Maternal Medical Information Extraction (MaMIE) System

Allan Borra, Karla Delos Santos, Daniel Gonzales, Alden Reyes
College of Computer Studies, De La Salle University
borgz.borra@delasalle.ph, krldelossantos@ymail.com, dcndgonzales@yahoo.com, roaldaldenreyes@gmail.com

ABSTRACT
This paper discusses the development of an information extraction system for maternal health records. Current investigated maternal health records that exist in hospitals are written in a long and detailed paragraphs in text documents. Information extraction processes unstructured, natural language text, such as maternal health records, to extract useful information from the text. MaMIE (Maternal Medical Information Extraction) System is designed to extract information from medical narratives obtained from De La Salle University Medical Center (DLSUMC) in Dasmarinas, Cavite. MaMIE System is composed of seven modules namely Text Zoner, Sentence Splitter, POS Tagger, Named-Entity Recognizer, Preparser, Coreferencer and Semantic Interpreter. An Obstetric Gynecologist from DLSUMC defined the fields of the template. The system was tested over 12 documents with the results of 87.39% Precision, 86.74% Recall and 87.06% F-measure.

Keywords
Information Extraction, Maternal Health, Medical Natural Language Processing.

1. INTRODUCTION
Maternal health denotes to the physical state of women during their pregnancy, childbirth and the postpartum period [1]. It deals with the health care aspects of family planning, preconception, prenatal and postnatal care. In the Philippines, there is an alarming high Maternal Mortality Rate. Over 11 women die everyday due to pregnancy-related complications. Despite the increase in health-related spending and the devolution of health services, the Philippine MMR continues to be at an unacceptable level [1].

The current maternal health challenges call for innovative and creative solutions. New technology-based solutions are created to address the wide-ranging issues pertaining to maternal health. Information Extraction (IE) is one technology that can be used to aid in providing solution to maternal health issues. IE is a type of information retrieval whose objective is to automatically extract structured information and or semi-structured machine-readable documents [2].

Currently, there is no known research done for information extraction under the domain of maternal health in the Philippines. It is known that hospitals in the Philippines are written in a long and detailed manner, which prolongs reading of these documents. Patient records written by health workers, describe the long term course of patient’s illnesses and treatments and contain information vital to immediate patient care. However, the long and detailed textual contents of these records make it difficult to survey even a single patient’s complete record. Extracting useful information can result to reliable data on the levels and causes of maternal mortality. Such reliable data can be used for planning, monitoring and evaluating maternal health care.

The following sections explain more about the MaMIE system. Section 2 discusses other medical IE systems. Section 3 explains the system architecture of MaMIE. Section 4 shows the evaluation used by the system, while in Section 5 results and observations are presented; and Section 6 presents the overall results as well as recommendations for improvements.

2. RELATED SYSTEMS
A typical IE system has phases for input tokenization, lexical and morphological processing, some basic syntactic analysis, and identifying the information being sought in the particular application. These IE phases are shown in Figure 1. Depending on what one is interested in, some phases may not be necessary. Along with the modules in the left-hand column, as shown below, IE systems may include modules from the right-hand column, depending on the particular requirements of the application [3]. The green items represent modules, which are frequently used in IE systems. The yellow items display rarely used modules.

There are two classifications of IE systems; they can be either domain independent or domain dependent. Domain independent IE systems deal with heterogeneous texts and subject domains. This type of IE systems usually makes use of generic classification schemes, which can be refined through modifications of the information processing task and provision of more specific identification of semantic information [2]. Domain independent IE systems allows acceptance of various document types and could only provide shallow analysis of the documents. Examples of domain independent IE systems would be the AVATAR [4] and Nearly New Information Extraction (ANNIE) [5]. AVATAR IE system is a rule-based system. The source of its information is text documents. ANNIE is an open-source, IE system that depends on finite state algorithms and JAPE (Java Annotation Patterns Engine).

And the other type of IE system is domain dependent. It works as a specialized and well-depicted knowledge domain therefore using a more explicit set of classification rules. Systems like Acquiring Medical and Biological Information from Text (AMBIT) [6] and Medication Extraction System (MedEx) [7] are examples of domain dependent IE systems. These systems are specifically used in the medical domain. The AMBIT is a text analysis system which is developed to extract key information from clinical and biomedical texts. It recognizes important entities and relations between these entities referred to in documents such as case notes, laboratory reports and discharge summaries. Its specific objective is to address the requirement to extract information regarding the treatment of cancer patients. While MedEx, another medical IE system, extracts medication finding information from clinical narratives. It obtains 11 medication
3. SYSTEM ARCHITECTURE

The MaMIE System is a domain dependent IE system that extracts important information from clinical narratives of expectant mothers recorded by health workers. The system works by accepting electronic documents that are stored as text files. The electronic version of the documents serves as the input for the system. Each document will be processed through IE and fields would be filled in by the system. There is a template that contains pre-defined fields as shown in Figure 2 and the system fills the fields with the corresponding values from the documents. The output of the IE system are completed templates and stored as records in the system's database.

The MaMIE System contains cascaded modules that at each step add structure and eliminate irrelevant information through the application of certain rules. The architecture of the MaMIE System, as shown in Figure 3, includes seven modules namely Text Zoner, Sentence Splitter, POS Tagger, Named-Entity Recognizer, Preparser, Semantic Interpreter and Template Generator.

3.1 Text Zoner

Text Zoner identifies the location of the paragraphs, for the system to separate each of them. It basically functions as a paragraph finder and separator. The paragraphs of the documents will be classified as paragraph 1, paragraph 2 until the last paragraph. The main purpose of the Text Zoner is to reduce unnecessary load on the subsequent modules.

The sample input is shown in Figure 4 and the sample output of this module is shown in Figure 5.

3.2 Preprocessor

The preprocessor takes the output of the text zoner as its input. The preprocessor is composed of different sub-modules: sentence splitter, POS tagger and named-entity recognizer.

3.2.1 Sentence Splitter

Sentence Splitter identifies punctuation marks such as period, question mark and exclamation mark. It takes note of where the punctuation marks are placed wherein it is possible that that mark is the end of a sentence. It separates each sentence from one another.

3.2.2 POS Tagger

The POS Tagger simply assigns parts of speech to each word or token, such as noun, verb, adjective, etc. The output will be a list of tokens with its corresponding part-of-speech. Output for this module is shown in Figure 6.

3.2.3 Named-Entity Recognizer

The Named-Entity Recognizer processes the tokens from the POS tagger and focuses only on the proper nouns. The LingPipe's Named-Entity Recognizer is used in this research. It is a statistical named-entity recognizer. Other approaches for named-entity
recognition are dictionary-matching, maximum-entropy and rule-based. A dictionary-based named-entity recognizer demands a lot of resources to be able to generate the correct classification, these are

- Dictionary-matching
- Maximum-entropy
- Rule-based

A dictionary-based named-entity recognizer demands a lot of resources to be able to generate the correct classification, these are composed of names of all possible persons, locations and companies which not only cover English names but also Filipino ones; a maximum entropy approach requires intensive feature specific labeling to create the training data, while rule-based needs time consuming work in creating the rules and low frequency patterns are easy to be neglected. The LingPipe’s named-entity recognizer could determine basic named-entities such as names of persons, organizations and locations. Figure 6 also displays the sample output for this module.

3.3 Prepapser

The Prepapser basically recognizes phrases that are noun phrase and verb phrase. Grouping the nouns together would make the identification of entities easier for the Semantic Interpreter and the Coreference Resolution.

3.4 Semantic Interpreter

The Semantic Interpreter fills up the template with the corresponding information for each field. The proponents with the help of a maternal health expert designed the template.

3.5 Template Generator

The template generator normalizes instances of dates and places, and saves the information into the database. A date can be written in different formats. It can be written as January 2, 2007 or 01-02-07. The date is normalized into this format: month, date, year, so it becomes January 2, 2007.

4. EVALUATION

The evaluation module automatically checks the correctness of the output of the system using the common metrics: precision, recall and f-measure. Each field of the template is compared to the answer key. Direct string comparison whether they are both equal is not going to score well. Not all fields however used LCS to compute the correctness of each field, if a field has only a limited set of answers that are possible, such as whether it is an automatic review or appeal and whether the decision was affirmed, modified or reversed, the score would be only be either 0% or 100%. To check how correct the output of each field is, the longest common substring of the two is to be obtained. The formula used in calculating precision, recall and f-measure is shown in Figure 7.

A sample output of the Evaluator is shown in Figure 8. The Evaluator module compares the answer key (left) and the output of the system (right). Precision, recall and f-measure for each document is displayed. To check the accuracy of the evaluation module, manual computation of precision, recall and f-measure of two maternal health documents are done.

5. RESULTS AND OBSERVATION

There are three types of testing done: system testing, modular testing and User Acceptance Testing. For the system testing, there were 12 maternal health documents used. An expert manually
annotated these documents to serve as the answer key, which is stored in a database. An interface was provided where the expert can view and examine maternal health documents and fill out its corresponding template. The output generated by the system is compared to the answer key and evaluated by Precision, Recall and F-measure.

The total performance of the system is as follow, Precision is 87.39, Recall is 86.74 and F-measure is 87.06. These are computed using LCS of the fields.

The following are reasons why the system produced error results:

- Different writing styles/patterns of the documents by different authors
- Lack of heuristics in handling the Family History and Past Medical History fields
- Lack of lexical entries and rules for names, places and diseases
- Typographical errors in the maternal health documents

For the modular testing, testing is done to each module of the system to show the performance of each. The expected results for the POS tagger are filled by a linguist while the results for the Text Zoner, Sentence Splitter and Named-Entity Recognizer are created by the proponents. One document is used for modular testing although not the entire document is used for the testing, only some parts of it. The Text Zoner and Sentence Splitter both have a 100% accuracy rate while the POS Tagger got 82% and the Named-Entity Recognizer has 70.58% accuracy rate.

The last type of testing done is the User Acceptance Testing. This testing is done to evaluate the over-all usability of the system. This testing is targeted to be done by the intended users, which are professionals in the field of maternal health. The system is evaluated by 9 maternal health experts by answering a survey made by the proponents. The system’s performance is highly dependent on the structure of these documents, which are the basis of the rules that are created.

There are three types of testing done: system testing, modular testing and User Acceptance Testing. These testing are done in order to evaluate the performance and usability of the system. For the system testing, the MaMIE system was tested over 12 maternal health documents and has an overall performance of 87.39% Precision, 86.74% Recall and 87.06% F-measure. These evaluation metrics for the system testing were done through comparing manually annotated documents made by an expert and the output of the system. While for the modular testing, each module was tested by comparing manually annotated documents made by experts and proponents and the output of the system. The Text Zoner and Sentence Splitter have 100% accuracy rate, POS Tagger has 82% and the Named-Entity Recognizer has 70.58%. For the User Acceptance Testing, 9 maternal health experts evaluated the system by answering a survey made by the proponents. The rating is from 1 to 4 where 4 is the highest. There were 10 categories and the highest rating obtained is 3.78 for text contents readability and search function effectiveness, while the lowest rate is 3.1 for overall user interface rating and system functionalities sufficiency.
Some issues are identified during the development of the system and for further studies, these issues are recommended to be worked on.

- The POS tagger used is limited to general English words only. There is a need to tag medical terms correctly so there must be a lexicon containing medical words, which should be used for POS tagger.
- The Named-Entity Recognizer is highly dependent on the lexicon to determine if the token is a possible name of a person, place or organization. The Named-Entity Recognizer could not detect Filipino names and places such as Francisca and Bayombong since the lexicon used contains Western names and places. A Named-Entity Recognizer which learns is recommended to be used in order to avoid creating a database with Filipino names and places which is a tedious task.
- There must be a lexical management system that enables automatic adding of medical terms in a database. This provides ease in updating lexicons that would be used by the system.
- Only basic fields of the template are extracted by the system. If possible, it would be encouraged that future extensions could be done to improve the number of fields it can fill up in the template.
- The rules are all manually made by the proponents. It is possible that a machine learning module be added so that it would create the rules on its own.
- The rules of the system could only be used by a certain format of maternal health documents. The creation of rules that may cater to all types of maternal health documents could be done.
- Development of web-based front-end would also help improve the accessibility of the system. This way, medical workers would be able to retrieve information by the system even if they are in a different hospital.
- The addition of generating statistics from the extracted information may be added as a functionality in order for the users to get a view of how all the patients are doing.

7. REFERENCES