describes the architecture of SOMIDIA.

For SOMIDIA to adapt to new instances, the rules must be adaptable. SOMIDIA has a pattern

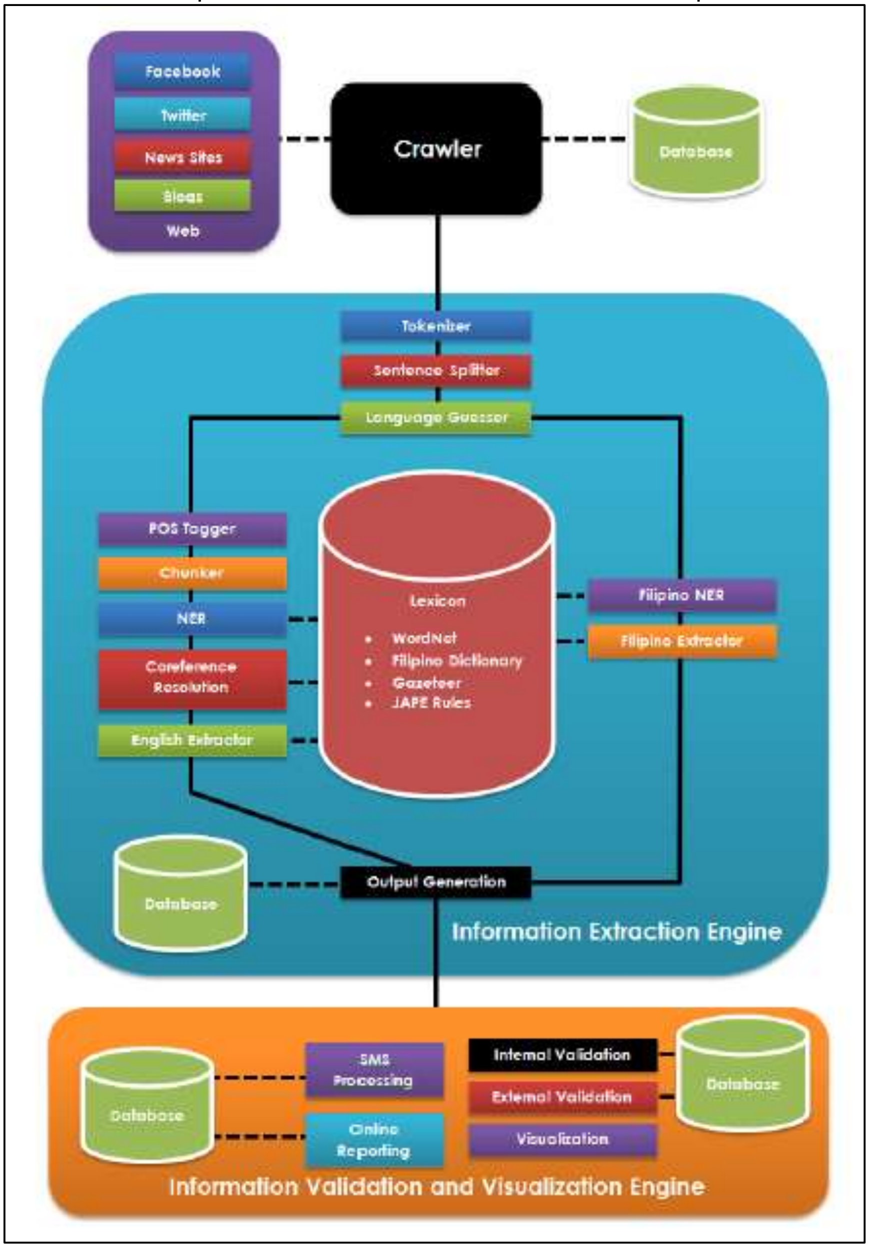

Figure 3-8. SOMIDIA Architecture

extractor module that is mainly responsible for extracting different patterns from a set of documents and seed words so that they can be later used for the extraction process. SOMIDIA defines a document as any text that is related to the domain of the extraction system. This module of the system works in this manner: For each document, it will identify first the seed words present in the document. Seed words are words that will be extracted. For each seed word identified, the module will try to generate possible rules by using Windowing, a term to describe the section of the document that is considered for computation. The module experiments with all possible combinations of tokens and window setups to produce as much rules by considering a number of windowing concepts like the minimum window size, maximum left window size, and maximum right window size. The minimum window size is the minimum number of tokens that is included in the window. In addition, the maximum left window size is the maximum number of tokens included in the window that is found to the left of the seed word. On the other hand, the maximum right window size is the maximum number of tokens included in the window that is found to the right of the seed word. After generating all possible rules from the combination of tokens and various window setups, it then stores the generated rules for that specific seed word in a HashMap together with the number of times the rules were generated. This process is done continuously until rules are generated for all the seed words in the document and until all of the documents are completely processed.

After the process of generating rules, the module will do some optimization of the rules generated to further improve the efficiency of the extraction module. The module will minimize rules by removing rules that fall into these two scenarios: (1) rules that occur only once because they are too specific and they would only work with a very small percentage of the documents and (2) rules that are able to extract more than its corresponding occurrence because these rules are too general and may have the tendency to extract irrelevant data.

3.4 Ontology

Ontologies are sets of classes (concepts), attributes, and relationships that are used to represent a domain knowledge. They are in a language (first-order logic) that can be abstracted from the data structures and implementations. Because ontologies are in the semantic level, they could be used to combine heterogeneous database, thus, making interoperability between systems possible (Gruber, 2009). Cimiano (2006) said that as the number of applications using ontologies is growing, then every such application must now be clearly and formally defined into an ontology.

Cimiano (2006) formally defines ontology as

$$O \coloneqq (C,\ \leq_C, \mathcal{R}, \sigma_{\mathcal{R}}, \mathcal{A}\sigma_{\mathcal{A}}, \mathcal{T})$$

where,

$C, \mathcal{R}, \mathcal{A}, \text{and } \mathcal{T}$ are disjoint sets, whose elements are called the concept identifier, relation identifier, attribute identifier, and data type, respectively.

$\leq_C$ $are\ semi-upper\ lattice\ with\ top\ element\ root_c\ called\ concept\ hierarchy$

$a\ function\ \sigma_r: \mathcal{R} \rightarrow C^+\ called\ relation\ signature$

$a\ partial\ order\ on\ \leq_R\ on\ \mathcal{R}\ called\ relation\ hierarchy$

$a\ function\ \sigma_A: A \rightarrow C \times \mathcal{T}\ called\ attribute\ signature$

$a\ set\ of\ datypes\ (i.e. strings, integer)$

In Vangelis et al. (2011), they presented four levels of classification on how an IE system exploited the ontology. The first level is the use of domain entities (including the variations), and word classes. For the first level, they can be represented by a gazetteer (flat) or ontologies (structured). By using ontologies, it can identify the text based on some constraints posed by the conceptual properties. An example system that uses the first level ontology is LearningPinocchio (Ciravegna & Lavelli, 2004). The second level uses concept hierarchies. In the second level, they focus more on taxonomic relations (consists of super/sub-ordination, is-a and part-of relationships). They could be used to generalize or specify extraction rules or check constraints. An example system is NAMIC (Basili et al., 2003). The third level uses the concepts' properties and relationships between concepts. These properties and relationships could then be used as guides for the information extraction process. An example system would be OBIE (Wang et al., 2005). The fourth level is the domain model. It combines the first three levels to be able to semantically interpret information. Domain models can merge with different structures, check consistency, make valid assumptions (for missing values), and discover implicit information. An example is BOEMIE (Maedche, 2002).

BOEMIE uses bootstrap or layered extraction process for its information extraction process. First, it extracts the entities, and then the relations. BOEMIE populates and enriches the ontology. It adds new individual entities and at the same time adds new concepts and relations.

3.4.1 Ontology Design

In creating a domain-specific ontology, the following tasks must be done: selection of domain and scope, consideration of reusability, finding important terms, defining classes and class hierarchy, defining properties of classes and constraints and creation of instances of classes (Saloun & Klimanek, 2011).

There are different approaches to creating ontology: hand-making by expert, automatic, and semi-automatic. In hand-made by expert, the ontology is manually done by the experts. Its advantage is that the result will be in high quality. However, they are very expensive and time consuming. In an automatic approach, the creation of the model is done by a machine. It is fast and low cost, but the problem is that implementing it will be very difficult and will result to inaccurate models. In a semi-automatic approach, the concepts and relationship will be generated by a machine, and the expert will complete and validate it. It produces relatively good results at a short amount of time. The disadvantage is that the machine-generated concepts and relations might be inaccurate and might also cause an inconvenience (Saloun & Klimanek, 2011).

3.4.2 Ontology Population

Ontology Population is the extraction and classification instances of classes and relationships of an ontology. There are three approaches for ontology population: manual, semi-automatic and automatic approaches. The manual population of ontology should be done by experts and a knowledgeable engineer. This could be costly and time consuming and the automatic approach might be inaccurate. For automatic and semi-automatic approaches, they have a common approach. They do entity name recognition, NLP techniques, and information extraction.

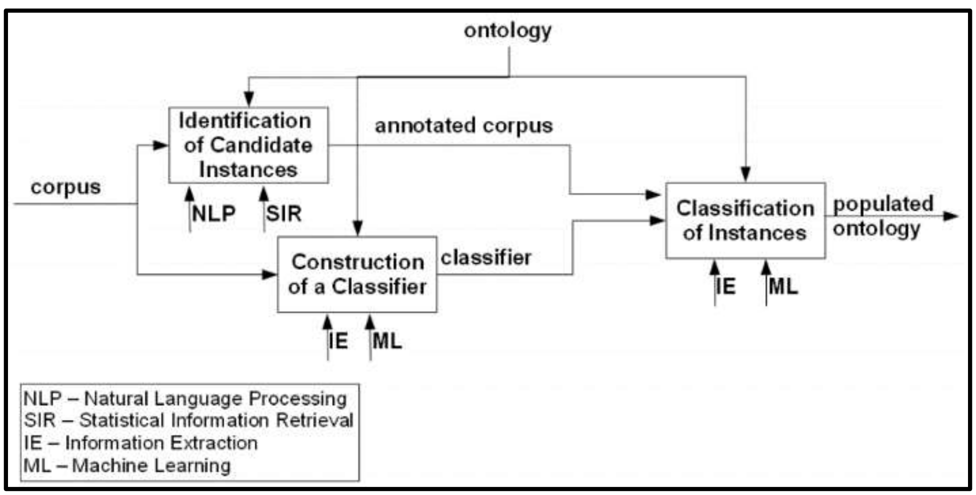

Figure 3-9. Process of Semi-Automatic Ontology Population

In Faria & Girardi (2011), the techniques they used are NLP and IE. The process has two phases: Extraction and Classification of Instances and Instance Representation. For the Extraction and Classification of Instances, all the possible relationships and class instances are generated. They consist of Corpus Analysis (Morpho-lexical analysis, Named-Entity Recognition and Co-reference), Specification of Extraction and Classification Rules, and Extraction and Classification of Instances. Then, they will manually generate a set of extraction rules based on the last task. After generating the rules, it will now use the extraction rules to look for text matching the patterns. This will now produce the instances(I'). After the first phase, it will now go to Instance Representation. Instance Representation has two tasks: Refinement of Instances and Ontology Population. For Refinement of Instances, it will try first to see if the instance already exists in the ontology. If it does not, then it will go to (I''). If it already exists in the ontology, it will look in (I'') to see if the instance needs to be updated. If it is, then it will be part of (I''). If not, it will be discarded. After refinement, the instance is now ready to fill the ontology. Given (I''), it will now look in the ontology to find the class. Then if a class is found, the instance will now be instantiated. Figure 3-9 shows the process of Faria & Girardi's (2011) semi-automatic ontology population.

3.5 Twitter[4]

Twitter is a microblogging social media platform wherein users may post messages of up to 140 characters long. Each of these posts is known as a "tweet". Mainly, a tweet is an expression of a moment or idea. Tweets may contain text, photos, and videos. Millions of tweets are shared in real time, every day.

A tweet may be replied to, retweeted, ‘favorited’, and may contain hashtags. A "reply" to a tweet is when another user comments or joins in the conversation of a tweet. A "retweet" is where you share the tweet of another user. A "favorite" indicates that a user likes the tweet. "Hashtags" assign a topic to the tweet. Thus, if one searches for #WorldYouthDay, the search results will contain all tweets with related topics about World Youth Day. When a Twitter user "follows" another user, this means that they subscribe to the tweets posted by that user (Twitter, n.d.).

3.5.1 Use of Twitter

Aside from Twitter's social media aspect, Twitter has been used as a source of data for various fields, one of which is in disaster management (Imran et al., 2013). Other fields that Twitter data have contributed to are linguistics (Mocanu et al., 2013), prediction (Tumasjan et al., 2010; Choy et al., 2012), real-time event detection (Sakaki et al., 2010), marketing (Jansen et al., 2009; Bollen et al., 2011), sentiment analysis, and opinion mining (Pak et al., 2010), education (Grosseck et al., 2008; Junco et al., 2011), newscasting (Phelan et al., 2009), medicine (Hawn, 2009; Chew & Eysenbach, 2010), and business processes (Culnan et al., 2010).

3.5.2 Twitter and Disasters

During disasters, Filipino Twitter users tend to retweet about request for help and prayer. Other tweets pertain to traffic updates, weather updates, observations, and class suspensions. While some users have a preference to post in English, there is still a larger number of users that use their native language when tweeting during disasters (Lee et al., 2013).

As part of the disaster management of the Philippines for natural calamities, the government has released an official newsletter indicating the official social media accounts and hashtags (Official Gazette of the Republic of the Philippines, 2012). Table 3-1 shows some of the official twitter accounts of government institutions as well as the official hashtags being used during disasters.

Table 3-2 shows the extractable information from the tweets per disaster.

Table 3-1. Examples of official government institution

Category	Official Government Institution Twitter Account	Unified Hashtag
Typhoon	@dost_pagasa	#(storm name)PH

[4]Twitter, a microblogging social media platform. http://www.twitter.com/

		(i.e. #YolandaPH, #GlendaPH)
Flood	@PAGASAFFWS, @MMDA	#FloodPH
Volcanic activities, earthquakes, and tsunamis	@phivolcs_dost	#EarthquakePH
Relief and rescue efforts	@PIAalerts, @PIANewsDesk, @NDRRMC_Open, @pcdspo, @DSWDserves	#ReliefPH #RescuePH
Suspension of classes	@DepEd_PH	#walangpasok

Table 3-2. Examples of disaster-related tweets with extractable information

Type of Disaster	Tweet	Extractable Information
Typhoon	@ANCALERTS: NDRRMC says 77 dead, 220 injured, 5 missing due to Typhoon Glenda #GlendaPH	• 77 dead • 220 injured • 5 missing • Typhoon Glenda
Typhoon	@ABSCBNChannel2: Bagyong Glenda patuloy na nagbabanta sa Luzon. #GlendaPH pic.twitter.com/2ygRWj6Z3D	• Typhoon Glenda • Luzon
Typhoon	@rapplerdotcom: #GlendaPH: Marikina River now at alert level 1 rplr.co/1mSTdnRpic.twitter.com/mECHfZfiyK	• Marikina River • Alert level 1
Typhoon	@ABSCBNNews: 200 families in Lagna lose homes due to 'Glenda' bit.ly/UfEDeO#southAlerts#GlendaPH	• 200 families • Laguna • Glenda
Earthquake	@dswdserves: DSWD Region 11 prepositioned 12,170 food packs&55,206 assorted food for victims of recent quake in Davao Occ. #EarthquakePH@dinkysunflower	• DSWD Region 11 • 12,170 food packs • 55,206 assorted food • Davao Occ
Earthquake	@phivolcs_dost: No expected damage from 6.1-magnitude #earthquakePH off Davao Occidental; aftershocks expected: bit.ly/1ra30ZZa	• magnitude • Davao Occidental
Earthquake	@manila_bulletin: BREAKING: 6.1 magnitude quake felt, east of Davao at 3:59PM. #EarthquakePH	• magnitude • Davao • 3:59pm
Earthquake	@seanbofill: Magnitude 6.1 earthquake recorded in Davao earlier today. #EarthquakePH	• Magnitude 6.1 • Davao
Flood	@saabmagalona:	• Ortigas st • La Salle GH

	Ortigas st across La Salle GH ankle-deep #floodph	• Ankle-deep
Flood	@MMDA: #FloodPH: As of 11:12 am, Orense to Estrella Southbound, leg deep, not passable to light vehicles	• 11:12am • Orense • Estrella Southbound • Leg deep • Not passable to light vehicles
Flood	@rqskye: @MovePH MT @PIAalerts 5m: #FLOODPH ALERT: Greenhills, La Salle Street, San Juan, Metro Manila: Knee-high. #TrafficPH	• Greenhills • La Salle Street • San Juan • Metro Manila • Knee-high
Flood	@rqskye: @MovePH MT @MakatiTraffic 11:27am: Flooded area in Brgy. Pio del Pilar: Medina St. corner... tl.gd/n_1s2geia |#FloodPH #TrafficPH	• 11:27am • Brgy. Pio del Pilar • Medina St. corner

3.6 Evaluation Metrics

This section discusses the different metrics that will evaluate the performance of the information extraction system.

3.6.1 F-measure

Precision and recall are the two primary metrics. Given a subject and a gold standard, precision is the percentage of cases that the subject is correctly classified as positive or true in the gold standard.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

Recall is the percentage of cases in the gold standard that is correctly classified as positive or true by the subject.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

The two metrics are often combined as their harmonic mean known as the F-measure (Hripcsak and Rothschild, 2005).

$$F = 2\,x\,\frac{precisionxrecall}{precision + recall}$$

The True positive category means a positive instance is correctly predicted as positive while the False positive category denotes a negative instance is predicted as positive. Then, the True negative category signifies a negative instance predicted correctly as negative while the False negative means a positive instance is predicted as negative (Davis and Goadrich, 2006). Figure 3-3 shows its confusion matrix.

Table 3-3. Confusion Matrix (Davis and Goadrich, 2006)

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

3.6.2 Kappa Statistics

The common way of summarizing inter-rater agreement among observers is the kappa statistics. It allows measurement not only by chance and the observed agreement beyond chance is divided by the maximum agreement (beyond chance) that is possible for the dataset. The general kappa formula is

$$k = \frac{p_o - p_e}{1 - p_e}$$

where p_o and p_e are the observed and expected proportions of agreement, respectively (Malpica et al., 2005).

3.7 Tools

This section discusses the different NLP tools that could be used in implementing the information extraction system.

3.7.1 ANNIE (Cunningham et al., 2002)

ANNIE or A Nearly New IE System is a system that contains different modules for NLP tasks. ANNIE is part of the GATE framework. ANNIE uses finite state transducers and JAPE rules to implement the modules. ANNIE has a tokenizer, gazetteer, sentence splitter, semantic tagger, and name matcher. This will be used for the POS tagger and the JAPE.

3.7.1.1 Gazetteer

The gazetteer contains the list of names, organizations, cities, days of the weeks, and others in plain text. It uses index files to access the lists which will be compiled in the finite state machines.

3.7.1.2 Sentence Splitter

The sentence splitter uses finite state transducers to split the text into sentences. It uses the gazetteer to check if punctuation is part of abbreviations or signals the end of the sentence. The sentence is annotated with the type “Sentence”; the breaks with “Split”. The sentence splitter is domain and application independent.

3.7.1.3 Part-Of-Speech (POS) Tagger

ANNIE POS Tagger uses a modified version of Brill Tagger. It uses lexicons and rule sets that have been trained in the Wall Street Journal corpus. However, the lexicon and rule sets can be changed based on the requirements. There are two additional lexicons, the lexicon for all caps and the lexicon for lowercase.

3.7.1.4 Semantic Tagger

The semantic tagger uses JAPE rules to annotate the entities. The grammar could be designed in such a way that it would recognize the entities. The output of the semantic tagger is the annotated text, which will be needed by the orthographic co-reference.

3.7.2 Weka (Weka 3, n.d.)

Waikato Environment Knowledge Analysis (Weka) is a Java-based open source collection of machine-learning algorithms that are used in data-mining tasks. It contains various tools for preprocessing, classification, regression, clustering, and visualization. It provides a library that could be used and it is also flexible as users can extend the API to customize the machine-learning algorithms (Weka 3, n.d.).

3.7.3 JENA API (McBride, 2002)

JENA is a semantic web application that helps in building ontologies. It is a Java-based API that handles OWL and SPARQL. It also includes inference engines based on OWL and RDF. This will be used to create and manage the ontology.

Code Listing 3-1 shows how to create an ontology.

Code Listing 3-1. Ontology Model Creation

```
OntModel ontModel = ModelFactory.createOntologyModel(<model spec>);
```

Code Listing 3-2 shows how to create a class.

Code Listing 3-2. Ontology Class Creation

```
Resource r = m.getResource(NS+"Paper");
OntClass paper = r.as(OntClass.class);
```

Code Listing 3-3 shows how to create object properties.

Code Listing 3-3. Ontology Object Property Creation

```
ObjectProperty hasProgramme = m.createObjectProperty( NS +
     "hasProgramme" );
hasProgramme.addDomain( orgEvent );
body.addRange( programme );
body.addLabel( "has programme", "en" );
```

Code Listing 3-4 shows how to create instance/individuals.

Code Listing 3-4. Ontology Instance Creation

```
OntClass c = m.createClass( NS + "SomeClass" );
Individual ind0 = m.createIndividual( NS + "ind0", c );
// second way: use a call on OntClass
Individual ind1 = c.createIndividual( NS + "ind1" );
```

3.7.4 ArkNLP (Gimpel et al., 2011)

ArkNLP developed by Carnegie Mellon is a Java-based Tokenizer and POS tagger that was specifically made for Twitter. For the tokenizer, it now identifies the emoticon tokens. For the

POS tagger, it can also tag slangs and emoticons. This will be used for tokenizing the tweets. Code Listing 3-5 shows a sample code of how to use the tokenizer feature.

Code Listing 3-5. Tweet Tokenization

```
List<String> tokens = Twokenize.tokenizeRawTweetText(text);
```

3.7.5 NormAPI (Nocon et al., 2014)

NormAPI is a text normalization API that is specifically built for the Filipino language. It currently has implementations for Dictionary Substitution Approach (DSA) and Statistical Machine Translation (SMT). The user can choose if the normalization will perform: (1) DSA only, (2) SMT only, (3) SMT after DSA, or (4) SMT before DSA. NormAPI accepts file or text as input. It also allows setting configuration files and training a new model. This will be used for the text normalization.

Code Listing 3-6. Text Normalization with NormAPI

```
String normalizedText = NormAPI.normalize_Text(shortcutText);
```

4.0 The FILIET System

This chapter presents the proposed system. It is divided into six sections. The first section will discuss the system overview. The second section outlines the objectives of the system. The third section tackles the scope and limitations of the system based on the outlined objectives. The fourth section presents the architectural design. The fifth section discusses the front-end and back-end features. Lastly, the sixth section will present the resources that will be used in implementing the system.

4.1 System Overview

Filipino Information Extraction for Twitter (FILIET) is a hybrid information extraction system that incorporates the architectures of an adaptive IE system and a rule-based IE system for Filipino disaster related tweet. The FILIET system will work with extracting information from tweets that were written in Filipino and English, along with their variations such as TXTSPK and code-switch. The system will follow the methodology described below. The disaster-related tweets will be loaded into the system. The system will then classify according to the following categories: (1) caution and advice, (2) casualties and damage, (3) donations, (4) call for help, and (5) others. The tweets will now proceed to the information extraction engine of the system wherein the system will extract relevant information from the tweets with regard to its given type of disaster. Extracted information from the given tweets will vary based on the type of information the tweet contains.

4.2 System Objectives

This section will discuss the objectives of the system.

4.2.1 General Objective

To develop an information extraction system that extracts relevant information from disaster-related tweets and considers the different available variations of the Filipino language.

4.2.2 Specific Objectives

The following are the specific objectives of the system:

1. To preprocess the tweets;
2. To extract relevant features from the tweets;
3. To classify the tweets into according to their content (i.e. caution and advice, casualties and damages, donations, and others);
4. To extract relevant information according to the type of tweet.

4.3 System Scope and Limitations

The system to be developed in this research is expected to be able to do a number of tasks that are within the scope of extracting information from Filipino disaster-related tweets. These tasks include the following: Text Preprocessing, Feature Extraction, Disaster Classification, and actual Information Extraction.

The system must be able to perform some preprocessing techniques onto the input tweet. These preprocessing tasks shall be limited to the following: (1) text normalization to include support for input tweets that were written in the TXTSPK format; (2) text tokenization, to enable word level analysis of the input tweet; (3) part-of-speech tagging, to enable semantic level analysis of the input tweet; (4) named-entity recognition, to enable proper identification of named-entities; and lastly, (5) disaster keyword tagging, to enable proper recognition of disaster words in the input tweet. Lastly, by looking at the initial data and from the study of (Lee et al., 2013), it was observed that a high probability that Filipinos will post tweets in the Filipino language and that TXTSPK and code-switching were the variations being used.

Moving on, the system must be able to extract features from the input tweet. The features that will be extracted from the input tweet are categorized into two: (1) binary features, those that have discrete values 0 and 1; and (2) nominal features, those that have continuous values. For the binary features, they will be limited to the following: Presence features (presence of keywords like disaster words, mentions, hashtags, emoticons, retweets, and Code Switching).On the other hand, the nominal features will be limited to the following: (1) Tweet length; (2) User; and lastly, (3) Location.

Using the extracted features, the system must be able to classify the input tweet based on the type of tweets. The tweet must be classified into the following: caution and advice (CA), casualties and damage (CD), donations (D), and others (O). This is important because each type of tweet will have different extracted information. The categories are based on Extracting Relevant Information Nuggets from Disaster-Related Messages in Social Media by (Imran et al., 2013).

The system must be able to extract two types of information from the given input tweet. The two main types of information are (1) General Information and the (2) Type-Specific Information. For the General Information, only the location references, time references, and source shall be extracted from the input tweet. On the other hand, for the Type-Specific Information, the following shall be extracted from the input tweet: (a) for caution and advice tweets: the caution and/or advice part of the tweet; (b) for casualties and damage tweets: the number of casualties and the damaged objects; and (c) for donation tweets: if the tweet is a donation effort or a request for help and what are the objects being donated or requested. The information to be extracted are also based on the study by (Imran et al., 2013).

The data that will be used in the development of the system will come from the Twitter Web Crawler developed by the De La Salle University - College of Computer Studies as well as from the crawler to be developed by the group. The system will only be processing data that are written in the Filipino language.

4.4 Architectural Design

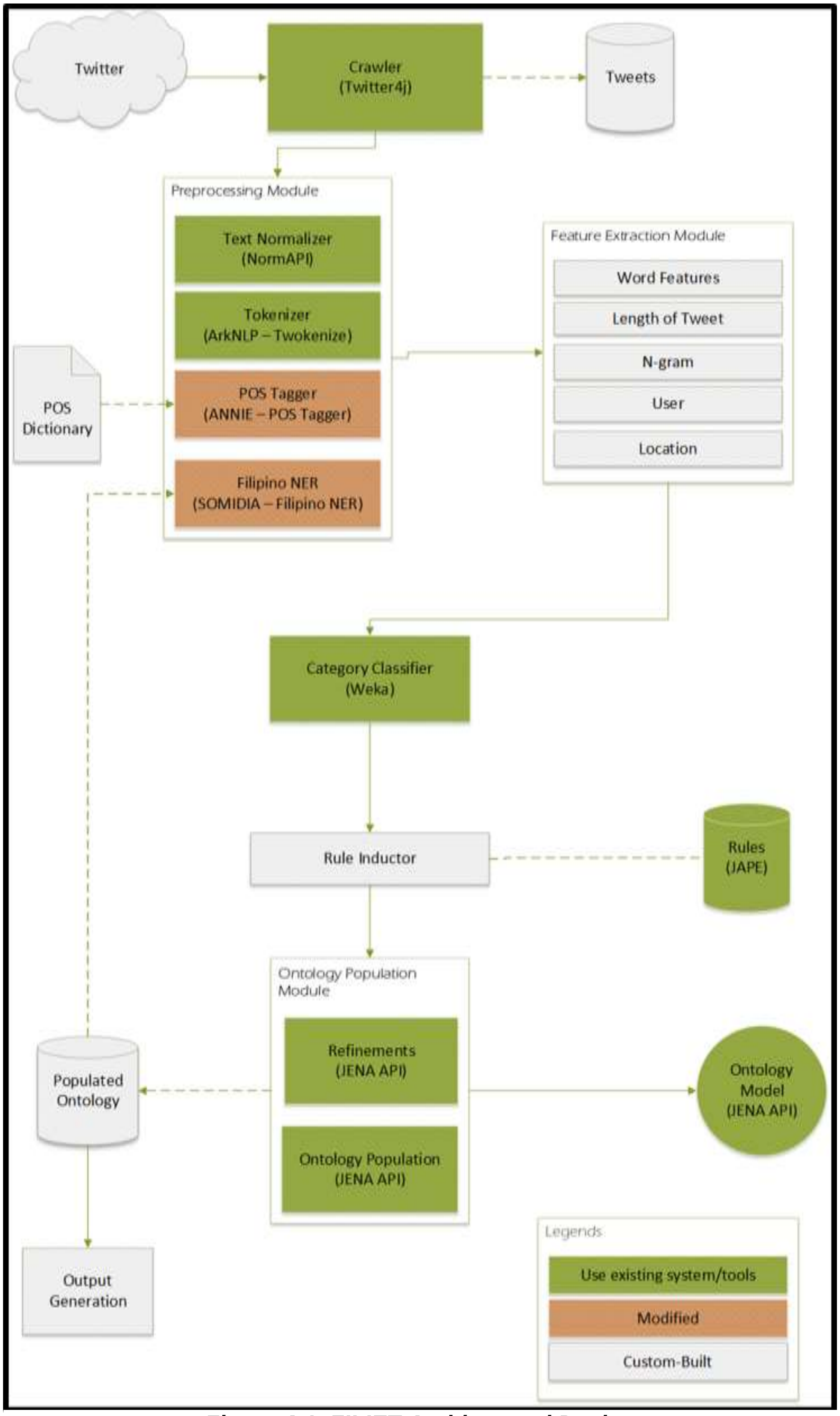

Figure 4-1. FILIET Architectural Design

4.4.1 Crawler Module

This module will be crawling Twitter to retrieve tweets. The system will continuously collect the tweets using Twitter's Stream API through the Twitter4j library.

4.4.2 Preprocessing Module

This module will be responsible for preprocessing the input tweets before they are passed on to the information extraction module. This module will include the following text processing techniques: text normalizer, tokenizer, and POS Tagger. After going through this module, the preprocess tweets will then be passed on to the Information extraction module.

4.4.2.1 Text Normalizer

The first step in preprocessing the input tweets is text normalization. The main responsibilities of the text normalizer are (1) to convert the TXTSPK format of the tweets into full-word format so that the information when extracted will be consistent and (2) remove emoticons, links, and hashtags. The text normalizer will accept a text as input. The output of this module is the normalized tweets where the TXTSPK is converted to its full form, and links and emoticons are removed. For this module, the researchers will use NormAPI (Nocon et al., 2014). Table 4-1 shows a sample input and its corresponding output.

Table 4-1. Sample Input/Output for Text Normalizer

Input	Output

`<tweet>`
`Dear Adnu sana po damit naman ang idonate natin para sa mga binagyo in case na may donation na ganapin. Plus canned goods na rin. Haha. :)`
`</tweet>`
	`<tweet>`
	`Dear Adnu sana po damit naman ang idonate natin para sa mga binagyo in case na may donation na ganapin. Plus canned goods na rin. Haha.`
	`</tweet>`

`<tweet>`
`Kailangan na talaga ng military efforts sa most part of Leyte. Nagkakagulo na.`
`</tweet>`
	`<tweet>`
	`Kailangan na talaga ng military efforts sa most part of Leyte. Nagkakagulo na.`
	`</tweet>`

4.4.2.2 Tokenizer

After normalizing the tweets, the tokenizer will now split the input tweets into tokens like numbers, punctuations, words, abbreviations and other special characters like emoticons, hashtags, mentions, and the like. The tokenizer will take as an input the normalized tweet from the Text Normalizer. The tokenizer will output an array containing the tokenized tweet in a form that is similar to this. Tokenized = {"@<username>", "<punctuations>", "#<hashtag>"...} or an array that would contain all the tokens in a given tweet. For this module, the researchers will use ArkNLP's Twokenize (Gimpel et al., 2011). Table 4-2 shows a sample input and its corresponding output.

Table 4-2. Sample Input/Output Tokenizer

Input	Output

`<tweet>`
`Dear Adnu sana po damit naman ang idonate natin para sa mga binagyo in case na may donation na ganapin. Plus canned goods na rin. Haha.`
`</tweet>`
	`<tweet>`
	`"Dear", "Adnu", "sana", "po", "damit", "naman", "ang", "idonate", "natin", "para", "sa", "mga", "binagyo", "in", "case", "na", "may", "donation", "na", "ganapin", ".",`

	"Plus", "canned", "goods", "na", "rin", ".", "Haha", "." </tweet>
<tweet> Kailangan na talaga ng military efforts sa most part of Leyte. Nagkakagulo na. </tweet>	<tweet> "Kailangan", "na", "talaga", "ng", "military", "efforts", "sa", "most", "part", "of", "Leyte", ".", "Nagkakagulo", "na", "." </tweet>

4.4.2.3 POS Tagger

After tokenizing the tweets, the POS tagger will accept the tokenized Filipino tweet as an input and then, it will tag each token with its corresponding part-of-speech. Each of the tokens can be tagged as a noun, a verb, an adjective, an adverb or others. After tagging the tokens, the POS tagger will then output the tokens with their corresponding POS tag in the form of a text. For the module, the researchers are considering modifying ANNIE's POS Tagger (Cunningham et al, 2002) for Filipino, or use Filipino Tagger Dictionary (Oco & Borra, 2011). Table 4-3 shows the sample input and output of POS tagger.

Table 4-3. Sample Input/Output POS Tagger

Input	Output
<tweet> "Dear", "Adnu", "sana", "po", "damit", "naman", "ang", "idonate", "natin", "para", "sa", "mga", "binagyo", "in", "case", "na", "may", "donation", "na", "ganapin", ".", "Plus", "canned", "goods", "na", "rin", ".", "Haha", "." </tweet>	<tweet> "Dear_UH", "Adnu", "sana_VOTF", "po_MAHM", "damit_NCOM", "naman_ENCL", "ang_NA", "idonate", "natin_PNGP", "para_PRTA", "sa_NCOM", "mga_NA", "binagyo", "in_IN", "case_VBP", "na_NA", "may_MAEM", "donation_NN:UN", "na_NA", "ganapin", "._PSNS", "Plus_JJ", "canned_JJ", "goods_NNS", "na_NA", "rin_ENCL", "._PSNS", "Haha_NN", "._PSNS" </tweet>
<tweet> "Kailangan", "na", "talaga", "ng", "military", "efforts", "sa", "most", "part", "of", "Leyte", ".", "Nagkakagulo", "na", "." </tweet>	<tweet> "Kailangan_VOTF", "na_NA", "talaga_IRIA", "ng_NA", "military_NCOM", "efforts_NNS", "sa_NCOM", "most_JJS", "part_JJ", "of_IN", "Leyte_NPRO", "._PSNS", "Nagkakagulo", "na_NA", "._PSNS" </tweet>

4.4.2.4 Filipino NER

The Filipino NER will identify the proper nouns in the tweets. The module will accept the tweets that have passed through the preprocessing module. The outputs of the NER are tagged as proper nouns in the tweet. For the gazetteer, the plan is to use the SOMIDIA gazetteer and update the gazetteer. Table 4-4 shows a sample input and its corresponding output.

Table 4-4. Sample Input/Output Gazetteer

Input	Output
<tweet> "Dear_UH", "Adnu", "sana_VOTF", "po_MAHM", "damit_NCOM",	<tweet> "Dear_UH", "Adnu", "sana_VOTF", "po_MAHM", "damit_NCOM",

"naman_ENCL", "ang_NA", "idonate", "natin_PNGP", "para_PRTA", "sa_NCOM", "mga_NA", "binagyo", "in_IN", "case_VBP", "na_NA", "may", "donation_NN:UN", "na_NA", "ganapin", "._PSNS", "Plus_JJ", "canned_JJ", "goods_NNS", "na_NA", "rin_ENCL", "._PSNS", "Haha_NN", "._PSNS" </tweet>	"naman_ENCL", "ang_NA", "idonate", "natin_PNGP", "para_PRTA", "sa_NCOM", "mga_NA", "binagyo", "in_IN", "case_VBP", "na_NA", "may", "donation_NN:UN", "na_NA", "ganapin", "._PSNS", "Plus_JJ", "canned_JJ", "goods_NNS", "na_NA", "rin_ENCL", "._PSNS", "Haha_NN", "._PSNS" </tweet>
<tweet> "Kailangan_VOTF", "na_NA", "talaga_IRIA", "ng_NA", "military_NCOM", "efforts_NNS", "sa_NCOM", "most_JJS", "part_JJ", "of_IN", "Leyte_NPRO", "._PSNS", "Nagkakagulo", "na_NA", "._PSNS" </tweet>	<tweet> "Kailangan_VOTF", "na_NA", "talaga_IRIA", "ng_NA", "military_NCOM", "efforts_NNS", "sa_NCOM", "most_JJS", "part_JJ", "of_IN", "<location: Leyte/>", "._PSNS", "Nagkakagulo", "na_NA" "._PSNS" </tweet>

4.4.3 Feature Extraction Module

This module is responsible for extracting the feature from the tweet. The module will extract the presence of disaster words, tweet length, character n-gram, user, location, and trusted accounts. The Feature Extraction Module will take the preprocessed tweets as inputs, then output the tweet with the features. Table 4-13 shows a sample of the features and their respective values.

4.4.3.1 Presence

The Presence feature is a binary feature that indicates the presence of keywords like disaster words, mentions, hashtags, emoticons, retweets, and Code Switching in the input tweet. The value of “1” is given if the keyword is present; otherwise it is given “0”.

4.4.3.2 Tweet Length

The Tweet Length feature essentially counts the length of the input tweet.

4.4.3.3 N-gram

The N-gram feature is mainly responsible for generating/extracting the different n-grams for the input tweets, specifically, the bi-gram and the tri-gram of the input tweets. To accomplish the n-gram generation/extraction tasks, the module will make use of the SRILM tool, which is specifically built for generating/extracting n-gram models.

4.4.3.4 User

The User feature will help in determining the type of disaster. For example, @dost_pagasa will tweet about typhoons.

4.4.3.5 Location

The location feature is where the disaster occurred. There are instances which are specific to certain disasters, for example, the disaster is flood, and the location given is usually a street. It can be also be a region, city or province for typhoon- or earthquake- related tweets.

4.4.4 Category Classifier Module

Using the extracted features, the Category Classifier Module will classify the tweets into the following categories: (1) caution and advice (CA), (2) casualties and damage (CD), (3) donations (D), (4) call for help (CH), and (5) others (O). The module will use Weka (Weka, n.d.) and will try out different classifiers. Table 4-5 shows a sample input/output of the Category Classifier Module.

Table 4-5. Sample Input/Output Category Classifier Module

Input	Output
<tweet> "Dear_UH", "Adnu", "sana_VOTF", "po_MAHM", "damit_NCOM", "naman_ENCL", "ang_NA", "idonate", "natin_PNGP", "para_PRTA", "sa_NCOM", "mga_NA", "binagyo", "in_IN", "case_VBP", "na_NA", "may", "donation_NN:UN", "na_NA", "ganapin", "._PSNS", "Plus_JJ", "canned_JJ", "goods_NNS", "na_NA", "rin_ENCL", "._PSNS", "Haha_NN", "._PSNS" </tweet>	<tweet type="D"> "Dear_UH", "Adnu", "sana_VOTF", "po_MAHM", "damit_NCOM", "naman_ENCL", "ang_NA", "idonate", "natin_PNGP", "para_PRTA", "sa_NCOM", "mga_NA", "binagyo", "in_IN", "case_VBP", "na_NA", "may", "donation_NN:UN", "na_NA", "ganapin", "._PSNS", "Plus_JJ", "canned_JJ", "goods_NNS", "na_NA", "rin_ENCL", "._PSNS", "Haha_NN", "._PSNS" </tweet>
<tweet> "Kailangan_VOTF", "na_NA", "talaga_IRIA", "ng_NA", "military_NCOM", "efforts_NNS", "sa_NCOM", "most_JJS", "part_JJ", "of_IN", "Leyte_NPRO", "._PSNS", "Nagkakagulo", "na_NA", "._PSNS" </tweet>	<tweet type="D"> "Kailangan_VOTF", "na_NA", "talaga_IRIA", "ng_NA", "military_NCOM", "efforts_NNS", "sa_NCOM", "most_JJS", "part_JJ", "of_IN", "<location: Leyte/>", "._PSNS", "Nagkakagulo", "na_NA" "._PSNS" </tweet>

4.4.5 Rule Inductor Module

The Rule Inductor module will accept tokenized and tagged tweets. It will now apply the rules coming from the database. It will look for patterns in the text and apply the classification. It will generate the instances that will be used to populate the ontology.

4.4.6 Ontology Population Module

The ontology population module is responsible for filling up the ontology with extracted information from the previous module instances. It has two sub-modules: Ontology Population and Ontology Retrieval. Both submodules takes advantage of the OWL API to manipulate and modify the contents of the ontology. The structure of the ontology that will be used for this module and this system will be made using the Protégé ontology tool.

4.4.6.1 Ontology Population

The Ontology Population module will be responsible for storing the extracted information to the ontology by filling up the fields and asserting the relations that exists within the ontology. This sub-module would include pre-requisite functions to facilitate a seamless exchange pf information between the system and the ontology like saving, loading, and etc. Due to the nature of ontologies, there is no need to check for duplicate instances within the ontology as the ontology will do the checking and validating for this kind of scenario.

4.4.6.2 Ontology Retrieval

The Ontology Retrieval module will be responsible for retrieving the information that was stored in the ontology by getting all the instances of ontological classes and assertions that exists within the ontology. This sub-module would include pre-requisite functions to facilitate a seamless retrieval of stored information between the system and the ontology like saving, loading, and etc. Due to the structure of the ontology designed for the system, the main point of contact for the retrieval process is the instances of the Tweet class of the ontology. Further retrieval of information will be based on the relations that different instances from different classes have with the instances of the Tweet class.

4.4.7 Data Sources

The data that will be collected will come from the filtered tweets. Some of these will be provided by the Twitter Web Crawler developed by the De La Salle – College of Computer Studied, while the rest will come from the Crawler module to be discussed in the next section. The list of trusted Twitter accounts is based on the list provided by SOMIDIA.

To be able to crawl the tweets that are strictly related to disaster relief operations, the researchers will make use of certain national official hashtags that are used by a number of relief organizations in the country. Examples of the unified hashtags are #ReliefPH, #RescuePH, #PHalert

The output of the crawler will be saved in a CSV file. Each entry in the CSV file will have the following content: `<tweet ID>,<username>,”<tweet>”,”<date and time it was tweeted>”,<longitude>,<latitude>`. **Error! Reference source not found.** Table 4-6 shows a sample of what can be seen in the CSV file.

Table 4-6. Sample Entries of Tweets in CSV File

#	Sample Output
1	`5280d16567833c59e17ebb66, SandyCervas, Dear Adnu sana po damit naman ang idonate natin para sa mga binagyo in case na may donation na ganapin. Plus canned goods na rin. Haha. :) , 11/11/2013 8:45, 13.7053384, 123.1980436`
2	`414017377517326337,Ehmai123,"""@ANCALERTS: Magnitude 4.3 quake jolts Antique, Boracay http://t.co/c2BczJEa6Y"" Lindol everywhere :3","Fri Dec 20 21:00:09 CST 2013",14.527157,121.0033549`

4.4.7.1 Gazetteer

The gazetteer is a text file that contains the list of names and locations to identify the proper nouns in the tweets. This will be used for the Filipino NER module. The plan is to update and use SOMIDIA’s gazetteer. Table 4-7**Error! Reference source not found.** shows a sample gazetteer for the storm names in the Philippines.

Table 4-7. Sample Gazetteer for Storm Names (Philippines)

Agaton	Falcon	Kabayan	Pablo	Udang
Amang	Feria	Karen	Paeng	Unding
Ambo	Florita	Katring	Pedning	Ursula
Auring	Frank	Kiko	Pepeng	Usman
Basyang	Gener	Labuyo	Quedan	Venus
Bebeng	Gloria	Lando	Queenie	Vinta
Bising	Goring	Lawin	Quiel	Violeta
Butchoy	Gorio	Luis	Quinta	Viring
Caloy	Hanna	Marce	Ramil	Waldo
Chedeng	Helen	Maring	Ramon	Weng

Cosme Crising Dante Dindo Dodong Domeng Egay Emong Enteng Ester	Henry Huaning Igme Inday Ineng Isang Jolina Juan Juaning Julian	Milenyo Mina Nando Neneng Nina Nonoy Ofel Ompong Ondoy Onyok	Reming Rolly Santi Seiang Sendong Siony Tino Tisoy Tomas Tonyo	Wilma Winnie Yayang Yolanda Yoyong Yoyoy Zeny Zigzag Zoraida Zosimo

4.4.7.2 Rules

Based on the tweets, the rules will be handcrafted using JAPE. Then, the rules will now be stored in the database which will be used for extracting the information. Table 4-8 shows a sample of the rules.

Table 4-8. Sample Extracted Rules

Rules
`<string: naman><disaster><string:sa> AS Disaster`
`<string: magnitude><number>AS Intensity`
`<POS: NNS><location><POS: PSNS>AS Location`

4.4.7.3 Seed Words

The seed words will be used for generating the rules. The list of seed words will be stored in a text file. It will SOMIDIA's seed word and will update it. Table 4-9 shows the excerpts of the list of seed words.

Table 4-9. Excerpts of the List of Seed Words

tubig kuryente pagkain tulong donation damit gutom	rice kanin bigas inumin sardinas sardines canned goods	water clothes food help bahay gamot medicine	health kit medical kit relief goods kasuotan instant noodles damit pera

4.4.7.4 POS Dictionary

The POS Dictionary is a dictionary that contains a list of words with its POS tag. This will be used in the POS Lookup. The dictionary is stored in a file. It contains a list of English and Filipino words. Table 4-10 shows a sample of the excerpts of the POS dictionary

Table 4-10. Excerpts of the POS Dictionary

storms storm ENG NNS	buko buko TAG NCOM 2
storms storm ENG VBZ	bula bula TAG NCOM 2
storm storm ENG NN	bulag bulag TAG NCOM 2
storm storm ENG VB	bulak bulak TAG NCOM 2
bukid bukid TAG NCOM 2	bulalas bulalas TAG NCOM 2

4.4.7.5 Ontology

For the ontology, this will be created manually. The domain of the ontology will be disaster, specifically for relief operations. The next step would be identification of the terms. After identifying the terms, the concept, properties, and constraints will be defined. Class instantiations then follow. The format of the ontology will be in OWL. **Error! Reference source not found.** shows the ontology of the system.

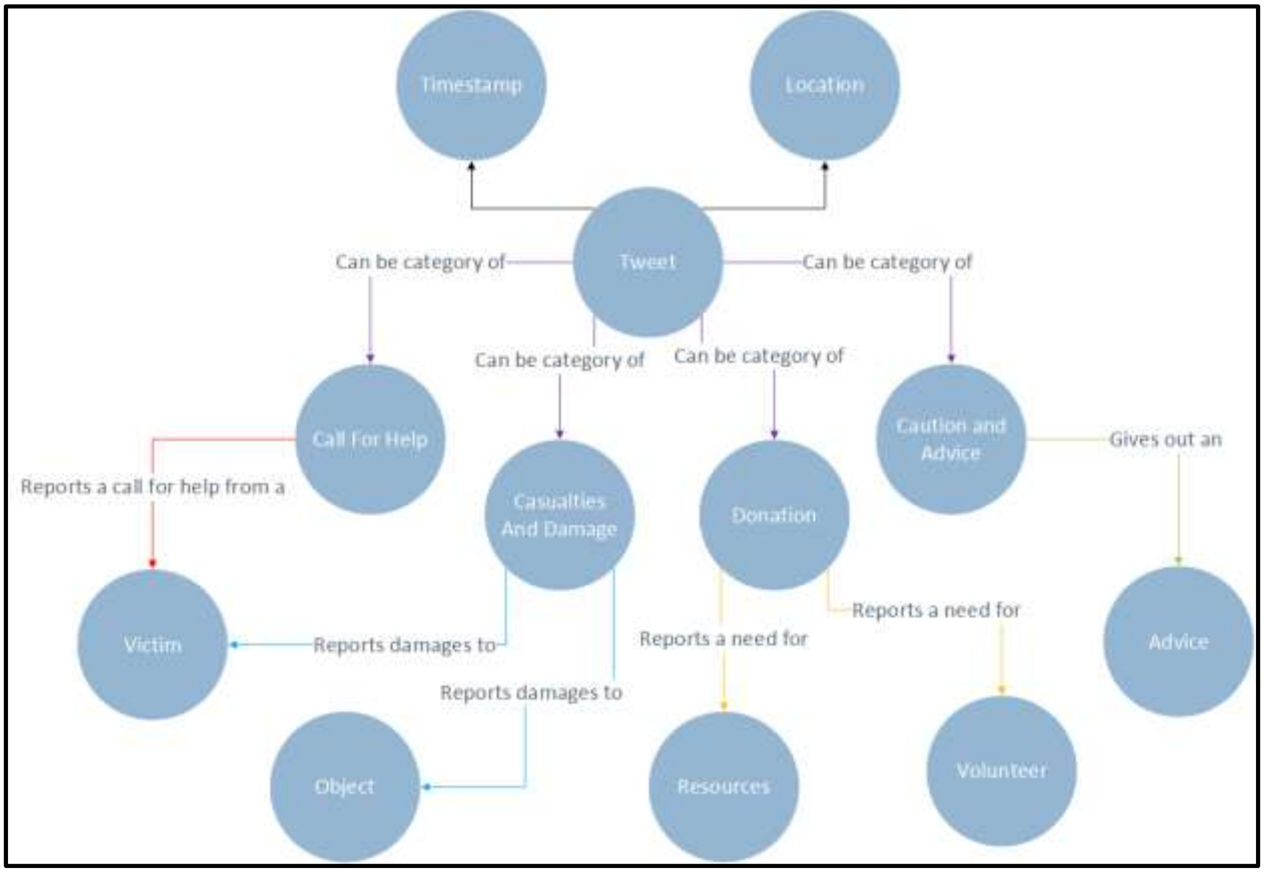

Figure 4-2. FILIET Ontology

4.5 System Functions

This section discusses the functions of the proposed systems.

4.5.1 Tweet Retrieval

Figure 4-3. Tweet Retrieval Screenshot

In this function, the system will access the tweets that were stored in the database by the Twitter crawler. The user can opt to filter the tweets for retrieval. Figure 4-3 shows the screenshot of this function.

4.5.2 Information Extraction

In this system function, the information extraction process starts with feature extraction which shall then be used for the classification of the tweets based on the categories defined in the system. After classification, the tweets shall then be examined for possible rules. The rules to be generated will then be applied to the tweets. Extracted information will be fed into the next function. Figure 4-4 shows a screenshot of this function.

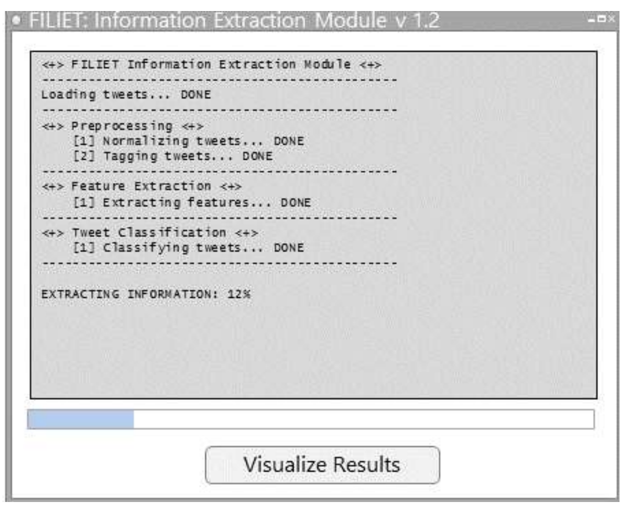

Figure 4-4. Information Extraction Screenshot

4.5.3 Ontology Population

Figure 4-5. Ontology Population Screenshot

In this system function, the extracted information will be initialized as entity instances for population of the ontology. The system, by default, automatically validates each of the extracted information before being introduced to the ontology. During validation, the system will check if the entity instances exist and if they do, the system will match the instances to their corresponding entity class/es. If they do not, the instances will immediately be discarded. Figure 4-5 shows the screenshot of this function.

4.5.4 Ontology Retrieval

In this system function, the populated ontology and the details of the instances per entity class can also be viewed. Relationships within entities can be seen or searched given a selected instance from the ontology model. Figure 4-6 shows a screenshot of this function.

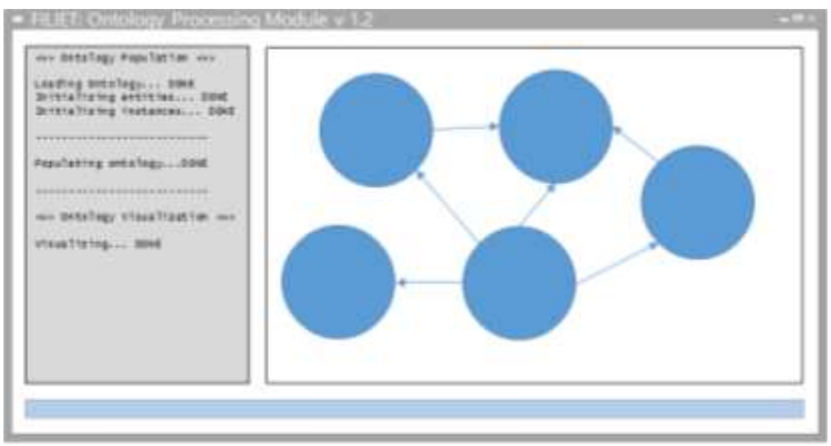

Figure 4-6. Ontology Access Screenshot

4.6 Physical Environment and Resources

This section outlines the minimum software and hardware requirements of the system.

4.6.1 Minimum Software Requirements

- Windows 7

- MySQL
- Java 1.7.0

4.6.2 Minimum Hardware Requirements

- 2 GB RAM
- Server

5.0 Design and Implementation Issues

This chapter discusses the design and implementation of the system as well as the issues encountered during its development.

5.1 Resource Gathering

5.1.1 Dataset Building

There are two datasets that are used. The first dataset is composed of combined Mario and Ruby tweets, it is categorized into three categories, Caution and Advice (CA), Casualty and Damage (CD) and Donation (D). There are 567 CA, 72 CD, 45 D and 681 other tweets in the first dataset. The second dataset is composed of purely Ruby tweets, it is categorized into four categories, Caution and Advice (CA), Casualty and Damage (CD), Call for Help (CH) and Donation (D). There are 1000 CA, 202 CD, 63 CH, 43 D, and 999 other tweets in the second dataset. Both of the datasets are manually tagged.

5.1.2 Feature Words

Feature Words are words that are used for the classification module. In order to get the list of feature words, a large number of tweets is needed. The dataset contained 1028 instances. The instances is then manually classified them into the categories. Then, they were separated into files. After that, the TFIDF scores were computed for each of the word in the dataset. The top 100 scoring for each dataset is used as the feature words. If there are words that appeared in other dataset, that word is compiled into one.

5.2 Crawler Module

The crawler module is responsible for the collection of disaster-related tweets. It uses Twitter4j API (Twitter4j, n.d.), an unofficial Java library that uses Twitter API, to crawl Twitter. This module uses an authenticated account that follows users. In order to get disaster-related tweets, the official hashtags used by the government were used to filter the tweets (please refer to Appendix C: for the list of hashtags used). The module uses Twitter's User Stream API to continuously listen for new tweets. Each tweet is then stored into the database.

Code Listing 5-1. Crawl for tweets with the specified keywords

```
// Filters
FilterQuery fq = new FilterQuery();
String keywords[] = {"#reliefPH","#nopower", "#nowater",
     "#roadalert", "#tracingPH", "#rescuePH", "#floodPH",
     "#queenieph"};
fq.track(keywords);

TwitterStream tweetStream = twitterStreamFactory.getInstance();
tweetStream.addListener(listener);
tweetStream.filter(fq);
```

Twitter4j retrieves the tweets the moment a tweet is updated in the account's timeline. The listener is handled by the `UserStreamListener`, specifically the `onStatus()` method. The method will first receive a `Status` object. This contains all information regarding the tweet. Then, the following information is then processed: `TweetID`, `User`, `Tweet`, `Latitude`, `Longitude`, `Language`, `IsUrl`, `IsHashtag`, and `IsRetweet` to a `Tweet` object. After binding the `Status` object to the `Tweet` object, the `Tweet` object handles the storing of the tweet information to the database.

Code Listing 5-2. Store the Twitter status to the database

```
Tweet tweet = new Tweet(status);
try {
     tweet.StoreTweet();
} catch (SQLException e) {
     e.printStackTrace();
}
```

There were several issues encountered in the crawler module. In the first implementation of the crawler module, it was ran to collect tweets during the Typhoon Mario. The some of the tweets that were collected were in multi-line format. The collected tweets are then manually cleaned in order to remove the line breaks. Another problem encountered was the amount of irrelevant tweets that the crawler is collected. Even with using the official hashtags as a filter, the crawler is still getting numerous irrelevant tweets because people use the hashtags irresponsibly.

5.3 Preprocessing Module

5.3.1 Text Normalizer

The first task to do in the Preprocessing Module is to clean the tweets by normalizing the text the tweet contains. To accomplish this task, the NormAPI library was used. Using the API is easy to use because only the `normalize_Text(String)` method is needed.

Code Listing 5-3. Using NormAPI

```
String normalizedText = NormAPI.normalize_Text(shortcutText);
```

The normalization starts with the initialization of the `Normalizer`. The `Normalizer` needs a class that has implemented a `NormalizerInterface`.

Code Listing 5-4. NormAPI Approach

```
normalizer = new Normalizer(new NormApiImpl());
```

To use the `Normalizer`, invoke the `executeStrategy(String)`. The executeStrategy`()` will expect a `String` object that contains the actual tweet. The `Normalizer` will return a `String` object that is now normalized.

Code Listing 5-5. FILIET Normalizer Execution

```
normalizedTweet = normalizer.executeStrategy(text);
```

The only issue in using NormAPI to set up all the necessary pre-requisite tools and packages that NormAPI needs. There came a point in time where all the pre-requisites were installed and properly working but NormAPI still would not work. With the help of one of the API's proponents, the cause of the error was found out to be that the copy of NormAPI.jar included in the installation files was the wrong one.

5.3.2 Tokenizer

This is the second step in processing the incoming tweets that will be coming from web crawler or a CSV file. The tokenizer takes a tweet and parses them into tokens. It returns a Sentence object that contains the array list of Tokens.

The Tokenizer module has two implementations: ArkNLP (Gimpel et al., 2011) and OpenNLP (Apache Software Foundation, 2010). However, other implementations could be easily added by implementing the `TokenizerInterface`. The `TokenizerInterface` returns a Sentence object that contains the tweets.

The tokenizer from ArkNLP is called using the static method `tokenizeRawTweetText(String text)` from the ArkNLP Library. It will then return the tokenized tweet through an array list of string.

Code Listing 5-6. ArkNLP Tokenizer Approach

```
List<String> tokens = Twokenize.tokenizeRawTweetText(text);
```

For the OpenNLP tokenizer, a `TokenizerModel` is needed to initialize the `TokenizerME`.

Code Listing 5-7. OpenNLP Tokenizer Approach

```
List<String> tokens = Twokenize.tokenizeRawTweetText(text);
```

To use the tokenizer in FILIET, a `Tokenizer` must be initialized first. The `Tokenizer` is accepting a class that implemented the `TokenizerInterface`, which is either the ArkNLP or OpenNLP approach. This is done so that tokenizer could be easily modified in the future.

Code Listing 5-8. Tokenizer Implementations in FILIET

```
// ArkNLP implementation
Tokenizer tokenizer = new Tokenizer(new ArkNLPTokenizerImpl());

// OpenNLP implementation
Tokenizer tokenizer = new Tokenizer(new OpenNLPTokenizerImpl());
```

Both implementations implement the `TokenizerInterface` wherein they will only call the `tokenize()` method to tokenize the tweet. This returns an array of strings where each element is a token of the tweet.

Code Listing 5-9. OpenNLP Tokenizer Approach

```
tokens = tokenizer.tokenize(text);
```

To execute the tokenizer in FILIET, call the `executeStrategy(String)`.

Code Listing 5-10. FILIET Tokenizer Execution

```
tokens = tokenizer.executeStrategy(normalizedTweet);
```

The issue of implementing the OpenNLP tokenizer is that it needs a model. There are currently no Filipino model that could be used for the tokenizer, so the group used an English model. Another issue in OpenNLP is tokenizing emoticons. It splits them into two tokens, when it should only be one. In comparison to ArkNLP tokenizer, ArkNLP is very simple to use as it only needs to call a static method from the library. It is much more adept to tweets, because it can handle emoticons. Also, ArkNLP performs significantly faster than OpenNLP. Therefore, ArkNLP tokenizer was preferred.

5.3.3 POS Tagger

After tokenization, each of the tokens will be tagged with its corresponding parts-of-speech tag. The `POSTagger` will return the `Sentence` object that now contains the POS Tags. The `POSTagger` uses a lookup to tag each token.

The POS Tagger starts with the initialization of the `POSTagger`. The `POSTagger` needs a class that has implemented a `POSTaggerInterface`.

Code Listing 5-11. POS Dictionary Look-up Approach

```
POSTagger post = new POSTagger(new POSHashLookupImpl());
```

To use the POS Tagger, invoke the `executeStrategy(Sentence)`. The `executeStrategy()` will expect a `Sentence` object that contains the array list of tokens. The `POSTagger` will return a `Sentence` object that is now tagged with POS.

Code Listing 5-12. FILIET POS Tagger Execution

```
tokens = post.executeStrategy(tokens);
```

The main issue in implementing the POS Tagger is that there are no available tools for the Filipino Language. The look-up approach implemented uses a dictionary that contains the tags for each word. However, the words found in the dictionary is not suitable for tweets since it was extracted from the novels *Noli Me Tangere* and *El Filibusterismo*.

5.3.4 Filipino NER

Named Entity Recognition module accepts a `Sentence` object that contains the tokenized and POS tagged tweet. Using a dictionary, it tags the words that are locations. The NER module needs an object that implements the NER Interface. This object contains the actual implementation of the NER. The `SomidiaNERImpl` contains the implementation of a lookup NER. It uses the dictionary from SOMIDIA.

Code Listing 5-13. SOMIDIA's NER Approach

```
NamedEntityRecognizer ner = new NamedEntityRecognizer (new
     SomidiaNERImpl());
```

After the initialization, the `NamedEntityRecognizer` can now be used by calling the `executeStrategy(Sentence)`. The method returns the `Sentence` that is now tagged with NER.

Code Listing 5-14. FILIET Filipino NER Execution

```
tokens = ner.executeStrategy(tokens);
```

The main issue in this module is that it can only tag one-word locations. It only looks at one token at a time. The problem now arises from locations that contains two or more words because the current approach does not support them. Another problem facing the NER module is that there are locations that are considered as adjectives and vice versa. In the NER gazetteer, locations like “Maginhawa” and “Salamat” could be considered as other type. So, when the NER encountered a tweet like “Maraming salamat po”. The word “Salamat” will be tagged as a named entity when it should not be.

5.3.5 Preprocessor Manager

This module is responsible for initializing all of the sub-modules under the preprocessing module. The preprocessor manager accepts a `String` and then outputs a `Sentence` object that has been normalized, tokenized, POS tagged and NER tagged.

First, the preprocessor must be initialized. The preprocessor manager will then initialize the sub-modules. For the normalizer, it will initialize a normalizer using a `NormApiImpl` object. The tokenizer is initialized with `ArkNLPTokenizer` object. The POS Tagger is initialized with `POSHashLookupImpl`. Lastly, the NER Tagger is initialized with `SomidiaNERImpl`.

Code Listing 5-15. FILIET Preprocessor Manager Initialization

```
PreprocessorManager preprocess = new PreprocessorManager();
Sentence preprocessed = preprocess.PreprocessText(String);
```

5.4 Feature Extraction Module

The feature extractor module is for extracting word features and n-gram the features inside a tweet. The module requires word files that contains the word features. The feature extractor will then count how many times a word from the features appeared in the tweet. The feature extractor uses the top 30 highest count n-grams features for each category and top 100 TFIDF scores for the word features for each category, excluding stop words.

To use this, FeatureExtractor must first be initialized. It requires two parameters, the path to the n-gram files and the path to the word files.

Code Listing 5-16. FILIET Feature Extraction Initialization

```
String ngram = "./resources/model/ngram/ruby-ngram";
String word = "./resources/model/word/ruby-word";
FeatureExtractor fe = new FeatureExtractor(word,ngram);
```

After initializing, the `FeatureExtractor` can either do single processing or batch processing. For the single processing, the method `extract(Sentence)` is used. The method will be expecting a `Sentence` object that has been preprocessed. Then it will output a Sentence object that now contains extracted features. For the batch processing, the method `extract(String, String)` is used, where the method expects the path to a CSV file that contains the tweets, and the path where the output will be saved.

Code Listing 5-17. FILIET Feature Extraction Execution

```
// Single Processing
fe.extract(sentence);

// Batch Processing
String tweets = "./resources/tweets/mario-datasets/original/ruby-
     dataset.csv”;
String saveModel = "./resources/tweets/test-extracted/mario-
     tfidf/ruby-extracted.csv
fe.extractFeatures(tweets,saveModel);
```

5.5 Category Classifier Module

The classifier module categorizes the tweet into one of the following categories: Caution and Advice (CA), Casualties and Damage (CD), Donation (D), Call for Help (CH), and Other (O). The classifier uses a model in order to classify the tweets. Models are trained using Weka (Weka, n.d.). The classifier module accepts a `String` object that has passed through the Preprocessing Module and Feature Extractor Module and returns the category of the tweet.

First, the classifier must be initialized. The classifier accepts a class that has implemented a `ClassifierInterface`. The `ClassifierImpl` class has two constructors. If there is no provided path, the classifier will use the default model.

Code Listing 5-18. FILIET Classifier Implementations

```
// Default model
Classifier classifier = new Classifier(new ClassifierImpl());
// Path to the model resource.
Classifier classifier = new Classifier(new
     ClassifierImpl("./resources/model/classifier/testmodel.model"));
```

The classifier will then initialize a `ClassifierBuilder` class. This class is responsible for binding the `Sentence` object to an Instance object that will be used for the classification. The `ClassifierBuilder` has two constructors. The first constructor takes no parameters. This will set the word file to its default. The second constructor takes a parameter: path to the word file.

Code Listing 5-19. Classifier Builder Initialization

```
// Default model
ClassifierBuilder builder = new ClassifierBuilder();
// Path to the model resource.
ClassifierBuilder builder = new ClassifierBuilder(wordPath);
```

To run the classification, invoke the `executeStrategy(Sentence)` method.

Code Listing 5-20. FILIET Classifier Execution

```
String category = classifier.executeStrategy(temp);
```

5.6 Rule Inductor Module

Information extractor is the module responsible for extracting the relevant information from the tweets. The module accepts preprocessed and classified tweets and outputs the Sentence object with the extracted information. It uses the hand-crafted rules to extract the information.

To initialize the Rule Inductor module, the constructor accepts a single string parameter. This is the file path to the rule file.

Code Listing 5-21. Rule Inductor Initialization

```
RuleInductor ruleInductor = new RuleInductor(rulePath);
```

For the rule file, the rules are categorized into four categories. They are separated by `<Category>: [category]`. One rule is listed per line. Then, the `<end>` tag is used to signify the end of list for that `[category]`.

Code Listing 5-22. Sample Extraction Rules for CA Category

```
<Category>: CA
<pos:JJ> <pos:NN> <pos:PSNS> <number:ANY>
<ner:LOCATION>[as]LOCATION
<pos:JJ> <string:#1>
<pos:JJ> <string:#2>
<pos:JJ> <string:#3>
<pos:VBZ> <string:classes> <pos:IN> <pos:JJ> <pos:VBZ>
<pos:VBP> <string:classes> <pos:IN> <pos:JJ> <pos:VBZ>
<string:#walangpasok> <pos:JJ> <pos:VBZ>
<string:signal> <pos:NN> <pos:PSNS> <number:ANY>
<string:#walangpasok> <pos:PSNS> <string:klase>
<string:#walangpasok> <string:sa> <pos:PIDP> <pos:NA> <string:antas>
<end>
```

For the construction of extraction rules, each rule can consist of the following tags: `string`, `number`, `pos`, `ner`. The string tag will match to the token's word. The `pos` tag will match to the token's POS. The `number` tag is used to match numbers. Lastly, the `ner` will match to the token's NER. To use wildcards, the key "`ANY`" to match any values.

The Rule Inductor module uses `match(Sentence)` method to apply the rules.

Code Listing 5-23. Rule Inductor Initialization

```
ruleInductor.setExtractedInformation( ruleInductor.match(
     extractedTweet ) );
```

The implementation of the FILIET architecture itself presents the problem of propagation of errors that may arise from the different modules that precede it. Another problem is the over application of rules. Even if the tweets are already categorized, thus only applying these specific set of rules, the tweet can still be matched to various rules making the system extract extraneous information.

5.7 Ontology Population Module

After extracting the relevant information from the tweets based on their respective categories, they are now stored to an ontology that contains object relations between the different extracted information. The actual structure of the ontology was made using an external tool called Protegé that makes use of the OWL API. This module takes an instance or a list of instance of categorized tweet classes. The categorized tweet classes include the following: the `CallForHelpTweet` class for containing the information that were gathered under the Call For Help category; `CasualtiesAndDamageTweet` class for containing the information that were gathered under the Casualties and Damage category; `CautionAndAdviceTweet` class for containing the information that were gathered under the Caution and Advice category; lastly, `DonationTweet` class for containing the information that were gathered under the Casualties and Damage category. This module has two sub-parts that are both responsible for storing and accessing information in the ontology. The `OntologyModule` class is responsible for storing the extracted information to the ontology and the `OntologyRetriever` class is responsible for retrieving the information that was stored in the ontology.

5.7.1 OntologyModule

The `OntologyModule` class is the main class responsible for working around the storage and verification of the extracted information in the ontology that was designed for the use of the system. It has respective functions for the different pre-requisite steps that shall be taken before actually accessing and modifying the contents of the ontology. There are respective functions for loading, saving and removing the ontology from its manager. Also, general-purpose functions were included to streamline the process of verifying the information to the ontology. These functions include a categorized tweet information viewer and a data property value viewer.

To store information into the ontology, certain classes have to be initialized so that they can be manipulated within the module.

Code Listing 5-24. OntologyModule Initialization

```
// Classes for containing the extracted information per category
CallForHelpTweet oCH = new CallForHelpTweet();
CasualtiesAndDamageTweet oCD = new CasualtiesAndDamageTweet();
CautionAndAdviceTweet oCA = new CautionAndAdviceTweet();
DonationTweet oD = new DonationTweet();

// Class for actually initializing the module
OntologyModule oModule = new OntologyModule();
```

After initializing the necessary classes, the extracted information will now be stored into the new initialized classes based on its given category. For instance, let’s take the information that was extracted from a tweet that was categorized to be a Caution and Advice tweet. A sample code listing below shows how to temporarily store the extracted information into its respective categorized tweet class.

Code Listing 5-25. Sample Initialization of Categorized Tweet Information Instance

```
// SAMPLE INITIALIZATION FOR CAUTION AND ADVICE REPORTS
oCA.setTweetHandle("theonlykyleeeee");
oCA.setTweetContent(":(( RT WARNING! Baha sa Guadalupe!");
oCA.setTweetGeoLocation("10.00000121, 145.345300023");
oCA.setLocationInTweet("Guadalupe");
oCA.setTweetTimestamp("12/27/2014:00:13:67:40");
oCA.setTweetDate("December 27, 2014");
oCA.setTweetAdvice("WARNING! Baha sa Guadalupe!");
```

After temporarily storing the extracted information, actual storage of the information to the ontology now follows. Storing the extracted information is a fairly simple process; as it would only require calling one function and requires only one input. The catch, though, is that there are different methods for certain categories of tweet. Also, if you would want to verify if the storage process has been successful, you could do a rough view of the contents of the ontology just by calling a simple view method.

Code Listing 5-26. Storing Information in the Ontology

```
try {
     oModule.loadOntology();

     // Permanently store information to the ontology
     oModule.addCautionAndAdviceReport(oCA);
     oModule.addCasualtiesAndDamageReport(oCD);
     oModule.addDonationReport(oD);
     oModule.addCallForHelpReport(oCH);

     // View the contents of the ontology
     oModule.displayStoredTweets();

     oModule.removeOntologyFromManager();
} catch (OWLOntologyCreationException e) {
     e.printStackTrace();
}
```

In developing the `OntologyModule` class, there were some issues encountered with its actual implementation because of a number of reasons. First, the documentation that came with the API was not that comprehensive and streamlined in a way that there were actually no complete descriptions about the different methods that can be utilized; though, there were code samples that were confusing. With this type of documentation, there was difficulty in customizing the implementation of the different methods within the API to suit the needs and requirements of the system. Also, debugging/testing of the module was difficult because there was no complete reference for the actual functions of each method including its parameters and its outputs. Manipulation and modification of the actual ontology is a challenge because two different tools used. It is imperative to check if the changes are properly reflected in order for the module to function properly. If there are changes to the ontology, use Protegé first to fix the structure of the ontology and the different dependencies that might be affected upon performing the changes and then, modify the Java code so that there will be seamless interaction between the system and the ontology behind it.

5.7.2 OntologyRetriever

The `OntologyRetriever` class is the main class responsible for working around the retrieval of the extracted information in the ontology that was designed for the use of the system. It has respective functions for the different pre-requisite steps that shall be taken before actually accessing and modifying the contents of the ontology. There are respective functions for loading and removing the ontology from its manager. Also, general-purpose functions were included to streamline the process of retrieving the information to the ontology. These functions include a categorized tweet information retriever and a data property value retriever. Also, category-specific functions were included to properly organize the information that was retrieved. These functions include a constructor method for retrieved Caution and Advice, Casualties and Damage, Call for Help, and Donation tweets, which just make each of the set of extracted information an instance of each of the classes representing the mentioned categories.

To retrieve the information from the ontology, one class has to be initialized so that other modules can manipulate it.

Code Listing 5-27. OntologyRetriever Initialization

```
OntologyRetriever or = new OntologyRetriever();
```

After initializing the needed class, actual retrieval of the information from the ontology now follows. Retrieving the information stored in the ontology is a fairly simple process, as it would only require calling one function and requires no input. A variable of the type `RetrievedTweet` is needed to contain the information that will be returned by the retrieval function of the module. The variable should be of type `RetrievedTweet` because the retrieval function, logically, returns four `ArrayLists` for the different categories of the information.

Code Listing 5-28. RetrievedTweet Class Structure

```
public class RetrievedTweet {
     ArrayList<CallForHelpTweet> retrievedCFHTweets;
     ArrayList<CasualtiesAndDamageTweet> retrievedCADTweets;
     ArrayList<CautionAndAdviceTweet> retrievedCATweets;
     ArrayList<DonationTweet> retrievedDTweets;

     ......
}
```

Code Listing 5-29. OntologyRetriever Execution

```
try {
     or.loadOntology();
     RetrievedTweet rt = or.getStoredTweets();
     or.removeOntologyFromManager();
} catch (Exception e) {
     e.printStackTrace();
}
```

In developing the `OntologyRetriever` class, almost the same issues as the ones mentioned in the `OntologyModule` were encountered. First, the documentation that came with the API was not that comprehensive and streamlined. With this type of documentation, there is difficulty in customizing the implementation of the different methods that are concerned with ontology retrieval within the API to suit the needs and requirements of the system's retrieval method. Debugging/testing the module is hard because there is no complete reference for the actual output of each method including the effects of combining different methods to achieve a certain output from the ontology. Second, there have been some difficulties in structuring and implementing the actual retrieval method code because there is a need to look at the initial contents of the ontology and its actual relations and structures to be able translate the logical connections and relations between the instances stored in the ontology to a physical working code that will facilitate the exchange of information between the ontology and system. There should be a way to remember the specific the relational connections between the instances to and "simulate" them in order to reverse the process of adding/storing to the ontology; thus, essentially, enabling a retrieval process. Also, with this, essentially, the same difficulty of working with two separate tools was encountered.

6.0 Results and Observation

6.1 Classification

6.1.1 Word Feature Experiment

6.1.1.1 Mario Dataset

For Table 6-1, this uses the top 10% features for each category. The table shows that the Random Forest got the highest score with 0.84 F-measure and 0.7201 kappa statistics, while Naïve Bayes with 0.719 F-measure and 0.5141 kappa statistics.

Table 6-1. Using Top 10% Word Features for Mario Dataset

Algorithm	Precision	Recall	F-measure	Kappa
J48	0.812	0.813	0.807	0.6636
Random Forest	0.841	0.842	0.84	0.7201
kNN-3	0.792	0.791	0.786	0.6245
kNN-5	0.781	0.779	0.779	0.6001
kNN-7	0.77	0.768	0.759	0.5771
Naïve Bayes	0.731	0.725	0.719	0.5141
Bayesian Network	0.752	0.752	0.747	0.5553

For Table 6-2 this uses the top 20% features for each category. The table shows that Random Forest has the highest F-measure and kappa statistics, scoring 0.854 and 0.942 respectively, while Naïve Bayes scored the lowest with 0.726 F-measure and 0.5372 kappa statistics.

Table 6-2. Using Top 20% Word Features for Mario Dataset

Algorithm	Precision	Recall	F-measure	Kappa
J48	0.82	0.819	0.814	0.675
Random Forest	0.855	0.856	0.854	0.942
kNN-3	0.796	0.794	0.789	0.6293
kNN-5	0.784	0.782	0.776	0.606
kNN-7	0.78	0.78	0.773	0.6015
Naïve Bayes	0.751	0.73	0.726	0.5372
Bayesian Network	0.751	0.73	0.726	0.862

For Table 6-3, this uses the top 30% features for each category. The table shows that Random Forest has the highest F-measure and kappa statistics, scoring 0.86 and 0.7527 respectively, while Naïve Bayes scored the lowest with 0.734 F-measure and 0.5464 kappa statistics.

Table 6-3. Using Top 30% Word Features for Mario Dataset

Algorithm	Precision	Recall	F-measure	Kappa
J48	0.833	0.834	0.831	0.7043
Random Forest	0.86	0.862	0.757	0.7527
kNN-3	0.798	0.796	0.788	0.6293
kNN-5	0.794	0.793	0.786	0.6244
kNN-7	0.807	0.804	0.798	0.6451
Naïve Bayes	0.751	0.731	0.734	0.5464
Bayesian Network	0.739	0.733	0.73	0.5365

Based on the results, the random forest algorithm is consistently the highest among the classifiers in all three settings, while Naïve Bayes is consistently the lowest. All the algorithm with the exception of Bayesian Network increase in precision and recall as the number of features increase. The reason that the Bayesian Network is suffering in precision and recall is because of the increasing network. The increasing number of attributes introduces a lot of noise.

6.1.1.2 Ruby Dataset

For Table 6-4, this uses the top 10% features for each category. The table shows that the Random Forest got the highest score with 0.952 F-measure and 0.9219 kappa statistics, while kNN-7 with 0.908 F-measure and 0.8494 kappa statistics.

Table 6-4. Using Top 10% Word Features for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.924	0.91	0.914	0.8559
kNN-5	0.927	0.927	0.925	0.8793
kNN-7	0.908	0.909	0.905	0.8494
Random Forest	0.953	0.952	0.952	0.9219
J48	0.921	0.922	0.92	0.8721
Naïve Bayes	0.917	0.917	0.916	0.8649
Bayesian Network	0.931	0.93	0.928	0.8848

For Table 6-5, this uses the top 20% features for each category. The table shows that Random Forest has the highest F-measure and kappa statistics, scoring 0.966 and 0.9446 respectively, while kNN-7 scored the lowest with 0.895 F-measure and 0.8376 kappa statistics.

Table 6-5. Using Top 20% Word Features for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.936	0.934	0.931	0.8896
kNN-5	0.924	0.922	0.918	0.8695
kNN-7	0.908	0.904	0.895	0.8376
Random Forest	0.967	0.966	0.966	0.9446
J48	0.931	0.931	0.93	0.8875
Naïve Bayes	0.926	0.925	0.925	0.8782
Bayesian Network	0.931	0.93	0.928	0.8848

For Table 6-6, this uses the top 30% features for each category. The table shows that Random Forest has the highest F-measure and kappa statistics, scoring 0.964 and 0.943 respectively, while kNN-7 scored the lowest with 0.874 F-measure and 0.984 kappa statistics.

Table 6-6. Using Top 30% Word Features for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.934	0.931	0.928	0.8857
kNN-5	0.916	0.912	0.906	0.8521
kNN-7	0.892	0.887	0.874	0.984
Random Forest	0.966	0.965	0.964	0.943
J48	0.928	0.929	0.927	0.98
Naïve Bayes	0.932	0.929	0.929	0.8851
Bayesian Network	0.934	0.932	0.931	0.8891

Based on the results, Random Forest got the highest score in all the measure of the three settings. The 20% setting got the highest score in most of the measures, but there are measures that increased in the 30% setting.

6.1.2 Single Classifier

For the single classifier, the classifier must be able to identify the tweets into the four categories (CA, CD, CH, and D).

Table 6-7 shows the summary of results for the single classifier. It shows that the Random Forest algorithm got the highest score among all the classifiers. The kNN-7 classifier got the lowest score, and kNN-3 classifier got the highest in the kNN algorithm. The larger the neighbors the lower the score because it is introducing noise.

Table 6-7. Summary of Single Classifier Results for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.963	0.964	0.961	0.9399
kNN-5	0.954	0.955	0.952	0.9254
kNN-7	0.943	0.943	0.938	0.9054
Random Forest	0.978	0.978	0.977	0.9638
J48	0.969	0.967	0.967	0.9465

6.1.3 Multiple Binary Classifier

For the multiple binary classifier, each classifier will only classify two categories, either it is classified to the classifier's assigned category or it is not. If it is classified as not belonging to the category, it will cascade onto the next binary classifier until a category is chosen. If the tweet is not categorized at all, only then will it be classified as Others (O).

Table 6-8 shows the results of the CA binary classifier. Almost all the algorithms got a perfect score for all the measures but the highest scores are from Random Forest and J48, the two algorithms got the same scores.

Table 6-8. (CA) Binary Classifier Results for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.996	0.996	0.996	0.9921
kNN-5	0.996	0.996	0.996	0.9917
kNN-7	0.996	0.996	0.996	0.9917
Random Forest	0.999	0.999	0.999	0.9976
J48	0.999	0.999	0.999	0.9976

Table 6-9 lists the results of the CD binary classifier. The algorithms got a high score, Random Forest got a perfect score in all the measures. All the kNN classifiers got the same score in all the measures, even if the number of neighbors increased the score remained the same.

Table 6-9. (CD) Binary Classifier Results for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.983	0.983	0.983	0.9653
kNN-5	0.983	0.983	0.983	0.9653
kNN-7	0.983	0.983	0.983	0.9653
Random Forest	1	1	1	1
J48	0.99	0.99	0.99	0.9802

Table 6-10 shows the results of the CH binary classifier. Both the Random Forest and J48 got a perfect score in all the measures, the algorithms correctly classified all the instances.

Table 6-10. (CH) Binary Classifier Results for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.95	0.944	0.944	0.8889
kNN-5	0.932	0.921	0.92	0.8413
kNN-7	0.938	0.929	0.928	0.8571
Random Forest	1	1	1	1
J48	1	1	1	1

Table 6-11 lists the D binary classifier results. The Random Forest algorithm correctly classified all the instances, it got a perfect score in all the measures. The kNN classifiers did not reach 0.9 in any measure.

Table 6-11. (D) Binary Classifier Results for Ruby Dataset

Algorithm	Precision	Recall	F-measure	Kappa
kNN-3	0.891	0.86	0.858	0.7209
kNN-5	0.884	0.849	0.845	0.6977
kNN-7	0.877	0.837	0.833	0.6744
Random Forest	1	1	1	1
J48	0.989	0.988	0.988	0.9767

6.2 Information Extraction

Table 6-12 shows the results of the information extraction for the category CD. The Location got a low score for the measures because there are locations that are not a single word, these locations cannot be extracted by the system. The Object Name got a high score because it mostly consists of one word which is easily extracted by the system. While the Object Detail got a fair score because the system extracts any value that looks like it is part of the Object Name even if it is irrelevant, also the system is not able to extract spelled out values. Victim Name got a high score for the measures because most of the tweets do not have a victim name so the system extracts a null.

Table 6-12. Ruby Information Extraction Results (CD)

	Precision	Recall	F-Measure
Location	0.2396	0.3433	0.2822
Object Name	1.0	0.8529	0.9206
Object Detail	1.0	0.6485	0.7867
Victim Name	1.0	0.9901	0.9950

7.0 Conclusion and Recommendation

This chapter is divided into two sections. The first section presents the conclusion of the study and whether objectives of the research was achieved. The second section details the recommendations for possible areas of improvement per module.

7.1 Conclusion

The results shows that the random forest is the best algorithm for the classifier. The result for the random forest is consistent to the Mario and Ruby datasets. The features used is the top 30% features resulted the best for the classifier. For the information extraction, the module is having difficulty in extracting the location in the tweet because of the NER as the module could not completely extract the multi-words location. Also, some of the words tagged that are originally tagged to be a location entity could also be tagged to be an entity of a different type like an adjective, verb and the like. There will be problem extracting victim names as the NER could not identify people's name.

7.2 Recommendation

Listed below are some areas where improvements can be made for the system.

7.2.1 Preprocessing Module

- Inclusion of a lemmatizer sub-module so as to facilitate a cleaner dataset for subsequent modules. With Filipino being a morphologically rich language, words such as "bumabaha", "binabaha", and "binaha" all pertain to their root word which is "baha". With a lemmatizer, certain words will be tagged to their corresponding root word which will then simplify the job of the other modules.
- The current implementation of the POS Tagger makes use of a look-up dictionary extracted from a novels. Possible recommendations for this is to either improve upon the POS dictionary itself so that it can accommodate Twitter corpora or to create a stable POS Tagger for the Filipino language.
- Multi-worded named entities are not recognized because the system only processes on a one-token basis. Inclusion of a chunker sub-module will group related words which can help the next module identify multi-worded named entities (i.e. St. Ana, Tacloban City)
- Improve the Filipino NER dictionary in order to recognize the names of people and to be able to detect the named entities in hashtags (i.e. #cebucity, #taclobancity). By doing so, this may contribute to more accurate extraction of information.

7.2.2 Category Classifier Module

- Improve Categorization of tweets because there are ambiguous tweets due to ambiguity of the Filipino language, there are tweets that fall under two or more categories. There are also tweets that are uncategorized that does not fall under the existing categories.

7.2.3 Ontology Population Module

- Inclusion of a mechanism to be able to store multiple instances of classes that are found within the structures of the actual ontology. Ontological classes that could include multiple instances per instance of the root Tweet class are the Location class, Victim Class, and Object classes. As encountered in the different tweet instances that were processed by the system, there are actually tweets that had multiple locations, objects and victims related to the said instances. For instance, “Laguna, Cavite & Quezon” can be found in the location that can be extracted from the given tweet instance.

- Inclusion of more class specific fields for the ontology since, currently, the information that is stored for instances of different ontological classes seemed to be lacking. Like for example, the information that is stored for the Victim class is only limited to just the victim name. Further details can be added to this class like victim details and the like.
- Inclusion of a more powerful and streamlined visualization for the ontology. With the current implementation of the visualization of the ontology, the system does not take advantage of the powerful features of using ontology to store information. The current view is limited to just viewing the information stored in the ontology in table form and there is no concrete step for the users to search and manipulate its results within the ontology.
- Inclusion of a more generalized ontological population approach to facilitate a more open and non-strict way of storing information into the ontology. With the current implementation, the system provides a very linear way of storing information by starting with the root Tweet class instance until the related instances from other classes are all linked before actually storing them into the ontology. With a generalized ontological population approach, the system can be able to manipulate the information stored in a more customizable way by being able to separately store instances for different classes.

Appendix A: References

alias-i. (2011). What is lingpipe?. Retrieved from http://alias-i.com/lingpipe/

Aone, C., Halverson, L., Hampton, T., & Ramos-Santacruz, M. (1998, April). SRA: Description of the IE2 system used for MUC-7. In Proceedings of the seventh message understanding conference (MUC-7).

Apache Software Foundation. (2010). Welcome to apache opennlp. Retrieved from https://opennlp.apache.org/

Asahara, M., & Matsumoto, Y. (2003, May). Japanese named entity extraction with redundant morphological analysis. In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1 (pp. 8-15). Association for Computational Linguistics.

Aw, A., Zhang, M., Xiao, J., & Su, J. (2006, July). A phrase-based statistical model for SMS text normalization. In Proceedings of the COLING/ACL on Main conference poster sessions (pp. 33-40). Association for Computational Linguistics.

Basili, R., Moschitti, A., Pazienza, M. T., & Zanzotto, F. M. (2003). Personalizing web publishing via information extraction. IEEE Intelligent Systems, 18(1), 62-70.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. Journal of Computational Science, 2(1), 1-8.

Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M.A., Maynard, D., Aswani, N. TwitIE: An Open-Source Information Extraction Pipeline for Microblog Text. Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013).

Bradlee, D., Knoll, S., & Pentheroudakis, J. (2001). Tokenizer for a natural language processing system.

Califf, Mary Elaine. 1998. Relational Learning Techniques for Natural Language Information Extraction. Ph.D. thesis, Univ. of Texas at Austin, http://www.cs.utexas.edu/users/mecaliff

Cheng, H., Chua, J., Co, J., & Magpantay, A. B. (2013). Social media monitoring for disasters. Unpublished undergraduate thesis, De La Salle University, Manila, Philippines.

Cheng, T. T., Cua, J. L., Tan, M. D., Yao, K. G., & Roxas, R. E. (2009, October). Information extraction from legal documents. In Natural Language Processing, 2009. NLP'09. Eighth International Symposium on (pp. 157-162). IEEE.

Chew, C., & Eysenbach, G. (2010). Pandemics in the age of Twitter: content analysis of Tweets during the 2009 H1N1 outbreak. PloS one, 5(11), e14118.

Choy, M., Cheong, M., Laik, M. N., & Shung, K. P. (2012). US Presidential Election 2012 Prediction using Census Corrected Twitter Model. arXiv preprint arXiv:1211.0938.

Ciravegna, F., & Lavelli, A. (2004). LearningPinocchio: Adaptive information extraction for real world applications. Natural Language Engineering, 10(02), 145-165.

Corney, D., Byrne, E., Buxton, B., & Jones, D. (2008). A logical framework for template creation and information extraction. In Data Mining: Foundations and Practice (pp. 79-108). Springer Berlin Heidelberg.

Culnan, M. J., McHugh, P. J., & Zubillaga, J. I. (2010). How large US companies can use Twitter and other social media to gain business value. MIS Quarterly Executive, 9(4), 243-259.

Cunningham, H. (2002). GATE, a general architecture for text engineering. Computers and the Humanities, 36(2), 223-254.

Cunningham, H., Maynard, D., Bontcheva, K., &Tablan, V. (2002, July). A framework and graphical development environment for robust NLP tools and applications. In ACL (pp. 168-175).

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Ursu, C., Dimitrov, M. & Aswani, N. (2002). Developing language processing components with GATE (a user guide). University of Sheffield, Sheffield UK, 5.

Davis, J., & Goadrich, M. (2006, June). The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd international conference on Machine learning (pp. 233-240). ACM.

Dimitrov, M. (2005). A Lightweight Approach to Conference Resolution for Named Entities in Text Marin Dimitrov, Kalina Bontcheva, Hamish Cunningham and Diana Maynard. Anaphora Processing: Linguistic, cognitive and computational modelling, 263, 97.

Dung, T. Q., & Kameyama, W. (2007, March). A proposal of ontology-based health care information extraction system: Vnhies. In Research, Innovation and Vision for the Future, 2007 IEEE International Conference on (pp. 1-7). IEEE.

Faria, Carla, and Rosario Girardi. "An Information Extraction Process for Semi-automatic Ontology Population." Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO 2011. Springer Berlin Heidelberg, 2011.

Farkas, R. (2009). Machine learning techniques for applied information extraction (Doctoral dissertation, University of Szeged).

Feilmayr, C. (2011, August). Text Mining-Supported Information Extraction: An Extended Methodology for Developing Information Extraction Systems. In Database and Expert Systems Applications (DEXA), 2011 22nd International Workshop on (pp. 217-221). IEEE.

Freitag, D. (2000). Machine learning for information extraction in informal domains. Machine Learning, 39(2-3), 169-202.

Freitag, D., & Kushmerick, N. 2000. Boosted wrapper induction. In: Basili, R., Ciravegna, F., & Gaizauskas, R. (eds), ECAI2000 Workshop on Machine Learning for Information Extraction. http://www.dcs.shef.ac.uk/ fabio/ecai-workshop.html

Freitag, D., & McCallum, A. 1999. Information Extraction with HMMs and Shrinkage. In: AAAI-99 Workshop on Machine Learning for Information Extraction.

Gao, H., Barbier, G., & Goolsby, R. (2011). Harnessing the crowdsourcing power of social media for disaster relief. Intelligent Systems, IEEE, 26(3), 10-14. doi: 10.1109/MIS.2011.52

Ghedin, G. (2011, November 16). A social media lesson. from the Philippines. Retrieved from http://www.youngdigitallab.com/en/social-media/a-social-media-lesson-from-the-philippines

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., & Smith, N. A. (2011, June). Part-of-speech tagging for twitter: Annotation, features, and experiments. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2 (pp. 42-47). Association for Computational Linguistics.

Grishman, R. (1997). Information extraction: Techniques and challenges. In Information Extraction A Multidisciplinary Approach to an Emerging Information Technology (pp. 10-27). Springer Berlin Heidelberg.

Grosseck, G., & Holotescu, C. (2008, April). Can we use Twitter for educational activities. In 4th international scientific conference, eLearning and software for education, Bucharest, Romania.

Gruber, T. (2009). Ontology. Encyclopedia of database systems, 1963-1965.

Han, B., & Baldwin, T. (2011, June). Lexical normalisation of short text messages: Makn sens a# twitter. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 368-378). Association for Computational Linguistics.

Hawn, C. (2009). Take two aspirin and tweet me in the morning: how Twitter, Facebook, and other social media are reshaping health care. Health Affairs, 28(2), 361-368.

Horridge, M., & Bechhofer, S. (2011). The owl api: A java api for owl ontologies. Semantic Web, 2(1), 11-21.

Howard, B. (2013) . Scanning social media to improve typhoon Haiyan relief efforts. National Geographic Daily News. Retrieved from http://news.nationalgeographic.com/news/2013/11/131108-typhoon-haiyan-philippines-crisis-mapping/

Hripcsak, G., & Rothschild, A. S. (2005). Agreement, the f-measure, and reliability in information retrieval. Journal of the American Medical Informatics Association, 12(3), 296-298.

Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., & Meier, P. (2013, May). Practical extraction of disaster-relevant information from social media. In Proceedings of the 22nd international conference on World Wide Web companion (pp. 1021-1024). International World Wide Web Conferences Steering Committee.

Intarapaiboon, P., Nantajeewarawat, E., & Theeramunkong, T. (2009). Information extraction from Thai text with unknown phrase boundaries. In Advances in Knowledge Discovery and Data Mining (pp. 525-532). Springer Berlin Heidelberg.

Jansen, B. J., Zhang, M., Sobel, K., & Chowdury, A. (2009). Twitter power: Tweets as electronic word of mouth. Journal of the American Society for Information Science and Technology, 60(11), 2169-2188.

Junco, R., Heiberger, G., & Loken, E. (2011). The effect of Twitter on college student engagement and grades. Journal of Computer Assisted Learning, 27(2), 119-132.

King, D. (2005, April). Humanitarian knowledge management. In Proceedings of the Second International ISCRAM Conference (Vol. 1, pp. 1-6). Brussels.

Ko, H. (1998). Empirical assembly sequence planning: A multistrategy constructive learning approach. Machine Learning and Data Mining. John Wiley & Sons LTD.

Krishnamurthy, R., Li, Y., Raghavan, S., & Reiss, F. SystemT: A system for declarative information extraction. SIGMOD Record, 37. Retrieved May 28, 2014, from http://www.almaden.ibm.com/cs/projects/avatar/

Lee, J. B., Ybañez, M., De Leon, M. M., Estuar, M., & Regina, E. (2013). Understanding the Behavior of Filipino Twitter Users during Disaster. GSTF Journal on Computing, 3(2).

Lee, Y. S., & Geierhos, M. (2009). Business specific online information extraction from german websites. In Gelbukh, A. (Eds.), CICLing (pp. 369-381). Germany: Springer-Verlag Berlin Heidelberg.

Lim, N. R., New, J. C., Ngo, M. A., Sy, M., & Lim, N. R. (2007). A Named-Entity Recognizer for Filipino Texts. Proceedings of the 4th NNLPRS.

Loponen, A., & Järvelin, K. (2010). A dictionary-and corpus-independent statistical lemmatizer for information retrieval in low resource languages. In Multilingual and Multimodal Information Access Evaluation (pp. 3-14). Springer Berlin Heidelberg

Maedche, A., Neumann, G., & Staab, S. (2003). Bootstrapping an ontology-based information extraction system. In Intelligent exploration of the web (pp. 345-359). Physica-Verlag HD.

Malpica, A., Matisic, J. P., Niekirk, D. V., Crum, C. P., Staerkel, G. A., Yamal, J. M., ... & Follen, M. (2005). Kappa statistics to measure interrater and intrarater agreement for 1790 cervical biopsy specimens among twelve pathologists: qualitative histopathologic analysis and methodologic issues. Gynecologic Oncology, 99(3), S38-S52.

Manguilimotan, E., & Matsumoto, Y. (2009). Factors affecting part-of-speech tagging for Tagalog. In PACLIC (pp. 763-770).

Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to information retrieval (Vol. 1, p. 6). Cambridge: Cambridge University Press.

Maynard, D., Bontcheva, K., & Rout, D. (2012). Challenges in developing opinion mining tools for social media. Proceedings of@ NLP can u tag# user_generated_content.

Maynard, D., Peters, W., & Li, Y. (2006, May). Metrics for evaluation of ontology-based information extraction. In International world wide web conference.

McBride, B. (2002). Jena: A semantic web toolkit. IEEE Internet Computing, 6(6), 55-59.

McCallum, A. (2005). Information extraction: Distilling structured data from unstructured text. Queue, 3(9), 48-57.

Meier, P. (2013, September 18). [Web log message]. Retrieved from http://irevolution.net/2013/09/18/micromappers/

Mocanu, D., Baronchelli, A., Perra, N., Gonçalves, B., Zhang, Q., & Vespignani, A. (2013). The Twitter of Babel: Mapping world languages through microblogging platforms. PloS one, 8(4), e61981.

N´edellec C., Nazarenko A., & Bossy R. (2009). Information extraction. In S. Staab & R. Studer (Eds), Handbook on ontologies (pp 683-685). Dordrecht: Springer.

Nebhi, K. (2012). Ontology-based information extraction for French newspaper articles. In KI 2012: Advances in Artificial Intelligence (pp. 237-240). Springer Berlin Heidelberg.

Neubig, G., Matsubayashi, Y., Hagiwara, M., & Murakami, K. (2011, November). Safety Information Mining-What can NLP do in a disaster-. In IJCNLP (pp. 965-973).

Official Gazette of the Republic of the Philippines. (2012, July 21). Prepare for natural calamities: Information and resources from the government. Retrieved July 15, 2014, from http://www.gov.ph/crisis-response/government-information-during-natural-disasters/

OpenNLP, A. (2011). Apache Software Foundation. URL http://opennlp.apache.org.

Özsu, M. T., & Liu, L. (2009). Text Categorization. Encyclopedia of database systems (p. 3044). New York: Springer.

Pak, A., & Paroubek, P. (2010, May). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In LREC.

Pham, L. V., & Pham, S. B. (2012, August). Information Extraction for Vietnamese Real Estate Advertisements. In Knowledge and Systems Engineering (KSE), 2012 Fourth International Conference on (pp. 181-186). IEEE.

Phelan, O., McCarthy, K., & Smyth, B. (2009, October). Using twitter to recommend real-time topical news. In Proceedings of the third ACM conference on Recommender systems (pp. 385-388). ACM.

Poibeau, T. An Open Architecture for Multi-Domain Information Extraction. IAAI-01. Retrieved May 28, 2014, from www.aaai.org

Quinlan, J. R. (1990). Learning logical definitions from relations. Machine Learning, 5(3), 239-266.

Ritter, A., Clark, S., & Etzioni, O. (2011, July). Named entity recognition in tweets: an experimental study. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 1524-1534). Association for Computational Linguistics.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010, April). Earthquake shakes Twitter users: real-time event detection by social sensors. In Proceedings of the 19th international conference on World wide web (pp. 851-860). ACM.

Saloun, P., Velart, Z., & Klimanek, P. (2011, December). Semiautomatic domain model building from text-data. In Semantic Media Adaptation and Personalization (SMAP), 2011 Sixth International Workshop on (pp. 15-20). IEEE.

Sammut, C. (2011). TF-IDF. In Encyclopedia of machine learning. New York: Springer.

Shafiei, M., Wang, S., Zhang, R., Milios, E., Tang, B., Tougas, J., & Spiteri, R. (2007, April). Document representation and dimension reduction for text clustering. In Data Engineering Workshop, 2007 IEEE 23rd International Conference on (pp. 770-779). IEEE.

Shen, D. (2009). Text Categorization. Encyclopedia of Database Systems, 3041-3044.

Soderland, S. 1999. Learning Information Extraction Rules for Semi-structured and Free Text. Machine Learning, 34(1), 233–272.

Southgate, R., Roth, C., Schneider, J., Shi, P., Onishi, T., Wengner, D., Amman, W., Ogallo, L., Beddington J., & Murray, V. (2013). Using science for disaster risk reduction. Retrieved from www.preventionweb.net/go/scitech

Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H., & Demirbas, M. (2010, July). Short text classification in twitter to improve information filtering. InProceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval (pp. 841-842). ACM.

Stockdale, C. & McIntyre, D.A. (2011, May 09). The ten nations where Facebook rules the internet. Retrieved from http://247wallst.com/technology-3/2011/05/09/the-ten-nations-where-facebook-rules-the-internet/

Stone, R. (2004). Natural language processing challenges and advantages for Philippine languages. Proceedings from 1st Natural Language Processing Research Symposium (pp.81-86).

Téllez-Valero, A., Montes-y-Gómez, M., & Villaseñor-Pineda, L. (2005). A machine learning approach to information extraction. In Computational Linguistics and Intelligent Text Processing (pp. 539-547). Springer Berlin Heidelberg.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. ICWSM, 10, 178-185.

Twitter4J - A Java library for the Twitter API. (n.d.). Twitter4J - A Java library for the Twitter API. Retrieved July 29, 2014, from http://twitter4j.org/en/

Universal McCann. (2008). Power to the people: Social media tracker wave 3. Retrieved from http://web.archive.org/web/20080921002044/http://www.universalmccann.com/Assets/wave_3_20080403093750.pdf

Vieweg, S., Hughes, A. L., Starbird, K., & Palen, L. (2010, April). Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 1079-1088). ACM.

Wajeed, M. A., & Adilakshmi, T. (2011). Using KNN Algorithm for Text Categorization. In Computational Intelligence and Information Technology (pp. 796-801). Springer Berlin Heidelberg.

Wang, T., Bontcheva, K., Li, Y., & Cunningham, H. (2005). D2. 1.2/Ontology-Based Information Extraction (OBIE) v. 2. EU-IST Project IST-2003-506826 SEKT SEKT: Semantically Enabled Knowledge Technologies.

Wei, G., Gao, X., & Wu, S. (2010, July). Study of text classification methods for data sets with huge features. In Industrial and Information Systems (IIS), 2010 2nd International Conference on (Vol. 1, pp. 433-436). IEEE.

Weka 3: Data Mining Software in Java. (n.d.). *Weka 3*. Retrieved July 15, 2014, from http://www.cs.waikato.ac.nz/ml/weka/

Xu, H., Stenner, S. P., Doan, S., Johnson, K. B., Waitman, L. R., & Denny, J. C. (2010). MedEx: a medication information extraction system for clinical narratives. Journal of the American Medical Informatics Association, 17(1), 19-24.

Zeng, Q. T., Goryachev, S., Weiss, S., Sordo, M., Murphy, S. N., & Lazarus, R. (2006). Extracting principal diagnosis, co-morbidity and smoking status for asthma research: evaluation of a natural language processing system. BMC Medical Informatics and Decision Making, 6(1), 30.

Zhou, G., & Su, J. (2002, July). Named entity recognition using an HMM-based chunk tagger. In proceedings of the 40th Annual Meeting on Association for Computational Linguistics (pp. 473-480). Association for Computational Linguistics.

Zhou, S., Ling, T. W., Guan, J., Hu, J., & Zhou, A. (2003, March). Fast text classification: a training-corpus pruning based approach. In Database Systems for Advanced Applications, 2003. (DASFAA 2003). Proceedings. Eighth International Conference on (pp. 127-136). IEEE

Zsu, M. (2009). TF*IDF. In Encyclopedia of database systems. New York: Springer.

Appendix B: Examples of Filipino Morphemes

Morpheme Element	Root Word	Prefix / Suffix	Filipino Word
Elision	bigay	na- ; -an	nabigyan
Epenthesis	patay	-an	patayan
Metathesis	peteh (cebuano)	-en	pehten
Replacement	utos	-an	utusan
Nasal Assimilation	bigay	pan-	pamigay
Infixation	kain	-um-	kumain
Reduplication	matamis	-	matamis-tamis

Appendix C: List of Unified Hashtags Used in the Crawler Module

#reliefph
#nopower
#nowater
#roadalert
#tracingPH
#rescuePH
#floodph
#marioph
#rubyph

Appendix D: TFIDF Word Features (10%)

Caution and Advice		
W_8pm 18.003748032113865	W_humina 20.7215969398293	W_navotas 21.151794569238803
W_live 18.003748032113865	W_occ 20.7215969398293	W_orange 21.151794569238803
W_tonight 18.003748032113865	W_daraan 20.726842987965018	W_ingat 21.151794569238803
W_tsansang 18.545288004643915	W_2/2 20.726842987965018	W_kanina 21.151794569238803
W_i 18.607934569160065	W_feel 20.726842987965018	W_habang 21.151794569238803
W_god 18.607934569160065	W_effects 20.726842987965018	W_mamayang 21.151794569238803
W_s 18.67384305436798	W_sea 20.726842987965018	W_phl 21.167280610137155
W_nakapasok 18.97307777332978	W_tagalog 20.726842987965018	W_pasig 21.167280610137155
W_dost_pagasa 18.97307777332978	W_malabon 20.726842987965018	W_burias 21.167280610137155
W_torrijos 18.97307777332978	W_@johnsonmanabat	W_m 21.187935293655332
W_umaabot 18.97307777332978	20.726842987965018	W_hagupit 21.187935293655332
W_izon 18.97307777332978	W_@meralco 20.746256217541404	W_bahagyang 21.19141271925574
W_e 19.055679147282532	W_now 20.748931744846068	W_hindi 21.19141271925574
W_... 19.197687547237337	W_news 20.749665284275313	W_muntinlupa 21.19141271925574
W_everyone 19.27735831057902	W_pang 20.77587811027188	W_valenzuela 21.19141271925574
W_says 19.319654207668115	W_keep 20.793061475142146	W_tuesday 21.19141271925574
W_latest 19.319654207668115	W_occidental 20.801164039595655	W_aklan 21.19246702244009
W_caloocan 19.319654207668115	W_# 20.80223612393714	W_pasok 21.19567498448848
W_kanluran 19.319654207668115	W_@dzrhnews 20.80223612393714	W_tubig 21.197867798817075
W_& 19.347488640266004	W_japan 20.80223612393714	W_lakas 21.20212155477237
W_@hadjirieta 19.60921856001766	W_provinces 20.80223612393714	W_tom 21.207600939352925
W_maramdaman 19.60921856001766	W_track 20.80223612393714	W_hernandez 21.207600939352925
W_ngayon 19.630422562536587	W_handa 20.80223612393714	W_sabado 21.215616318655638
W_eastern 19.656168085770492	W_polillo 20.80223612393714	W_ito 21.226898551578344
W_cuã±a 19.66587217692185	W_1/4 20.80223612393714	W_#3 21.22848969765599
W_isâ€¦ 19.66587217692185	W_semirara 20.80223612393714	W_layong 21.22848969765599
W_para 19.776658454442288	W_nag-landfall 20.80223612393714	W_km 21.22848969765599
W_sapat 19.845888677698312	W_8-10pm 20.80223612393714	W_kalupaan 21.249115533666846
W_suspended 19.845888677698312	W_bago 20.83499106660949	W_surge 21.253475972793122
W_pls 19.845888677698312	W_nueva 20.849393857813627	W_capiz 21.284458796046668
W_tulad 19.845888677698312	W_ecija 20.849393857813627	W_per 21.285431265854903
W_!! 19.845888677698312	W_leyte 20.863705719981553	W_@iamsumulong 21.285431265854903
W_hilagang-kanluran	W_borongan 20.86377006268562	W_ticao 21.285431265854903
19.845888677698312	W_violent 20.87911787925307	W_luzon 21.29635213581166
W_pumasok 19.845888677698312	W_circuit 20.87911787925307	W_posibleng 21.29635213581166
W_delata 19.845888677698312	W_patayin 20.87911787925307	W_par 21.297440906044358
W_lalo 19.898584335374846	W_amerika 20.87911787925307	W_n 21.300967842424626
W_the 20.050064374556886	W_landfall 20.87911787925307	W_5am 21.300967842424626
W_pagitan 20.058224314594536	W_makaiwas 20.87911787925307	W_loob 21.300967842424626
W_govt 20.058224314594536	W_miyerkules 20.87911787925307	W_lalawigan 21.300967842424626
W_tayo 20.058224314594536	W_estrada 20.87911787925307	W_inaasahang 21.302961558568867
W_project 20.058224314594536	W_itinuturing 20.87911787925307	W_|via 21.310442160607867
W_mata 20.058224314594536	W_breaker 20.87911787925307	W_responsibility
W_posibilidad 20.058224314594536	W_intense 20.87911787925307	21.310442160607867
W_@adamsonuni 20.058224314594536	W_tumutok 20.87911787925307	W_warning 21.310442160607867
W_.. 20.121909449496254	W_aksidente 20.87911787925307	W_on 21.310442160607867
W_ulan 20.132378841533498	W_guimaras 20.87911787925307	W_@radyopatrol39
W_calabarzon 20.201954243484558	W_camsur 20.89437535943061	21.313776328320976
W_@news5aksyon 20.22190894978084	W_area 20.906082434957888	W_): 21.31749655909268
W_padua 20.2251101359746	W_island| 20.916306437528796	W_paalala 21.31749655909268
W_school 20.2251101359746	W_just 20.936332476674664	W_ninyo 21.31749655909268
W_guys 20.2251101359746	W_araw 20.94461771787911	W_sumusunod 21.31749655909268
W_lumabas 20.2251101359746	W_gma 20.94461771787911	W_antique 21.31749655909268
W_pa-west 20.2251101359746	W_rains 20.94461771787911	W_aming 21.318611563181395
W_papalapit 20.2251101359746	W_@govramil 20.94461771787911	W_serbisyo 21.318611563181395
W_maynila 20.226781683122418	W_sibuyan 20.94461771787911	W_numero 21.318611563181395
W_rin 20.255728381572325	W_upang 20.94461771787911	W_kailanganin 21.318611563181395
W_malapit 20.28420994290172	W_number 20.94461771787911	W_pag-akyat 21.318611563181395
W_mayor 20.292302649436017	W_magla-landfall	W_grp 21.325369990264992
W_yellow 20.31580040293404	20.94461771787911	W_suspends 21.325369990264992
W_ilang 20.34593589932558	W_masbate 20.9487592939918	W_negros 21.325369990264992
W_bandang 20.386431556497865	W_din 20.968274407933336	W_tarlac 21.325369990264992
W_a 20.386431556497865	W_calauag 20.982040184373524	W_@rizalgov 21.325369990264992
W_pio 20.386431556497865	W_jma 21.00303050155502	W_monday 21.328737409119363
W_forecast 20.386431556497865	W_tanghali 21.00303050155502	W_lalabas 21.328737409119363

Caution and Advice		
W_thursday 20.386431556497865 W_lungsod 20.386431556497865 W_silangan 20.386431556497865 W_lumihis 20.386431556497865 W_[20.399035244223743 W_] 20.399035244223743 W_lumakas 20.399035244223743 W_gov 20.421433497914073 W_muling 20.448252183669062 W_iloilo 20.454832262800547 W_pasukin 20.483301419780755 W_including 20.503400639927143 W_@feutamz 20.503400639927143 W_feu-nrmf 20.503400639927143 W_or 20.503400639927143 W_alon 20.51088619401681 W_:(20.568009808944428 W_suspend 20.592923596043672 W_under 20.62652419672737 W_west 20.628742020975263 W_madaling 20.628742020975263 W_mag 20.628742020975263 W_bilis 20.628742020975263 W_martes 20.640284498272617 W_group 20.673286596971366 W_sana 20.69474517708961 W_mula 20.701000339306514	W_expect 21.00303050155502 W_kung 21.010157791316523 W_may 21.025044612014316 W_pero 21.041990491852765 W_cebu 21.054296886778427 W_areas 21.058618433071288 W_supertyphoon 21.058618433071288 W_pasay 21.058618433071288 W_1/2 21.058618433071288 W_tatama 21.058618433071288 W_zambales 21.07700274158617 W_las 21.090468114278227 W_erap 21.090468114278227 W_@rapplerdotcom 21.090468114278227 W_makati 21.090468114278227 W_classes 21.09326326245795 W_naman 21.093616795951434 W_tandaan 21.09600545526976 W_lang 21.103225854742906 W_kph 21.106555863946255 W_ay 21.116261546322598 W_feu 21.124248883257323 W_itinaas 21.12848095747243 W_gabi 21.128643658565398 W_december 21.143444843529906 W_hanggang 21.15061145377966 W_jtwc 21.151794569238803	W_umaga 21.331515403770837 W_huling 21.331515403770837 W_@robertmanodzmm 21.331515403770837 W_11pm 21.331515403770837 W_namataan 21.331515403770837 W_@iskomoreno 21.3344466614766 W_mamayaâ€¦ 21.504837490149264 W_http://t.â€¦ 21.504837490149264 W_queâ€¦ 21.504837490149264 W_http://t.coâ€¦ 23.007381555720713 W_http://t.co/ekfowâ€¦ 23.007381555720713 W_piã 25.378220190135295 W_paraã±aque 25.378220190135295 W_â€? 26.34889093451958 W_http://t.co/c61â€¦ 28.002653484065497 W_camarinesâ€¦ 28.71972956893416 W_piã±as 28.71972956893416 W_câ€¦ 29.990112475109143 W_http://t.co/i5ibubfnâ€¦ 30.558694882161575 W_http://t.co/lonurxpgâ€¦ 30.558694882161575 W_vâ€¦ 36.271042895375025 W_http://t.co/lkvkotzxâ€¦ 39.043969867121454 W_â€œ 39.237888841187626 W_http://tâ€¦ 41.56067383420147 W_http://t.câ€¦ 42.11961179761311 W_romblâ€¦ 42.515264229775624 W_http://t.co/â€¦ 44.03924179546624

Casualty and Damage		
W_baylon 14.988365721796793 W_bubong 14.988365721796793 W_ajuy 14.988365721796793 W_utos 14.988365721796793 W_#radyopatrol 15.388948396738515 W_katao 15.822762373529738 W_preemptive 15.822762373529738 W_brgy 15.822762373529738 W_dilg 16.356507497847893 W_kuryente 16.50520946752559 W_tacloban 16.58340940659383 W_by 16.827976478180986 W_@akosijaysent 16.923884280941536 W_northeastern 17.302356611842104 W_samar- 17.302356611842104 W_downed 17.302356611842104 W_electrical 17.302356611842104 W_@edlingao 17.302356611842104 W_abucay 17.302356611842104 W_nkklk 17.302356611842104 W_topples 17.302356611842104	W_uy-tan 17.302356611842104 W_zhander 17.302356611842104 W_6:00 17.302356611842104 W_electric 17.302356611842104 W_isinasagawa 17.302356611842104 W_photo 17.535843167999637 W_lumikas 17.98609544141216 W_#cebu 18.003748032113865 W_ann 18.003748032113865 W_barangay 18.284295645158867 W_nakatira 18.284295645158867 W_pswd 18.545288004643915 W_imprastraktura 18.545288004643915 W_total 18.545288004643915 W_iniwan 18.545288004643915 W_bagsak 18.545288004643915 W_mahigit 18.84348886290733 W_ulat 18.93698480446941 W_taclobanon 18.97307777332978	W_sapilitan 18.97307777332978 W_pinalilikas 18.97307777332978 W_komunikasyon 18.97307777332978 W_evacuees 18.97307777332978 W_linya 18.97307777332978 W_pamilya 19.00882843370869 W_catbalogan 19.195581931273235 W_residente 19.262639720301475 W_probinsya 19.27735831057902 W_naitalang 19.319654207668115 W_#aksyonsahagupit 19.348557039594567 W_30,689 19.60921856001766 W_post 19.60921856001766 W_malawak 19.82734315459205 W_casualty 20.058224314594536 W_pananalasa 20.09273460429346 W_video 20.25421585957266 W_dalawa 20.628742020975263 W_nagdulot 20.646448051992703 W_nasawi 20.726842987965018 W_pinsala 20.86377006268562

Call For Help		
W_tumulong 12.779923969230259	W_center 17.490647149520154 W_binuksan 18.003748032113865	W_bilang 19.055679147282532 W_kapilya 19.557409258992998

Call For Help		
W_nagsisiksikang 12.779923969230259 W_supply 12.779923969230259 W_nagsimula 14.988365721796793 W_tacloban 16.813880662710172 W_lubog 16.95589940235178 W_kailangang 17.200608882219456	W_village 18.003748032113865 W_kapuso 18.150363179415432 W_kuryente 18.924580994206224 W_tablas 18.97307777332978 W_8:30 18.97307777332978	W_ipinagagamit 19.557409258992998 W_simbahan 19.557409258992998 W_@_iancruz 20.058224314594536 W_gumaca 20.058224314594536

Donation		
W_dswd 12.779923969230259 W_http://t.co/oq8rw7jw… 13.953524163708405 W_salamat 14.16870916513339 W_san 14.237600391676462 W_food 17.200608882219456 W_@dswdserves 17.302356611842104 W_family 17.574868470828566 W_mateo 18.003748032113865	W_@vargasmannysen 18.003748032113865 W_ilikas 18.003748032113865 W_packs 18.003748032113865 W_navy 18.545288004643915 W_towns 18.97307777332978 W_inihahanda 19.60921856001766 W_relief 20.275841505975105 W_goods 20.399035244223743	W_transported 20.503400639927143 W_personnel 20.503400639927143 W_ps36 20.503400639927143 W_agutaya 20.628742020975263 W_magsaysay 20.628742020975263 W_http://t.co/xck… 23.007381555720713 W_http://t.co/… 24.277764461895703 W_http://t.co… 25.378220190135295

Appendix E: TFIDF Word Features (20%)

Caution and Advice		
W_abangan 12.779923969230259 W_hala 12.779923969230259 W_abscbnnews 12.779923969230259 W_implikasyon 12.779923969230259 W_flight 12.779923969230259 W_karatig 12.779923969230259 W_idineklarang 12.779923969230259 W_b 12.779923969230259 W_g 12.779923969230259 W_p 12.779923969230259 W_blue 12.779923969230259 W_y 12.779923969230259 W_tuluyan 12.779923969230259 W_dumiretso 12.779923969230259 W_kalamado 12.779923969230259 W_ala-una 12.779923969230259 W_sobrang 12.779923969230259 W_@beabinene 12.779923969230259 W_tatawaging 12.779923969230259 W_umalis 12.779923969230259 W_@mmaarryyeell 12.779923969230259 W_schools 12.779923969230259 W_anak 12.779923969230259 W_alas-sais 12.779923969230259 W_southeast 12.779923969230259 W_â?¤ 13.953524163708405 W_@dzmmteleradyo 14.683886862001419 W_@zhandercayabyab 14.804184822851791 W_#fb 14.988365721796793 W_+ 14.988365721796793 W_!!!! 14.988365721796793 W_teritoryo 14.988365721796793 W_entering 14.988365721796793 W_u 14.988365721796793 W_pag 14.988365721796793 W_@rida_reyes 14.988365721796793 W_officialmunti 14.988365721796793 W_with 14.988365721796793 W_08dec14 14.988365721796793 W_naka-red 14.988365721796793 W_2/4 14.988365721796793 W_surigao 14.988365721796793 W_paskuhan 14.988365721796793	W_izon 18.97307777332978 W_pala 18.97307777332978 W_pano 18.97307777332978 W_#superbalitasagabi 18.97307777332978 W_11:00 18.97307777332978 W_5:00 18.97307777332978 W_aguilar 18.97307777332978 W_kahit 18.97307777332978 W_isa 18.97307777332978 W_rod 18.97307777332978 W_meters 18.97307777332978 W_@dzbbsamnielsen 18.97307777332978 W_rp12 18.97307777332978 W_omg 18.97307777332978 W_p.m. 18.98363668371316 W_#imready 18.98363668371316 W_epekto 18.98363668371316 W_oriental 19.00216640961615 W_suspendido 19.048672358429886 W_e 19.055679147282532 W_klase 19.126095902383177 W_... 19.197687547237337 W_lunes 19.260486700138248 W_talisay 19.262639720301475 W_everyone 19.27735831057902 W_talaga 19.27735831057902 W_ko 19.27735831057902 W_halos 19.289216071161423 W_says 19.319654207668115 W_latest 19.319654207668115 W_caloocan 19.319654207668115 W_kanluran 19.319654207668115 W_light 19.319654207668115 W_#news 19.319654207668115 W_-- 19.319654207668115 W_taya 19.319654207668115 W_herbert 19.319654207668115 W_moderate 19.319654207668115 W_stay 19.319654207668115 W_nito 19.319654207668115 W_ano 19.319654207668115 W_ahensya 19.319654207668115 W_@ukgdos 19.319654207668115 W_bautista 19.319654207668115 W_alert 19.319654207668115 W_ka 19.319654207668115 W_alas-diyes 19.319654207668115 W_weather 19.319654207668115 W_12nn 19.319654207668115 W_& 19.347488640266004 W_to 19.473838825479692 W_cuyo 19.49467023127168 W_update 19.593023663635186	W_feel 20.726842987965018 W_effects 20.726842987965018 W_sea 20.726842987965018 W_tropical 20.726842987965018 W_tagalog 20.726842987965018 W_malabon 20.726842987965018 W_@johnsonmanabat 20.726842987965018 W_#2 20.746256217541404 W_@meralco 20.746256217541404 W_now 20.748931744846068 W_news 20.749665284275313 W_pang 20.77587811027188 W_keep 20.793061475142146 W_safe 20.79622522650962 W_occidental 20.801164039595655 W_# 20.80223612393714 W_@dzrhnews 20.80223612393714 W_japan 20.80223612393714 W_provinces 20.80223612393714 W_track 20.80223612393714 W_handa 20.80223612393714 W_polillo 20.80223612393714 W_1/4 20.80223612393714 W_semirara 20.80223612393714 W_nag-landfall 20.80223612393714 W_8-10pm 20.80223612393714 W_bago 20.83499106660949 W_nueva 20.849393857813627 W_ecija 20.849393857813627 W_leyte 20.863705719981553 W_borongan 20.86377006268562 W_violent 20.87911787925307 W_circuit 20.87911787925307 W_patayin 20.87911787925307 W_amerika 20.87911787925307 W_landfall 20.87911787925307 W_makaiwas 20.87911787925307 W_miyerkules 20.87911787925307 W_estrada 20.87911787925307 W_itinuturing 20.87911787925307 W_breaker 20.87911787925307 W_intense 20.87911787925307 W_tumutok 20.87911787925307 W_lalong 20.87911787925307 W_aksidente 20.87911787925307 W_guimaras 20.87911787925307 W_camsur 20.89437535943061 W_area 20.906082434957888 W_island| 20.916306437528796 W_just 20.936332476674664 W_visayas 20.936332476674664 W_araw 20.94461771787911

Caution and Advice		
W_@sherieanntorres 14.988365721796793 W_calapan 14.988365721796793 W_buti 14.988365721796793 W_nagbago 14.988365721796793 W_thank 15.02235776098519 W_list 16.0772879715977 W_catbalogan 16.323560398323025 W_!!! 16.356507497847893 W_your 16.356507497847893 W_nitong 16.356507497847893 W_mabalacat 16.356507497847893 W_@pasiginfo 16.356507497847893 W_sakop 16.356507497847893 W_@jonvicremulla 16.356507497847893 W_almost 16.356507497847893 W_pia 16.356507497847893 W_@pdrrmcbulacan 16.356507497847893 W_kalakasan 16.356507497847893 W_carcar 16.356507497847893 W_matinding 16.356507497847893 W_umuulan 16.356507497847893 W_hay 16.356507497847893 W_umali 16.356507497847893 W_itaas 16.356507497847893 W_#dobolbbalitangbalita 16.356507497847893 W_yey 16.356507497847893 W_compostela 16.356507497847893 W_camiguin 16.356507497847893 W_critical 16.356507497847893 W_lubao 16.356507497847893 W_detalye 16.356507497847893 W_@ernie_manio 16.356507497847893 W_pass 16.356507497847893 W_daanan 16.356507497847893 W_11:05 16.356507497847893 W_exit 16.356507497847893 W_guagua 16.356507497847893 W_ayan 16.356507497847893 W_12:00 16.356507497847893 W_mexico 16.356507497847893 W_siya 16.356507497847893 W_category 16.356507497847893 W_natitirang 16.427105773014038 W_you 16.427105773014038 W_namin 16.827976478180986 W_kababayan 16.827976478180986	W_@hadjirieta 19.60921856001766 W_maramdaman 19.60921856001766 W_hapon 19.60921856001766 W_del 19.60921856001766 W_raw 19.60921856001766 W_ncr 19.60921856001766 W_tomorrow 19.60921856001766 W_raised 19.60921856001766 W_tues 19.60921856001766 W_#hiritpanahon 19.60921856001766 W_gov't 19.60921856001766 W_@peeweehero 19.627237435130677 W_ngayon 19.630422562536587 W_walang 19.65221207276957 W_eastern 19.656168085770492 W_cuã±a 19.66587217692185 W_isâ€¦ 19.66587217692185 W_https://t.â€¦ 19.66587217692185 W_@abscbnnews 19.68440176194164 W_ayon 19.717386357239445 W_ba 19.771001980162918 W_para 19.776658454442288 W_nasa 19.81880684836712 W_sapat 19.845888677698312 W_suspended 19.845888677698312 W_pls 19.845888677698312 W_tulad 19.845888677698312 W_pagkain 19.845888677698312 W_coastal 19.845888677698312 W_!! 19.845888677698312 W_hilagang-kanluran 19.845888677698312 W_pumasok 19.845888677698312 W_delata 19.845888677698312 W_imbak 19.845888677698312 W_delikado 19.845888677698312 W_alas-otso 19.845888677698312 W_lalo 19.898584335374846 W_pm 19.8992613061474 W_is 19.899840905150437 W_rainfall 19.930934002176514 W_norte 19.940424095313666 W_palawan 19.944989480033932 W_offices 20.025667136521548 W_the 20.050064374556886 W_pagitan 20.058224314594536 W_govt 20.058224314594536 W_@deped_ph 20.058224314594536 W_tayo 20.058224314594536 W_project 20.058224314594536	W_gma 20.94461771787911 W_rains 20.94461771787911 W_@govramil 20.94461771787911 W_sibuyan 20.94461771787911 W_upang 20.94461771787911 W_number 20.94461771787911 W_magla-landfall 20.94461771787911 W_masbate 20.9487592939918 W_din 20.968274407933336 W_calauag 20.982040184373524 W_jma 21.00303050155502 W_tanghali 21.00303050155502 W_expect 21.00303050155502 W_kung 21.010157791316523 W_may 21.025044612014316 W_pero 21.041990491852765 W_cebu 21.054296886778427 W_areas 21.058618433071288 W_supertyphoon 21.058618433071288 W_pasay 21.058618433071288 W_1/2 21.058618433071288 W_tatama 21.058618433071288 W_zambales 21.07700274158617 W_las 21.090468114278227 W_erap 21.090468114278227 W_@rapplerdotcom 21.090468114278227 W_makati 21.090468114278227 W_classes 21.09326326245795 W_naman 21.093616795951434 W_tandaan 21.09600545526976 W_lang 21.103225854742906 W_kph 21.106555863946255 W_ay 21.116261546322598 W_feu 21.124248883257323 W_itinaas 21.12848095747243 W_gabi 21.128643658565398 W_december 21.143444843529906 W_hanggang 21.15061145377966 W_jtwc 21.151794569238803 W_navotas 21.151794569238803 W_orange 21.151794569238803 W_ingat 21.151794569238803 W_kanina 21.151794569238803 W_habang 21.151794569238803 W_mamayang 21.151794569238803 W_phl 21.167280610137155 W_pasig 21.167280610137155 W_burias 21.167280610137155 W_m 21.187935293655332 W_hagupit 21.187935293655332 W_bahagyang 21.19141271925574 W_hindi 21.19141271925574 W_muntinlupa 21.19141271925574 W_valenzuela 21.19141271925574

Caution and Advice		
W_ating 16.869577432501142 W_ulat 17.038543960286876 W_tsk 17.200608882219456 W_@24orasgma 17.287942481590918 W_hall 17.302356611842104 W_30km 17.302356611842104 W_slowly 17.302356611842104 W_nananatili 17.302356611842104 W_9am 17.302356611842104 W_juan 17.302356611842104 W_yes 17.302356611842104 W_eastern-northern 17.302356611842104 W_pa-kanluran 17.302356611842104 W_hanging 17.302356611842104 W_daw 17.302356611842104 W_bagal 17.302356611842104 W_@divinerey 17.302356611842104 W_biglaang 17.302356611842104 W_felt 17.302356611842104 W_inc 17.302356611842104 W_floods 17.302356611842104 W_adm 17.302356611842104 W_int'l 17.302356611842104 W_taglay 17.302356611842104 W_eto 17.302356611842104 W_waves 17.302356611842104 W_11am 17.302356611842104 W_totoo 17.302356611842104 W_parts 17.302356611842104 W_malalakas 17.302356611842104 W_winds 17.470667348306666 W_yan 17.801053391759794 W_island 17.80150292972814 W_bugsong 17.98591454978141 W_for 17.98609544141216 W_8pm 18.003748032113865 W_live 18.003748032113865 W_tonight 18.003748032113865 W_pray 18.003748032113865 W_dumaan 18.003748032113865 W_#breakingweathernow 18.003748032113865 W_25% 18.003748032113865 W_robertmanodzmm 18.003748032113865 W_@mandaluyongc3 18.003748032113865 W_dis 18.003748032113865 W_pumalaot 18.003748032113865 W_abet 18.003748032113865 W_tsansa 18.003748032113865 W_13kph-15kph 18.003748032113865	W_mata 20.058224314594536 W_posible 20.058224314594536 W_posibilidad 20.058224314594536 W_@adamsonuni 20.058224314594536 W_northwest 20.058224314594536 W_noah 20.058224314594536 W_dito 20.11611192645208 W_.. 20.121909449496254 W_ulan 20.132378841533498 W_camarines 20.1815870607581 W_hangin 20.18888165641876 W_advisory 20.192158042388535 W_calamian 20.201954243484558 W_calabarzon 20.201954243484558 W_2:30 20.205165103501887 W_@news5aksyon 20.22190894978084 W_padua 20.2251101359746 W_school 20.2251101359746 W_guys 20.2251101359746 W_lumabas 20.2251101359746 W_pa-west 20.2251101359746 W_bong 20.2251101359746 W_bay 20.2251101359746 W_mag-landfall 20.2251101359746 W_papalapit 20.2251101359746 W_maynila 20.226781683122418 W_rin 20.255728381572325 W_#hagupit 20.28395001937971 W_malapit 20.28420994290172 W_mayor 20.292302649436017 W_yellow 20.31580040293404 W_bagyo 20.332256528053765 W_ilang 20.34593589932558 W_bandang 20.386431556497865 W_a 20.386431556497865 W_pio 20.386431556497865 W_forecast 20.386431556497865 W_thursday 20.386431556497865 W_lungsod 20.386431556497865 W_silangan 20.386431556497865 W_bugso 20.386431556497865 W_lumihis 20.386431556497865 W_dadaanan 20.386431556497865 W_[20.399035244223743 W_] 20.399035244223743 W_lumakas 20.399035244223743 W_gov 20.421433497914073 W_muling 20.448252183669062 W_iloilo 20.454832262800547 W_pasukin 20.483301419780755 W_including 20.503400639927143 W_@feutamz 20.503400639927143	W_tuesday 21.19141271925574 W_aklan 21.19246702244009 W_pasok 21.19567498448848 W_tubig 21.197867798817075 W_lakas 21.20212155477233 7 W_tom 21.207600939352925 W_hernandez 21.207600939352925 W_sabado 21.215616318655638 W_ito 21.226898551578344 W_#3 21.22848969765599 W_layong 21.22848969765599 W_n.u. 21.22848969765599 W_km 21.22848969765599 W_kalupaan 21.249115533666846 W_surge 21.253475972793122 W_capiz 21.284458796046668 W_per 21.285431265854903 W_@iamsumulong 21.285431265854903 W_ticao 21.285431265854903 W_luzon 21.29635213581166 W_posibleng 21.29635213581166 W_par 21.297440906044358 W_n 21.300967842424626 W_5am 21.300967842424626 W_loob 21.300967842424626 W_lalawigan 21.300967842424626 W_inaasahang 21.302961558568867 W_|via 21.310442160607867 W_responsibility 21.310442160607867 W_warning 21.310442160607867 W_on 21.310442160607867 W_@radyopatrol39 21.313776328320976 W_): 21.31749655909268 W_paalala 21.31749655909268 W_ninyo 21.31749655909268 W_sumusunod 21.31749655909268 W_antique 21.31749655909268 W_aming 21.318611563181395 W_serbisyo 21.318611563181395 W_numero 21.318611563181395 W_kailanganin 21.318611563181395 W_pag-akyat 21.318611563181395 W_grp 21.325369990264992 W_suspends 21.325369990264992 W_negros 21.325369990264992 W_tarlac 21.325369990264992 W_@rizalgov 21.325369990264992 W_monday 21.328737409119363 W_lalabas 21.328737409119363 W_umaga 21.331515403770837

Caution and Advice		
W_garcia 18.003748032113865 W_bawal 18.003748032113865 W_9:15 18.003748032113865 W_binaklas 18.003748032113865 W_coding 18.003748032113865 W_kita 18.003748032113865 W_kumikilos 18.003748032113865 W_lalapitan 18.003748032113865 W_tolentino 18.003748032113865 W_wala 18.084299357859834 W_ngayong 18.114510881250915 W_@dennis_datu 18.150363179415432 W_baha 18.288475358516312 W_#tvpatrol 18.501474107018684 W_tsansang 18.545288004643915 W_asahan 18.545288004643915 W_mmda 18.545288004643915 W_bahagya 18.545288004643915 W_east 18.545288004643915 W_mabagal 18.545288004643915 W_calixto 18.545288004643915 W_15kph 18.545288004643915 W_@akoposizandro 18.545288004643915 W_75% 18.545288004643915 W_buhos 18.545288004643915 W_3am 18.545288004643915 W_be 18.545288004643915 W_pinag-iingat 18.545288004643915 W_kasi 18.545288004643915 W_@nababaha 18.545288004643915 W_downgraded 18.545288004643915 W_romblon 18.557690969543565 W_i 18.607934569160065 W_god 18.607934569160065 W_medyo 18.607934569160065 W_saan 18.607934569160065 W_pagasa 18.666733183531655 W_s 18.67384305436798 W_sorsogon 18.822279518036765 W_nakapasok 18.97307777332978 W_dost_pagasa 18.97307777332978 W_torrijos 18.97307777332978 W_umaabot 18.97307777332978	W_feu-nrmf 20.503400639927143 W_diliman 20.503400639927143 W_high 20.503400639927143 W_or 20.503400639927143 W_alon 20.51088619401681 W_strong 20.51818027921944 W_:(20.568009808944428 W_po 20.582952514262743 W_suspend 20.592923596043672 W_#1 20.60135986030804 W_under 20.62652419672737 W_storm 20.62769727974351 W_west 20.628742020975263 W_madaling 20.628742020975263 W_mag 20.628742020975263 W_bilis 20.628742020975263 W_kasama 20.628742020975263 W_martes 20.640284498272617 W_group 20.673286596971366 W_sana 20.69474517708961 W_mula 20.701000339306514 W_humina 20.7215969398293 W_occ 20.7215969398293 W_daraan 20.726842987965018 W_2/2 20.726842987965018	W_huling 21.331515403770837 W_@robertmanodzmm 21.331515403770837 W_11pm 21.331515403770837 W_namataan 21.331515403770837 W_@iskomoreno 21.3344466614766 W_mamayaâ€¦ 21.504837490149264 W_http://t.â€¦ 21.504837490149264 W_queâ€¦ 21.504837490149264 W_http://t.coâ€¦ 23.007381555720713 W_http://t.co/ekfowâ€¦ 23.007381555720713 W_piã 25.378220190135295 W_paraã±aque 25.378220190135295 W_â€? 26.34889093451958 W_http://t.co/c61â€¦ 28.002653484065497 W_camarinesâ€¦ 28.71972956893416 W_piã±as 28.71972956893416 W_câ€¦ 29.990112475109143 W_http://t.co/i5ibubfnâ€¦ 30.558694882161575 W_http://t.co/lonurxpgâ€¦ 30.558694882161575 W_vâ€¦ 36.271042895375025 W_http://t.co/lkvkotzxâ€¦ 39.043969867121454 W_â€œ 39.237888841187626 W_http://tâ€¦ 41.56067383420147 W_http://t.câ€¦ 42.11961179761311 W_romblâ€¦ 42.515264229775624 W_http://t.co/â€¦ 44.03924179546624

Casualty and Damage		
W_@ancalerts 10.503038540285301 W_you 10.658811881212419 W_galera 12.779923969230259	W_walang 15.108372282361477 W_#radyopatrol 15.388948396738515 W_matapos 15.388948396738515	W_daanbantayan 17.302356611842104 W_6:00 17.302356611842104 W_electric 17.302356611842104

Casualty and Damage		
W_naibalik 12.779923969230259 W_airport 12.779923969230259 W_#votynews 12.779923969230259 W_dead 12.779923969230259 W_herrera 12.779923969230259 W_ormoc 12.779923969230259 W_good 12.779923969230259 W_nagpapatupad 12.779923969230259 W_@victorcosare777 12.779923969230259 W_rail 12.779923969230259 W_front 12.779923969230259 W_20-anyos 12.779923969230259 W_pasahero 12.779923969230259 W_list 12.929985398341222 W_thank 13.211322591055797 W_ayon 13.584670498681378 W_talisay 13.812550291706431 W_legazpi 13.91523363537655 W_http 14.16870916513339 W_estancia 14.384632589319974 W_nanatili 14.384632589319974 W_#aksyon 14.419500657334172 W_buong 14.557634565767035 W_ulan 14.786785819809174 W_@zhandercayabyab 14.804184822851791 W_baylon 14.988365721796793 W_bubong 14.988365721796793 W_ajuy 14.988365721796793 W_utos 14.988365721796793 W_years 14.988365721796793 W_nasaktan 14.988365721796793 W_old 14.988365721796793 W_pinapakuan 14.988365721796793 W_agbobolo 14.988365721796793 W_@arnellozaeta 14.988365721796793 W_causalities 14.988365721796793 W_9,584 14.988365721796793 W_ernesto 14.988365721796793	W_odiongan 15.388948396738515 W_press 15.388948396738515 W_eastern 15.468282816699496 W_katao 15.822762373529738 W_preemptive 15.822762373529738 W_brgy 15.822762373529738 W_@gepelyle 15.822762373529738 W_higit 15.822762373529738 W_nang 16.041593592943087 W_baha 16.119947282327892 W_calamity 16.356507497847893 W_dilg 16.356507497847893 W_28,000 16.356507497847893 W_state 16.356507497847893 W_camalig 16.356507497847893 W_@sunstarcebu 16.356507497847893 W_tinangay 16.356507497847893 W_tan 16.356507497847893 W_romblon 16.490924059619076 W_kuryente 16.50520946752559 W_tacloban 16.58340940659383 W_by 16.827976478180986 W_puno 16.827976478180986 W_@akosijaysent 16.923884280941536 W_northeastern 17.302356611842104 W_samar- 17.302356611842104 W_downed 17.302356611842104 W_cayabyab 17.302356611842104 W_@dsm_sunstar 17.302356611842104 W_electrical 17.302356611842104 W_@edlingao 17.302356611842104 W_abucay 17.302356611842104 W_nkklk 17.302356611842104 W_topples 17.302356611842104 W_uy-tan 17.302356611842104 W_zhander 17.302356611842104	W_isinasagawa 17.302356611842104 W_evacuation 17.48210799446251 W_photo 17.535843167999637 W_lumikas 17.98609544141216 W_#cebu 18.003748032113865 W_ann 18.003748032113865 W_barangay 18.284295645158867 W_nakatira 18.284295645158867 W_pswd 18.545288004643915 W_imprastraktura 18.545288004643915 W_total 18.545288004643915 W_iniwan 18.545288004643915 W_bagsak 18.545288004643915 W_inilikas 18.607934569160065 W_mahigit 18.84348886290733 W_ulat 18.93698480446941 W_dagat 18.97307777332978 W_taclobanon 18.97307777332978 W_sapilitan 18.97307777332978 W_pinalilikas 18.97307777332978 W_komunikasyon 18.97307777332978 W_evacuees 18.97307777332978 W_linya 18.97307777332978 W_pamilya 19.00882843370869 W_catbalogan 19.195581931273235 W_residente 19.262639720301475 W_probinsya 19.27735831057902 W_naitalang 19.319654207668115 W_#aksyonsahagupit 19.348557039594567 W_30,689 19.60921856001766 W_post 19.60921856001766 W_malawak 19.82734315459205 W_casualty 20.058224314594536 W_pananalasa 20.09273460429346 W_video 20.25421585957266 W_dalawa 20.628742020975263 W_nagdulot 20.646448051992703 W_nasawi 20.726842987965018 W_pinsala 20.86377006268562

Call For Help		
W_mauubusan 8.24117615049496 W_pre-emptive 8.24117615049496 W_inirekumenda 8.24117615049496 W_near 8.24117615049496 W_as 9.786630986868186 W_narito 11.226722879586054 W_bayan 11.404210347066746 W_tumulong 12.779923969230259 W_nagsisiksikang 12.779923969230259 W_supply 12.779923969230259 W_complex 12.779923969230259 W_evacuate 12.779923969230259 W_hospital 12.779923969230259 W_sports 12.779923969230259	W_tumama 13.91523363537655 W_lumikas 14.777633867935151 W_nagsimula 14.988365721796793 W_taga-eastern 14.988365721796793 W_magkumpuni 14.988365721796793 W_photo 15.108372282361477 W_buong 15.217498960310474 W_nang 15.68141664291429 W_regional 16.356507497847893 W_@allangatus 16.503338690643297 W_@24orasgma 16.651069376032837 W_tacloban 16.813880662710172 W_lubog 16.95589940235178	W_kailangang 17.200608882219456 W_center 17.490647149520154 W_binuksan 18.003748032113865 W_village 18.003748032113865 W_kapuso 18.150363179415432 W_kuryente 18.924580994206224 W_tablas 18.97307777332978 W_8:30 18.97307777332978 W_bilang 19.055679147282532 W_kapilya 19.557409258992998 W_ipinagagamit 19.557409258992998 W_simbahan 19.557409258992998 W_@_iancruz 20.058224314594536 W_gumaca 20.058224314594536

Donation		
W_tumatanggap 8.24117615049496 W_offer 8.24117615049496 W_bangus 8.24117615049496	W_#unanghirit 10.149480641391612 W_volunteers 12.779923969230259 W_dswd 12.779923969230259	W_@vargasmannysen 18.003748032113865 W_ilikas 18.003748032113865

Donation		
W_supermalls 8.24117615049496 W_mgs 8.24117615049496 W_mangudadatu 8.24117615049496 W_victims 8.24117615049496 W_ginawa 8.24117615049496 W_#dswd 8.24117615049496 W_benloi 8.24117615049496 W_9-10 8.24117615049496 W_australia 8.24117615049496 W_countries 8.24117615049496 W_court 8.24117615049496 W_8am-6pm 8.24117615049496 W_globe 8.24117615049496	W_point 12.779923969230259 W_st. 12.779923969230259 W_http://t.co/oq8rw7jw… 13.953524163708405 W_salamat 14.16870916513339 W_san 14.237600391676462 W_200,000 14.268563439747771 W_photo 14.626378374680842 W_free 14.988365721796793 W_food 17.200608882219456 W_@dswdserves 17.302356611842104 W_family 17.574868470828566 W_mateo 18.003748032113865 W_preso 18.003748032113865	W_packs 18.003748032113865 W_navy 18.545288004643915 W_towns 18.97307777332978 W_inihahanda 19.60921856001766 W_relief 20.275841505975105 W_goods 20.399035244223743 W_transported 20.503400639927143 W_personnel 20.503400639927143 W_ps36 20.503400639927143 W_agutaya 20.628742020975263 W_magsaysay 20.628742020975263 W_http://t.co/xck… 23.007381555720713 W_http://t.co/… 24.277764461895703 W_http://t.co… 25.378220190135295

Appendix F: TFIDF Word Features (30%)

Caution and Advice		
W_@scph_pio 8.24117615049496	W_ramil 14.988365721796793	W_govt 20.058224314594536
W_kalmado 8.24117615049496	W_10kph 14.988365721796793	W_@deped_ph 20.058224314594536
W_teh 8.24117615049496	W_korte 14.988365721796793	W_tayo 20.058224314594536
W_kasalanan 8.24117615049496	W_masnamaspateros	W_project 20.058224314594536
W_lalawigang 8.24117615049496	14.988365721796793	W_mata 20.058224314594536
W_@mercadotrice 8.24117615049496	W_radius 14.988365721796793	W_posible 20.058224314594536
W_02:10 8.24117615049496	W_take 14.988365721796793	W_posibilidad 20.058224314594536
W_@ubeltmanila 8.24117615049496	W_lapad 14.988365721796793	W_@adamsonuni 20.058224314594536
W_pagalis 8.24117615049496	W_fresnedi 14.988365721796793	W_northwest 20.058224314594536
W_safeâ€? 8.24117615049496	W_lumapit 14.988365721796793	W_noah 20.058224314594536
W_#hiligaynon 8.24117615049496	W_thank 15.02235776098519	W_dito 20.11611192645208
W_cayetano 8.24117615049496	W_has 15.388948396738515	W_.. 20.121909449496254
W_mahinang 8.24117615049496	W_salamat 15.493632079992178	W_ulan 20.132378841533498
W_sama 8.24117615049496	W_list 16.0772879715977	W_camarines 20.1815870607581
W_paparating 8.24117615049496	W_catbalogan 16.323560398323025	W_hangin 20.18888165641876
W_...... 8.24117615049496	W_!!! 16.356507497847893	W_islands 20.192158042388535
W_d 8.24117615049496	W_your 16.356507497847893	W_advisory 20.192158042388535
W_k 8.24117615049496	W_nitong 16.356507497847893	W_calamian 20.201954243484558
W_@dextercruzat 8.24117615049496	W_mabalacat 16.356507497847893	W_calabarzon 20.201954243484558
W_#flashreport 8.24117615049496	W_@pasiginfo 16.356507497847893	W_2:30 20.205165103501887
W_r 8.24117615049496	W_sakop 16.356507497847893	W_@news5aksyon 20.22190894978084
W_heightened 8.24117615049496	W_@jonvicremulla	W_padua 20.2251101359746
W_w 8.24117615049496	16.356507497847893	W_school 20.2251101359746
W_@ycaycdc 8.24117615049496	W_almost 16.356507497847893	W_guys 20.2251101359746
W_@maddgil 8.24117615049496	W_pia 16.356507497847893	W_lumabas 20.2251101359746
W_nagkansela 8.24117615049496	W_@pdrrmcbulacan	W_pa-west 20.2251101359746
W_#philippines 8.24117615049496	16.356507497847893	W_bong 20.2251101359746
W_7:30 8.24117615049496	W_kalakasan 16.356507497847893	W_bay 20.2251101359746
W_akala 8.24117615049496	W_carcar 16.356507497847893	W_mag-landfall 20.2251101359746
W_beses 8.24117615049496	W_matinding 16.356507497847893	W_papalapit 20.2251101359746
W_relatives 8.24117615049496	W_umuulan 16.356507497847893	W_maynila 20.226781683122418
W_@jeyseeel 8.24117615049496	W_hay 16.356507497847893	W_rin 20.255728381572325
W_sand 8.24117615049496	W_umali 16.356507497847893	W_#hagupit 20.28395001937971
W_ofcs 8.24117615049496	W_itaas 16.356507497847893	W_malapit 20.28420994290172
W_#bataanph 8.24117615049496	W_#dobolbbalitangbalita	W_mayor 20.292302649436017
W_º 8.24117615049496	16.356507497847893	W_yellow 20.31580040293404
W_folks 8.24117615049496	W_yey 16.356507497847893	W_bagyo 20.332256528053765
W_patungo 8.24117615049496	W_compostela 16.356507497847893	W_ilang 20.34593589932558
W_ohshit 8.24117615049496	W_camiguin 16.356507497847893	W_bandang 20.386431556497865
W_@iskomorenooh 8.24117615049496	W_critical 16.356507497847893	W_a 20.386431556497865
W_mag-aral 8.24117615049496	W_lubao 16.356507497847893	W_pio 20.386431556497865
W_@xytollens 8.24117615049496	W_detalye 16.356507497847893	W_forecast 20.386431556497865
W_presyo 8.24117615049496	W_@ernie_manio 16.356507497847893	W_thursday 20.386431556497865
W_..!!! 8.24117615049496	W_pass 16.356507497847893	W_lungsod 20.386431556497865
W_@dahmercadooo 8.24117615049496	W_hoy 16.356507497847893	W_silangan 20.386431556497865
W_malapitan 8.24117615049496	W_??? 16.356507497847893	W_bugso 20.386431556497865
W_kagabi 8.24117615049496	W_paranaque 16.356507497847893	W_lumihis 20.386431556497865
W_tweet 8.24117615049496	W_could 16.356507497847893	W_dadaanan 20.386431556497865
W_paã±araque 8.24117615049496	W_valley 16.356507497847893	W_[20.399035244223743
W_128.5 8.24117615049496	W_daanan 16.356507497847893	W_] 20.399035244223743
W_dabarkads 8.24117615049496	W_11:05 16.356507497847893	W_gitna 20.399035244223743
W_there 8.24117615049496	W_exit 16.356507497847893	W_lumakas 20.399035244223743
W_tsktsk 8.24117615049496	W_guagua 16.356507497847893	W_gov 20.421433497914073
W_#southalerts 8.24117615049496	W_ayan 16.356507497847893	W_muling 20.448252183669062
W_named 8.24117615049496	W_12:00 16.356507497847893	W_iloilo 20.45483262800547
W_roro 8.24117615049496	W_mexico 16.356507497847893	W_pasukin 20.483301419780755
W_direksiyon 8.24117615049496	W_siya 16.356507497847893	W_including 20.503400639927143
W_muli 8.24117615049496	W_category 16.356507497847893	W_@feutamz 20.503400639927143
W_clc 8.24117615049496	W_natitirang 16.427105773014038	W_feu-nrmf 20.503400639927143
W_oscar 8.24117615049496	W_you 16.427105773014038	W_diliman 20.503400639927143
W_rappler 8.24117615049496	W_@dost_pagasa 16.747756225668212	W_high 20.503400639927143
W_santos-recto 8.24117615049496	W_namin 16.827976478180986	W_or 20.503400639927143
W_bula 8.24117615049496	W_kababayan 16.827976478180986	W_alon 20.51088619401681
W_ito'y 8.24117615049496	W_due 16.869577432501142	W_strong 20.51818027921944

Caution and Advice		
W_scph_pio 8.24117615049496	W_ating 16.869577432501142	W_:(20.568009808944428
W_orders 8.24117615049496	W_bataan 17.027043543709382	W_po 20.582952514262743
W_wait 8.24117615049496	W_ulat 17.038543960286876	W_suspend 20.592923596043672
W_fuck 8.24117615049496	W_tsk 17.200608882219456	W_#1 20.60135986030804
W_whyyyyy 8.24117615049496	W_@24orasgma 17.287942481590918	W_under 20.62652419672737
W_balitang 8.24117615049496	W_hall 17.302356611842104	W_storm 20.62769727974351
W_heading 8.24117615049496	W_30km 17.302356611842104	W_west 20.628742020975263
W_@racelisjerick 8.24117615049496	W_slowly 17.302356611842104	W_madaling 20.628742020975263
W_@radyopatrol39ito 8.24117615049496	W_nananatili 17.302356611842104	W_mag 20.628742020975263
W_tide 8.24117615049496	W_9am 17.302356611842104	W_bilis 20.628742020975263
W_@gmakf 8.24117615049496	W_juan 17.302356611842104	W_kasama 20.628742020975263
W_pips 8.24117615049496	W_yes 17.302356611842104	W_martes 20.640284498272617
W_@fillearlisfly 8.24117615049496	W_eastern-northern 17.302356611842104	W_group 20.673286596971366
W_@diimpeee 8.24117615049496	W_pa-kanluran 17.302356611842104	W_sana 20.69474517708961
W_piom 8.24117615049496	W_hanging 17.302356611842104	W_mula 20.701000339306514
W_ulet 8.24117615049496	W_daw 17.302356611842104	W_humina 20.7215969398293
W_#radyoinquirer 8.24117615049496	W_bagal 17.302356611842104	W_occ 20.7215969398293
W_tutok 8.24117615049496	W_@divinerey 17.302356611842104	W_daraan 20.726842987965018
W_we're 8.24117615049496	W_biglaang 17.302356611842104	W_2/2 20.726842987965018
W_@kimmymillo 8.24117615049496	W_tayong 17.302356611842104	W_feel 20.726842987965018
W_lani 8.24117615049496	W_5pm 17.302356611842104	W_effects 20.726842987965018
W_university 8.24117615049496	W_camnorte 17.302356611842104	W_sea 20.726842987965018
W_pedro 8.24117615049496	W_sadyang 17.302356611842104	W_tropical 20.726842987965018
W_makalawa 8.24117615049496	W_camotes 17.302356611842104	W_tagalog 20.726842987965018
W_cri 8.24117615049496	W_babala 17.302356611842104	W_malabon 20.726842987965018
W_bumubuhos 8.24117615049496	W_sinuspinde 17.302356611842104	W_@johnsonmanabat 20.726842987965018
W_possibility 8.24117615049496	W_felt 17.302356611842104	W_#2 20.746256217541404
W_islandâ€? 8.24117615049496	W_inc 17.302356611842104	W_@meralco 20.746256217541404
W_jusko 8.24117615049496	W_floods 17.302356611842104	W_now 20.748931744846068
W_#yellow 8.24117615049496	W_adm 17.302356611842104	W_news 20.749665284275313
W_hays 8.24117615049496	W_int'l 17.302356611842104	W_pang 20.77587811027188
W_pla 8.24117615049496	W_taglay 17.302356611842104	W_keep 20.793061475142146
W_@vicegandako 8.24117615049496	W_eto 17.302356611842104	W_safe 20.79622522650962
W_binayo 8.24117615049496	W_waves 17.302356611842104	W_occidental 20.801164039595655
W_plssss 8.24117615049496	W_11am 17.302356611842104	W_# 20.80223612393714
W_mag-roro 8.24117615049496	W_totoo 17.302356611842104	W_@dzrhnews 20.80223612393714
W_nilindol 8.24117615049496	W_parts 17.302356611842104	W_japan 20.80223612393714
W_6-beses 8.24117615049496	W_malalakas 17.302356611842104	W_provinces 20.80223612393714
W_#superbalita 8.24117615049496	W_alas- 17.302356611842104	W_track 20.80223612393714
W_naglabas 8.24117615049496	W_@valenzuelacity 17.302356611842104	W_handa 20.80223612393714
W_lee 8.24117615049496	W_winds 17.470667348306666	W_polillo 20.80223612393714
W_@russsselg 8.24117615049496	W_yan 17.801053391759794	W_1/4 20.80223612393714
W_mygaaaaadddd 8.24117615049496	W_island 17.80150292972814	W_semirara 20.80223612393714
W_http://t.co/is4â€¦ 8.24117615049496	W_@unanghirit 17.911231979274383	W_nag-landfall 20.80223612393714
W_lalangoy 8.24117615049496	W_bugsong 17.98591454978141	W_8-10pm 20.80223612393714
W_kona 8.24117615049496	W_for 17.98609544141216	W_bago 20.83499106660949
W_papalag 8.24117615049496	W_8pm 18.003748032113865	W_nueva 20.849393857813627
W_2/2)pero 8.24117615049496	W_live 18.003748032113865	W_ecija 20.849393857813627
W_kong 8.24117615049496	W_tonight 18.003748032113865	W_leyte 20.863705719981553
W_baterya 8.24117615049496	W_pray 18.003748032113865	W_borongan 20.86377006268562
W_i-charge 8.24117615049496	W_dumaan 18.003748032113865	W_violent 20.87911787925307
W_pnp 8.24117615049496	W_#breakingweathernow 18.003748032113865	W_circuit 20.87911787925307
W_yeaaaah 8.24117615049496	W_25% 18.003748032113865	W_patayin 20.87911787925307
W_okayyyy 8.24117615049496	W_robertmanodzmm 18.003748032113865	W_amerika 20.87911787925307
W_08dec 8.24117615049496	W_@mandaluyongc3 18.003748032113865	W_landfall 20.87911787925307
W_pasaway 8.24117615049496	W_dis 18.003748032113865	W_makaiwas 20.87911787925307
W_#quoteoftheday 8.24117615049496	W_pumalaot 18.003748032113865	W_miyerkules 20.87911787925307
W_lgu 8.24117615049496	W_abet 18.003748032113865	W_estrada 20.87911787925307
W_@joycee_legaspi 8.24117615049496	W_@lourddv 18.003748032113865	W_itinuturing 20.87911787925307
W_makatumba 8.24117615049496	W_mas 18.003748032113865	W_breaker 20.87911787925307
W_babagal 8.24117615049496	W_hour 18.003748032113865	W_intense 20.87911787925307
W_surigaâ€¦ 8.24117615049496	W_advisories 18.003748032113865	W_tumutok 20.87911787925307
W_yay 8.24117615049496		W_lalong 20.87911787925307
W_watch 8.24117615049496		W_aksidente 20.87911787925307
		W_guimaras 20.87911787925307
		W_camsur 20.89437535943061

Caution and Advice

W_katimugang 8.24117615049496
W_guysss 8.24117615049496
W_alas-syete 8.24117615049496
W_domestic 8.24117615049496
W_batan 8.24117615049496
W_anong 8.24117615049496
W_@jhayemsasis 8.24117615049496
W_lumayas 8.24117615049496
W_batay 8.24117615049496
W_http://t.co/ljiz43wo0vâ€? 8.24117615049496
W_campus 8.24117615049496
W_batch 8.24117615049496
W_umakyat 8.24117615049496
W_padin 8.24117615049496
W_umuungol 8.24117615049496
W_#laguna 8.24117615049496
W_probinsya 9.60577129421655
W_lubog 9.745318700964539
W_v 9.939444265371524
W_press 10.918918895442687
W_tumama 11.226722879586054
W_isang 11.605431000051457
W_muna 12.092519089920833
W_abangan 12.779923969230259
W_hala 12.779923969230259
W_abscbnnews 12.779923969230259
W_implikasyon 12.779923969230259
W_flight 12.779923969230259
W_karatig 12.779923969230259
W_idineklarang 12.779923969230259
W_b 12.779923969230259
W_g 12.779923969230259
W_p 12.779923969230259
W_blue 12.779923969230259
W_y 12.779923969230259
W_tuluyan 12.779923969230259
W_dumiretso 12.779923969230259
W_kalamado 12.779923969230259
W_ala-una 12.779923969230259
W_sobrang 12.779923969230259
W_@beabinene 12.779923969230259
W_tatawaging 12.779923969230259
W_umalis 12.779923969230259
W_@mmaarryyeell 12.779923969230259
W_schools 12.779923969230259
W_anak 12.779923969230259
W_alas-sais 12.779923969230259
W_southeast 12.779923969230259
W_retweet 12.779923969230259
W_340km 12.779923969230259
W_ncjr 12.779923969230259
W_naglandfall 12.779923969230259
W_6pm-8pm 12.779923969230259
W_maliban 12.779923969230259
W_#iamready 12.779923969230259
W_ahaha 12.779923969230259
W_datos 12.779923969230259
W_pagdating 12.779923969230259
W_magsuspended 12.779923969230259
W_tila 12.779923969230259
W_ancalerts 12.779923969230259
W_updates 12.779923969230259
W_netong 12.779923969230259
W_mindanao 12.779923969230259
W_hours 12.779923969230259

W_tsansa 18.003748032113865
W_stationary 18.003748032113865
W_13kph-15kph 18.003748032113865
W_garcia 18.003748032113865
W_bawal 18.003748032113865
W_9:15 18.003748032113865
W_binaklas 18.003748032113865
W_coding 18.003748032113865
W_kita 18.003748032113865
W_kumikilos 18.003748032113865
W_lalapitan 18.003748032113865
W_tolentino 18.003748032113865
W_wala 18.084299357859834
W_ngayong 18.114510881250915
W_maaaring 18.138888584974115
W_@dennis_datu 18.150363179415432
W_baha 18.288475358516312
W_#tvpatrol 18.501474107018684
W_tsansang 18.545288004643915
W_asahan 18.545288004643915
W_mmda 18.545288004643915
W_bahagya 18.545288004643915
W_east 18.545288004643915
W_mabagal 18.545288004643915
W_calixto 18.545288004643915
W_15kph 18.545288004643915
W_@akoposizandro 18.545288004643915
W_75% 18.545288004643915
W_buhos 18.545288004643915
W_3am 18.545288004643915
W_be 18.545288004643915
W_pinag-iingat 18.545288004643915
W_kasi 18.545288004643915
W_@nababaha 18.545288004643915
W_downgraded 18.545288004643915
W_romblon 18.557690969543565
W_i 18.607934569160065
W_god 18.607934569160065
W_medyo 18.607934569160065
W_saan 18.607934569160065
W_pagasa 18.666733183531655
W_s 18.67384305436798
W_sorsogon 18.822279518036765
W_nakapasok 18.97307777332978
W_dost_pagasa 18.97307777332978
W_torrijos 18.97307777332978
W_umaabot 18.97307777332978
W_izon 18.97307777332978
W_pala 18.97307777332978
W_pano 18.97307777332978
W_#superbalitasagabi 18.97307777332978
W_status 18.97307777332978
W_ramdam 18.97307777332978
W_11:00 18.97307777332978
W_5:00 18.97307777332978
W_aguilar 18.97307777332978
W_kahit 18.97307777332978
W_isa 18.97307777332978
W_rod 18.97307777332978
W_meters 18.97307777332978
W_@dzbbsamnielsen 18.97307777332978
W_rp12 18.97307777332978
W_omg 18.97307777332978
W_p.m. 18.98363668371316

W_area 20.906082434957888
W_island| 20.916306437528796
W_just 20.936332476674664
W_visayas 20.936332476674664
W_araw 20.94461771787911
W_gma 20.94461771787911
W_rains 20.94461771787911
W_@govramil 20.94461771787911
W_sibuyan 20.94461771787911
W_intensity 20.94461771787911
W_upang 20.94461771787911
W_number 20.94461771787911
W_magla-landfall 20.94461771787911
W_masbate 20.9487592939918
W_din 20.968274407933336
W_calauag 20.982040184373524
W_jma 21.00303050155502
W_tanghali 21.00303050155502
W_expect 21.00303050155502
W_kung 21.010157791316523
W_cam 21.010304623908105
W_may 21.025044612014316
W_pero 21.041990491852765
W_cebu 21.054296886778427
W_areas 21.058618433071288
W_supertyphoon 21.058618433071288
W_pasay 21.058618433071288
W_1/2 21.058618433071288
W_tatama 21.058618433071288
W_zambales 21.07700274158617
W_las 21.090468114278227
W_erap 21.090468114278227
W_@rapplerdotcom 21.090468114278227
W_makati 21.090468114278227
W_classes 21.09326326245795
W_naman 21.093616795951434
W_tandaan 21.09600545526976
W_lang 21.103225854742906
W_kph 21.106555863946255
W_ay 21.116261546322598
W_feu 21.124248883257323
W_itinaas 21.12848095747243
W_ruby 21.128643658565398
W_gabi 21.128643658565398
W_december 21.143444843529906
W_hanggang 21.15061145377966
W_jtwc 21.151794569238803
W_navotas 21.151794569238803
W_orange 21.151794569238803
W_ingat 21.151794569238803
W_kanina 21.151794569238803
W_habang 21.151794569238803
W_mamayang 21.151794569238803
W_phl 21.167280610137155
W_heavy 21.167280610137155
W_pasig 21.167280610137155
W_burias 21.167280610137155
W_m 21.187935293655332
W_hagupit 21.187935293655332
W_bahagyang 21.19141271925574
W_hindi 21.19141271925574
W_bumagal 21.19141271925574
W_muntinlupa 21.19141271925574
W_valenzuela 21.19141271925574
W_tuesday 21.19141271925574

Caution and Advice

W_hilaga 12.779923969230259
W_@israelmekaniko 12.779923969230259
W_kailan 12.779923969230259
W_rebecca 12.779923969230259
W_@vonneaquino 12.779923969230259
W_4:00 12.779923969230259
W_okay 12.779923969230259
W_lol 12.779923969230259
W_10:30 12.779923969230259
W_zone 12.779923969230259
W_papa 12.779923969230259
W_diperensya 12.779923969230259
W_lrt 12.779923969230259
W_naramdaman 12.779923969230259
W_ngang 12.779923969230259
W_namang 12.779923969230259
W_manakanakang 12.779923969230259
W_@melissy25 12.779923969230259
W_hahahaha 12.779923969230259
W_dec-09 12.779923969230259
W_9pm 12.779923969230259
W_diametro 12.779923969230259
W_ianunsyo 12.779923969230259
W_luv 12.779923969230259
W_ugh 12.779923969230259
W_dec08 12.779923969230259
W_!? 12.779923969230259
W_wednesday 12.779923969230259
W_nananatiling 12.779923969230259
W_wakas 12.779923969230259
W_03:00 12.779923969230259
W_@baaaaaaaaaaaae 12.779923969230259
W_piacentralviz 12.779923969230259
W_talagang 12.779923969230259
W_ltfrb 12.779923969230259
W_finally 12.779923969230259
W_kalangitan 12.779923969230259
W_suarez 12.779923969230259
W_video 13.17395924837424
W_â?¤ 13.953524163708405
W_http://â€¦ 13.953524163708405
W_teleradyo 14.384632589319974
W_dapat 14.384632589319974
W_-_- 14.384632589319974
W_mamaya 14.419500657334172
W_#aksyonsahagupit 14.450498753706825
W_@allangatus 14.618381241164112
W_@dzmmteleradyo 14.683886862001419
W_@zhandercayabyab 14.804184822851791
W_#fb 14.988365721796793
W_+ 14.988365721796793
W_!!!! 14.988365721796793
W_teritoryo 14.988365721796793
W_entering 14.988365721796793
W_u 14.988365721796793
W_pag 14.988365721796793
W_@rida_reyes 14.988365721796793
W_officialmunti 14.988365721796793
W_with 14.988365721796793
W_08dec14 14.988365721796793
W_#imready 18.98363668371316
W_epekto 18.98363668371316
W_oriental 19.00216640961615
W_suspendido 19.048672358429886
W_e 19.055679147282532
W_klase 19.126095902383177
W_... 19.197687547237337
W_lunes 19.260486700138248
W_talisay 19.262639720301475
W_everyone 19.27735831057902
W_talaga 19.27735831057902
W_ko 19.27735831057902
W_halos 19.289216071161423
W_yung 19.289216071161423
W_says 19.319654207668115
W_latest 19.319654207668115
W_caloocan 19.319654207668115
W_kanluran 19.319654207668115
W_light 19.319654207668115
W_#news 19.319654207668115
W_-- 19.319654207668115
W_taya 19.319654207668115
W_herbert 19.319654207668115
W_moderate 19.319654207668115
W_stay 19.319654207668115
W_nito 19.319654207668115
W_ano 19.319654207668115
W_ahensya 19.319654207668115
W_@ukgdos 19.319654207668115
W_bautista 19.319654207668115
W_alert 19.319654207668115
W_ka 19.319654207668115
W_alas-diyes 19.319654207668115
W_weather 19.319654207668115
W_12nn 19.319654207668115
W_& 19.347488640266004
W_to 19.473838825479692
W_cuyo 19.49467023127168
W_update 19.593023663635186
W_@hadjirieta 19.60921856001766
W_maramdaman 19.60921856001766
W_hapon 19.60921856001766
W_del 19.60921856001766
W_natin 19.60921856001766
W_raw 19.60921856001766
W_ncr 19.60921856001766
W_tomorrow 19.60921856001766
W_raised 19.60921856001766
W_tues 19.60921856001766
W_#hiritpanahon 19.60921856001766
W_gov't 19.60921856001766
W_@peeweehero 19.627237435130677
W_ngayon 19.630422562536587
W_walang 19.65221207276957
W_eastern 19.656168085770492
W_cuã±a 19.66587217692185
W_isâ€¦ 19.66587217692185
W_https://t.â€¦ 19.66587217692185
W_@abscbnnews 19.68440176194164
W_ayon 19.717386357239445
W_ba 19.771001980162918
W_para 19.776658454442288
W_nasa 19.81880684836712
W_sapat 19.845888677698312
W_suspended 19.845888677698312
W_pls 19.845888677698312
W_tulad 19.845888677698312
W_aklan 21.19246702244009
W_pasok 21.19567498448848
W_tubig 21.197867798817075
W_lakas 21.202121554772337
W_tom 21.207600939352925
W_hernandez 21.207600939352925
W_sabado 21.215616318655638
W_ito 21.226898551578344
W_#3 21.22848969765599
W_layong 21.22848969765599
W_n.u. 21.22848969765599
W_km 21.22848969765599
W_kalupaan 21.249115533666846
W_surge 21.253475972793122
W_capiz 21.284458796046668
W_per 21.285431265854903
W_@iamsumulong 21.285431265854903
W_ticao 21.285431265854903
W_luzon 21.29635213581166
W_posibleng 21.29635213581166
W_par 21.297440906044358
W_n 21.300967842424626
W_5am 21.300967842424626
W_catanduanes 21.300967842424626
W_loob 21.300967842424626
W_lalawigan 21.300967842424626
W_inaasahang 21.302961558568867
W_|via 21.310442160607867
W_responsibility 21.310442160607867
W_warning 21.310442160607867
W_on 21.310442160607867
W_@radyopatrol39 21.313776328320976
W_): 21.31749655909268
W_paalala 21.31749655909268
W_ninyo 21.31749655909268
W_sumusunod 21.31749655909268
W_antique 21.31749655909268
W_aming 21.318611563181395
W_serbisyo 21.318611563181395
W_numero 21.318611563181395
W_kailanganin 21.318611563181395
W_pag-akyat 21.318611563181395
W_grp 21.325369990264992
W_suspends 21.325369990264992
W_negros 21.325369990264992
W_tarlac 21.325369990264992
W_@rizalgov 21.325369990264992
W_monday 21.328737409119363
W_lalabas 21.328737409119363
W_umaga 21.331515403770837
W_huling 21.331515403770837
W_@robertmanodzmm 21.331515403770837
W_11pm 21.331515403770837
W_namataan 21.331515403770837
W_@iskomoreno 21.3344466614766
W_mamayaâ€¦ 21.504837490149264
W_http://t.â€¦ 21.504837490149264
W_queâ€¦ 21.504837490149264
W_http://t.coâ€¦ 23.007381555720713
W_http://t.co/ekfowâ€¦ 23.007381555720713
W_piã 25.378220190135295
W_paraã±aque 25.378220190135295

Caution and Advice		
W_naka-red 14.988365721796793 W_2/4 14.988365721796793 W_surigao 14.988365721796793 W_paskuhan 14.988365721796793 W_@sherieanntorres 14.988365721796793 W_calapan 14.988365721796793 W_buti 14.988365721796793 W_nagbago 14.988365721796793 W_quiz 14.988365721796793 W_bumaba 14.988365721796793 W_makulimlim 14.988365721796793 W_updated 14.988365721796793 W_early 14.988365721796793 W_700km 14.988365721796793 W_pagdaan 14.988365721796793 W_magdasal 14.988365721796793 W_dadaan 14.988365721796793 W_#prayforph 14.988365721796793	W_pagkain 19.845888677698312 W_coastal 19.845888677698312 W_!! 19.845888677698312 W_hilagang-kanluran 19.845888677698312 W_pumasok 19.845888677698312 W_delata 19.845888677698312 W_imbak 19.845888677698312 W_delikado 19.845888677698312 W_alas-otso 19.845888677698312 W_lalo 19.898584335374846 W_pm 19.8992613061474 W_is 19.899840905150437 W_rainfall 19.930934002176514 W_norte 19.940424095313666 W_palawan 19.944989480033932 W_offices 20.025667136521548 W_the 20.050064374556886 W_pagitan 20.058224314594536	W_â€? 26.34889093451958 W_http://t.co/c61â€¦ 28.002653484065497 W_camarinesâ€¦ 28.71972956893416 W_piã±as 28.71972956893416 W_câ€¦ 29.990112475109143 W_http://t.co/i5ibubfnâ€¦ 30.558694882161575 W_http://t.co/lonurxpgâ€¦ 30.558694882161575 W_vâ€¦ 36.271042895375025 W_http://t.co/lkvkotzxâ€¦ 39.043969867121454 W_â€œ 39.237888841187626 W_http://tâ€¦ 41.56067383420147 W_http://t.câ€¦ 42.11961179761311 W_romblâ€¦ 42.515264229775624 W_http://t.co/â€¦ 44.03924179546624

Casualty and Damage		
W_winalis 8.24117615049496 W_nagkalat 8.24117615049496 W_magkaiba 8.24117615049496 W_nawasak 8.24117615049496 W_iniligpit 8.24117615049496 W_inasuyan 8.24117615049496 W_radyopatrol44 8.24117615049496 W_barangays 8.24117615049496 W_nddrmc 8.24117615049496 W_cut-off 8.24117615049496 W_@pro5react 8.24117615049496 W_gikan 8.24117615049496 W_@katherineimson 8.24117615049496 W_subalit 8.24117615049496 W_dpwh 8.24117615049496 W_june 8.24117615049496 W_hypothermia 8.24117615049496 W_@radyopatrol44 8.24117615049496 W_magsilikas 8.24117615049496 W_@camsurppo 8.24117615049496 W_-hernani 8.24117615049496 W_isinara 8.24117615049496 W_lates 8.24117615049496 W_tala 8.24117615049496 W_passable 8.24117615049496 W_@pnppio 8.24117615049496 W_under 8.634824463502863 W_din 9.600544676592113 W_dzmm 9.70923927319281 W_s 9.889065954842735 W_gov 10.046721120369453 W_di 10.053122685107908 W_red 10.053122685107908 W_@peeweehero 10.473788508543128 W_@ancalerts 10.503038540285301 W_you 10.658811881212419 W_natitirang 10.658811881212419 W_mayor 11.214085662642553 W_e 11.475338207362467 W_as 11.989490039041598 W_culaba 12.092519089920833 W_bugsong 12.25724235587069 W_ilang 12.358815792052159	W_sherry 12.779923969230259 W_ozaeta 12.779923969230259 W_winasak 12.779923969230259 W_cross 12.779923969230259 W_compound 12.779923969230259 W_@victorcosare777 12.779923969230259 W_arnel 12.779923969230259 W_vendor 12.779923969230259 W_kisame 12.779923969230259 W_rail 12.779923969230259 W_front 12.779923969230259 W_20-anyos 12.779923969230259 W_pasahero 12.779923969230259 W_list 12.929985398341222 W_ngayon 12.986477707562292 W_thank 13.211322591055797 W_ayon 13.584670498681378 W_talisay 13.812550291706431 W_legazpi 13.91523363537655 W_http 14.16870916513339 W_estancia 14.384632589319974 W_nanatili 14.384632589319974 W_#aksyon 14.419500657334172 W_buong 14.557634565767035 W_ulan 14.786785819809174 W_@zhandercayabyab 14.804184822851791 W_baylon 14.988365721796793 W_bubong 14.988365721796793 W_ajuy 14.988365721796793 W_utos 14.988365721796793 W_years 14.988365721796793 W_establisimyento 14.988365721796793 W_nasaktan 14.988365721796793 W_old 14.988365721796793 W_pinapakuan 14.988365721796793 W_agbobolo 14.988365721796793 W_@arnellozaeta 14.988365721796793 W_causalities 14.988365721796793 W_9,584 14.988365721796793 W_ernesto 14.988365721796793	W_tan 16.356507497847893 W_romblon 16.490924059619076 W_kuryente 16.50520946752559 W_tacloban 16.58340940659383 W_by 16.827976478180986 W_puno 16.827976478180986 W_@akosijaysent 16.923884280941536 W_northeastern 17.302356611842104 W_samar- 17.302356611842104 W_downed 17.302356611842104 W_cayabyab 17.302356611842104 W_@dsm_sunstar 17.302356611842104 W_electrical 17.302356611842104 W_@edlingao 17.302356611842104 W_abucay 17.302356611842104 W_nkklk 17.302356611842104 W_topples 17.302356611842104 W_uy-tan 17.302356611842104 W_zhander 17.302356611842104 W_daanbantayan 17.302356611842104 W_6:00 17.302356611842104 W_electric 17.302356611842104 W_isinasagawa 17.302356611842104 W_evacuation 17.48210799446251 W_photo 17.535843167999637 W_lumikas 17.98609544141216 W_#cebu 18.003748032113865 W_ann 18.003748032113865 W_barangay 18.284295645158867 W_nakatira 18.284295645158867 W_pswd 18.545288004643915 W_imprastraktura 18.545288004643915 W_total 18.545288004643915 W_iniwan 18.545288004643915 W_bagsak 18.545288004643915 W_inilikas 18.607934569160065 W_mahigit 18.84348886290733 W_ulat 18.93698480446941 W_dagat 18.97307777332978 W_taclobanon 18.97307777332978 W_sapilitan 18.97307777332978 W_pinalilikas 18.97307777332978

Casualty and Damage		
W_strong 12.358815792052159	W_walang 15.108372282361477	W_komunikasyon 18.97307777332978
W_bayan 12.377099015883529	W_#radyopatrol 15.388948396738515	W_evacuees 18.97307777332978
W_bagyo 12.623746077823911	W_matapos 15.388948396738515	W_linya 18.97307777332978
W_galera 12.779923969230259	W_odiongan 15.388948396738515	W_pamilya 19.00882843370869
W_naibalik 12.779923969230259	W_press 15.388948396738515	W_catbalogan 19.195581931273235
W_airport 12.779923969230259	W_eastern 15.468282816699496	W_residente 19.262639720301475
W_#votynews 12.779923969230259	W_katao 15.822762373529738	W_probinsya 19.27735831057902
W_dead 12.779923969230259	W_preemptive 15.822762373529738	W_naitalang 19.319654207668115
W_herrera 12.779923969230259	W_brgy 15.822762373529738	W_#aksyonsahagupit 19.348557039594567
W_ormoc 12.779923969230259	W_@gepelyle 15.822762373529738	W_30,689 19.60921856001766
W_good 12.779923969230259	W_higit 15.822762373529738	W_post 19.60921856001766
W_nagpapatupad 12.779923969230259	W_nang 16.041593592943087	W_malawak 19.82734315459205
W_correspondent 12.779923969230259	W_baha 16.119947282327892	W_casualty 20.058224314594536
W_issue 12.779923969230259	W_calamity 16.356507497847893	W_pananalasa 20.09273460429346
W_itinumba 12.779923969230259	W_dilg 16.356507497847893	W_video 20.25421585957266
W_nagsitumbahan 12.779923969230259	W_28,000 16.356507497847893	W_dalawa 20.628742020975263
W_@amberpgonzales 12.779923969230259	W_plywood 16.356507497847893	W_nagdulot 20.646448051992703
W_poste 12.779923969230259	W_state 16.356507497847893	W_nasawi 20.726842987965018
	W_camalig 16.356507497847893	W_pinsala 20.86377006268562
	W_@sunstarcebu 16.356507497847893	
	W_tinangay 16.356507497847893	

Call For Help		
W_mauubusan 8.24117615049496	W_tumulong 12.779923969230259	W_@allangatus 16.503338690643297
W_pre-emptive 8.24117615049496	W_nagsisiksikang 12.779923969230259	W_@24orasgma 16.651069376032837
W_inirekumenda 8.24117615049496	W_supply 12.779923969230259	W_tacloban 16.813880662710172
W_near 8.24117615049496	W_complex 12.779923969230259	W_lubog 16.95589940235178
W_hotline 8.24117615049496	W_evacuate 12.779923969230259	W_evacuation 17.126091736653525
W_patag 8.24117615049496	W_delpan 12.779923969230259	W_kailangang 17.200608882219456
W_hashtag 8.24117615049496	W_hospital 12.779923969230259	W_highway 17.206005535451542
W_19,000 8.24117615049496	W_sports 12.779923969230259	W_si… 17.295033542507273
W_river 8.24117615049496	W_tumama 13.91523363537655	W_center 17.490647149520154
W_calumpang 8.24117615049496	W_lumikas 14.777633867935151	W_binuksan 18.003748032113865
W_namn 8.24117615049496	W_nagsimula 14.988365721796793	W_village 18.003748032113865
W_paglilista 8.24117615049496	W_taga-eastern 14.988365721796793	W_kapuso 18.150363179415432
W_offices 8.715465760364589	W_magkumpuni 14.988365721796793	W_kuryente 18.924580994206224
W_@zhandercayabyab 9.60577129421655	W_kanilang 14.988365721796793	W_tablas 18.97307777332978
W_maapektuhan 9.60577129421655	W_banta 15.023748268553774	W_8:30 18.97307777332978
W_as 9.786630986868186	W_photo 15.108372282361477	W_bilang 19.055679147282532
W_lugar 10.370591514291535	W_buong 15.217498960310474	W_kapilya 19.557409258992998
W_narito 11.226722879586054	W_nang 15.68141664291429	W_ipinagagamit 19.557409258992998
W_:/ 11.226722879586054	W_regional 16.356507497847893	W_simbahan 19.557409258992998
W_bayan 11.404210347066746		W_@_iancruz 20.058224314594536
W_nakatutok 12.25724235587069		W_gumaca 20.058224314594536

Donation		
W_tumatanggap 8.24117615049496	W_calls 8.24117615049496	W_free 14.988365721796793
W_offer 8.24117615049496	W_grocery 8.24117615049496	W_for 16.300422770030902
W_bangus 8.24117615049496	W_naka 8.24117615049496	W_food 17.200608882219456
W_supermalls 8.24117615049496	W_ngiti 8.24117615049496	W_@dswdserves 17.302356611842104
W_mgs 8.24117615049496	W_#9newsph 8.24117615049496	W_#reliefph 17.302356611842104
W_mangudadatu 8.24117615049496	W_offers 8.24117615049496	W_family 17.574868470828566
W_victims 8.24117615049496	W_biktima 8.24117615049496	W_mateo 18.003748032113865
W_ginawa 8.24117615049496	W_offices 8.715465760364589	W_preso 18.003748032113865
W_#dswd 8.24117615049496	W_the 9.155137639651025	W_@vargasmannysen 18.003748032113865
W_benloi 8.24117615049496	W_#unanghirit 10.149480641391612	W_ilikas 18.003748032113865
W_9-10 8.24117615049496	W_has 10.918918895442687	W_packs 18.003748032113865
W_australia 8.24117615049496	W_to 12.390056101795334	W_navy 18.545288004643915
W_countries 8.24117615049496	W_volunteers 12.779923969230259	W_towns 18.97307777332978
W_court 8.24117615049496	W_sampaloc 12.779923969230259	W_inihahanda 19.60921856001766
W_8am-6pm 8.24117615049496	W_@anakbayan_ph 12.779923969230259	W_relief 20.275841505975105
W_globe 8.24117615049496		

Donation		
W_handang 8.24117615049496 W_sagip 8.24117615049496 W_providing 8.24117615049496 W_donasyon 8.24117615049496 W_sm 8.24117615049496 W_hinihikayat 8.24117615049496 W_maaari 8.24117615049496 W_@enjoyglobe 8.24117615049496	W_dswd 12.779923969230259 W_point 12.779923969230259 W_st. 12.779923969230259 W_maapektuhan 13.538219321261609 W_http://t.co/oq8rw7jw… 13.953524163708405 W_salamat 14.16870916513339 W_san 14.237600391676462 W_200,000 14.268563439747771 W_photo 14.626378374680842	W_goods 20.399035244223743 W_transported 20.503400639927143 W_personnel 20.503400639927143 W_ps36 20.503400639927143 W_agutaya 20.628742020975263 W_magsaysay 20.628742020975263 W_http://t.co/xck… 23.007381555720713 W_http://t.co/… 24.277764461895703 W_http://t.co… 25.378220190135295

Appendix G: Extraction Rules

Caution and Advice
<pos:JJ> <pos:NN> <pos:PSNS> <number:ANY>
<ner:LOCATION>[as]LOCATION
<pos:JJ> <string:#1>
<pos:JJ> <string:#2>
<pos:JJ> <string:#3>
<pos:VBZ> <string:classes> <pos:IN> <pos:JJ> <pos:VBZ>
<pos:VBP> <string:classes> <pos:IN> <pos:JJ> <pos:VBZ>
<string:#walangpasok> <pos:JJ> <pos:VBZ>
<string:signal> <pos:NN> <pos:PSNS> <number:ANY>
<string:#walangpasok> <pos:PSNS> <string:klase>
<string:#walangpasok> <string:sa> <pos:PIDP> <pos:NA> <string:antas>

Casualty and Damage
<number:ANY>[as]NUMBER <pos:NA> <pos:NA> <string:ANY> <ner:UNIT>[as]UNIT
<pos:NA> <pos:NCOM> <pos:NA> <pos:NCOM>
<number:ANY>[as]NUMBER <ner:UNIT>[as]UNIT
<ner:LOCATION>[as]LOCATION
<pos:NCOM> <pos:NA> <pos:NCOM>[as]OBJECT
<string:sa> <pos:PINP> <pos:NN>
<string:sa> <pos:JJ> <ner:LOCATION>[as]LOCATION
<string:state>[as]DETAIL <string:of>[as]DETAIL <string:calamity>[as]DETAIL
<pos:JJ> <ner:LOCATION>[as]LOCATION
<ner:LOCATION>[as]LOCATION <pos:NN>[as]LOCATION
<pos:VOBF>[as]DETAIL <pos:NA> <string:#RubyPH>[as]DETAIL
<pos:NA> <pos:NCOM>[as]OBJECT
<pos:ADUN>[as]DETAIL <pos:NCOM> <pos:NCOM>[as]DETAIL
<pos:NA> <pos:NN>[as]OBJECT
<pos:ADOT> <pos:NCOM>
<number:ANY>[as]DETAIL <string:na> <string:ang> <pos:JJ> <ner:UNIT>[as]OBJECT
<number:ANY>[as]DETAIL <pos:NNS>[as]UNIT
<pos:NCOM>[as]DETAIL <string:at>[as]DETAIL <pos:NCOM>[as]DETAIL
<string:walang>[as]DETAIL <string:naitalang> <ner:UNIT>[as]UNIT

Call For Help
<pos:MANH> <pos:CONG> <pos:NCOM>

Donation
<ner:LOCATION>[as]LOCATION
<pos:VOBF> <pos:NA> <pos:NA> <pos:NN>[as]DONATION
<number:ANY> <ner:UNIT>[as]UNIT
<number:ANY> <pos:NN:UN> <ner:UNIT>[as]UNIT <string:packs>

Appendix H: Representation of Ontology in OWL Format

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#"
     xml:base="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
    <owl:Ontology rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5"/>

    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Object Properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->

    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#can_be_of_the_category -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#can_be_of_the_category">
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CallForHelp"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CasualtiesAndDamage"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CautionAndAdvice"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Donation"/>
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#donates -
->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#donates">
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Volunteer"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#from_a --
>

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#from_a">
```

```
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#gives_out_an -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#gives_out_an">
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Advice"/>
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CautionAndAdvice"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#has_this_information -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#has_this_information">
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Location"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#needs -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#needs">
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_a_call_for_help_from_a -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_a_call_for_help_from_a">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CallForHelp"/>
```

```
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_a_need_for_a -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_a_need_for_a">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Donation"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Volunteer"/>
    </owl:ObjectProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_damages_to -->

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#reports_damages_to">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CasualtiesAndDamage"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Object"/>
        <rdfs:range
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
    </owl:ObjectProperty>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Data properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#geoLocationOfTweet -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#geoLocationOfTweet">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Location"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#locationInTweet -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#locationInTweet">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Location"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#objectDetails -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#objectDetails">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Object"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#objectName -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#objectName">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Object"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#resourceDetails -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#resourceDetails">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#resourceName -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#resourceName">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetAdvice -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetAdvice">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Advice"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetContent -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetContent">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#tweetDate
-->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetDate">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetHandle -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetHandle">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetTimestamp -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#tweetTimestamp">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#victimName -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#victimName">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#volunteerName -->

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#volunteerName">
        <rdfs:domain
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Volunteer"/>
        <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    </owl:DatatypeProperty>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Classes
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Advice --
>

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Advice"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CallForHelp -->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#CallForHelp">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:Class>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CasualtiesAndDamage -->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#CasualtiesAndDamage">
```

```
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:Class>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CautionAndAdvice -->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#CautionAndAdvice">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:Class>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Donation
-->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Donation">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
    </owl:Class>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Location
-->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Location"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Object --
>

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Object"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Resource
-->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Resource"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Timestamp
-->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Timestamp"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Tweet -->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Tweet"/>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Victim --
>

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Victim"/>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Volunteer
-->

    <owl:Class rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-
ontology-5#Volunteer"/>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Individuals
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CD-
I_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CD-
I_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CasualtiesAndDamage"/>
        <reports_damages_to
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_OBJ-I_ERV_Elem_School"/>
        <reports_damages_to
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_VIC-I_ERV_Elem_School"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CFH-
I_pamilya_ng_mga_sundalo -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CFH-
I_pamilya_ng_mga_sundalo">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CallForHelp"/>
        <reports_a_call_for_help_from_a
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_VIC-I_pamilya_ng_mga_sundalo"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_D-
I_students_of_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_D-
I_students_of_ERV_Elem_School">
```

```
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Donation"/>
        <reports_a_need_for_a
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_RES-I_students_of_ERV_Elem_School"/>
    &lt;/owl:NamedIndividual&gt;



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_OBJ-
I_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_OBJ-
I_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Object"/>
        &lt;objectDetails rdf:datatype="http://www.w3.org/2001/XMLSchema#string"&gt;Binaha ang
classrooms.&lt;/objectDetails&gt;
        &lt;objectName rdf:datatype="http://www.w3.org/2001/XMLSchema#string"&gt;ERV Elem School&lt;/objectName&gt;
    &lt;/owl:NamedIndividual&gt;



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_RES-
I_students_of_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_RES-
I_students_of_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Resource"/>
        &lt;resourceDetails rdf:datatype="http://www.w3.org/2001/XMLSchema#string"&gt;1000
pcs&lt;/resourceDetails&gt;
        &lt;resourceName rdf:datatype="http://www.w3.org/2001/XMLSchema#string"&gt;notebooks and
bags&lt;/resourceName&gt;
        <from_a
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_VIC-I_students_of_ERV_Elem_School"/>
    &lt;/owl:NamedIndividual&gt;



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
        &lt;victimName rdf:datatype="http://www.w3.org/2001/XMLSchema#string"&gt;Mga estudyante ng ERV Elem
School&lt;/victimName&gt;
    &lt;/owl:NamedIndividual&gt;



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_pamilya_ng_mga_sundalo -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_pamilya_ng_mga_sundalo">
```

```
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
        <victimName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">pamilya ng mga
sundalo</victimName>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_students_of_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_VIC-
I_students_of_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Victim"/>
        <victimName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">students of ERV Elem
School</victimName>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_A-
I_WARNING!_Ito_ay_isang_advice_na_tweet! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_A-
I_WARNING!_Ito_ay_isang_advice_na_tweet!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Advice"/>
        <tweetAdvice rdf:datatype="http://www.w3.org/2001/XMLSchema#string">WARNING! Ito ay isang
advice na tweet!</tweetAdvice>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CA-
I_WARNING!_Ito_ay_isang_advice_na_tweet! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CA-
I_WARNING!_Ito_ay_isang_advice_na_tweet!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#CautionAndAdvice"/>
        <gives_out_an
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_A-
I_WARNING!_Ito_ay_isang_advice_na_tweet!"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Location"/>
        <geoLocationOfTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">418.582912321,
102.51084911</geoLocationOfTweet>
        <locationInTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Brgy.
Trese</locationInTweet>
    </owl:NamedIndividual>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Location"/>
        <geoLocationOfTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">10.00000121, 145.345300023</geoLocationOfTweet>
        <locationInTweet
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Guadalupe</locationInTweet>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Location"/>
        <geoLocationOfTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">9.011112321, 231.34569903</geoLocationOfTweet>
        <locationInTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Makati City</locationInTweet>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_LI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Location"/>
        <geoLocationOfTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">109.00540121, 45.378000003</geoLocationOfTweet>
        <locationInTweet rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Quezon City</locationInTweet>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_TI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_TI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Tweet"/>
        <tweetContent rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Ang mga pamilya ng mga sundalo sa Brgy. Trese ay nangangailangan ng tulong!</tweetContent>
        <tweetHandle rdf:datatype="http://www.w3.org/2001/XMLSchema#string">TheJPGarcia</tweetHandle>
        <can_be_of_the_category
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CFH-I_pamilya_ng_mga_sundalo"/>
```

```xml
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong!"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong!"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
        <tweetContent rdf:datatype="http://www.w3.org/2001/XMLSchema#string">BWAHAHAHAHAHA RT WARNING!
Ito ay isang tweet!</tweetContent>
        <tweetHandle
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">theonlykyleeeee</tweetHandle>
        <can_be_of_the_category
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CA-
I_WARNING!_Ito_ay_isang_advice_na_tweet!"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet!"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet!"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
        <tweetContent rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Hi everyone! The students
of ERV Elem School are in need of 1000 pcs of notebooks and bags!</tweetContent>
        <tweetHandle
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">vilsonluuufffy</tweetHandle>
        <can_be_of_the_category
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_D-
I_students_of_ERV_Elem_School"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags!"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags!"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School">
```

```
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Tweet"/>
        <tweetContent rdf:datatype="http://www.w3.org/2001/XMLSchema#string">PRAY! RT Binaha ang
classrooms ng mga estudyante ng ERV Elem School</tweetContent>
        <tweetHandle
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">addicteduserrr</tweetHandle>
        <can_be_of_the_category
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_CD-
I_ERV_Elem_School"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_LI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School"/>
        <has_this_information
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School"/>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Ang_mga_pamilya_ng_mga_sundalo_sa_Brgy._Trese_ay_nangangailangan_ng_tulong!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <tweetTimestamp
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">03/21/2015:03:27:00:34</tweetTimestamp>
        <tweetDate rdf:datatype="http://www.w3.org/2001/XMLSchema#string">March 21, 2015</tweetDate>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_BWAHAHAHAHAHA_RT_WARNING!_Ito_ay_isang_tweet!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <tweetTimestamp
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">12/27/2014:00:13:67:40</tweetTimestamp>
        <tweetDate rdf:datatype="http://www.w3.org/2001/XMLSchema#string">December 27, 2014</tweetDate>
    </owl:NamedIndividual>



    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags! -->

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#_TSI_Hi_everyone!_The_students_of_ERV_Elem_School_are_in_need_of_1000_pcs_of_notebooks_and_bags!">
        <rdf:type
rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-
5#Timestamp"/>
        <tweetTimestamp
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">12/30/2014:01:44:09:10</tweetTimestamp>
        <tweetDate rdf:datatype="http://www.w3.org/2001/XMLSchema#string">December 30, 2014</tweetDate>
    </owl:NamedIndividual>
```

```
    <!-- http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_TSI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School -->

    <owl:NamedIndividual rdf:about="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#_TSI_PRAY!_RT_Binaha_ang_classrooms_ng_mga_estudyante_ng_ERV_Elem_School">
        <rdf:type rdf:resource="http://www.semanticweb.org/kylemchalebdelacruz/ontologies/2014/9/untitled-ontology-5#Timestamp"/>
        <tweetTimestamp rdf:datatype="http://www.w3.org/2001/XMLSchema#string">12/27/2014:00:14:47:24</tweetTimestamp>
        <tweetDate rdf:datatype="http://www.w3.org/2001/XMLSchema#string">December 27, 2014</tweetDate>
    </owl:NamedIndividual>
</rdf:RDF>



<!-- Generated by the OWL API (version 3.5.0) http://owlapi.sourceforge.net -->
```

Appendix I: Resource Persons

Nicco Louis S. Nocon
NormAPI Proponent
noconoccin@gmail.com

Nathaniel Oco
Faculty Member
College of Computer Studies
National University
nathanoco@yahoo.com

Ralph Vincent J. Regalado
Thesis Adviser, Faculty Member
Software Technology Department
College of Computer Studies
De La Salle University
ralph.regalado@delasalle.ph

Appendix J: Personal Vitae

Kyle Mc Hale B. Dela Cruz
Blk 5, Lot 2A, Martires St., Brgy. Martires del 96, Pateros, Metro Manila
(0917) 880-5019
kylemchale_delacruz@yahoo.com

John Paul F. Garcia
36 Bohol St., Ayala Alabang Village, Muntinlupa City
(0927) 886-6999
johnpaulgarcia1208@gmail.com

Kristine Ma. Dominique F. Kalaw
28 New Years Avenue, GSIS Holiday Hills Village, San Pedro, Laguna
(0927) 854-4201
tintin.kalaw@gmail.com

Vilson E. Lu
739 – D A. Bonifacio St. Balintawak, Quezon City
(0917) 631-1374
vilson.espayos.lu@gmail.com