International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426

Information Extraction as a Natural Language Processing Technique

Akinbode K. B., Oguns Y. J., Fadiora B. O., Olalekan S. D.

Computer Studies Department, The Polytechnic Ibadan
ICT Department, Continuing Education Center, The Polytechnic Ibadan

Abstract: ***During the last two decades with the accelerated Internet development, a great amount of data has been being accumulated and stored on the Web. However, most of that data is stored in the form of natural language, which complicates its further analysis. Information extraction is a technology which creates the structured representation of unstructured texts by extracting relevant entities from them, thereby, making the data analysis realizable or feasible. Despite the fact that information extraction is a comparatively new area of science it evolves rather quickly and significant research has been done and are being conducted constantly. This paper closely investigates the information extraction field. The definitions for information extraction as well as its place in the text mining framework are discussed. The general structure of an information extraction system, two approaches for its creation and its evaluation framework are analyzed. Comparison of some of the systems is made. Finally, the outline of the information extraction project is given by determining its aim and objectives, research methods, tools that will be used and evaluation plan.***

Keywords: Unstructured text, text mining, Information extraction, Natural language processing techniques

1. Introduction

With a huge amount of data available on the Web it is important to have some technologies and tools to analyze it, derive information and gain knowledge from it which can be used later for any other purposes. Text mining is one of those technologies which allow obtaining useful information from data presented in any unstructured textual form. Information extraction is one of the initial links of the text mining chain. Its major goal is to transform the data from unstructured form into structured representation.

The information extraction task can be formulated as to process the collection of texts which belong to a particular field and derive from each of them a previously defined set of name types, relations between them and events in which they participate. Each set of extracted entities is added, for instance, as a record to a table of a relational database in order that data mining techniques can be applied to this structured dataset later. There are two approaches to the information extraction system design, namely knowledge engineering and automatic training approaches. Both of them have their own benefits and drawbacks and are applied depending on the resources available to the system's designer.

There are several issues that distinguish information extraction from other fields of study. Firstly, there is still no correct answer and probably there will not be any for the question about which components of the information extraction pipeline must be integrated into the system and which of them are not so important. There is always room for discussions and different approaches. Another thing is that the progress in this area and the state of the art are evaluated through the periodic conferences which are held in a form of competition with a specific predefined task and results review.

The main aim of this project is to understand the principles of information extraction by developing an information extraction system which must execute its major task applying to a particular domain. Natural Language Processing (NLP) is used to describe the function of software or hardware component in a computer system which analyze or synthesize written or spoken language. (Jackson P. et al. 2007).

2. Problem Statement

Terrorist attacks are the major problem for the society and there is a high increase on the rate of terrorism in Nigeria. The main problems that this research work addresses are:

i. To know the intentions of the terrorist early and avoid the attacks.
ii. To monitor and detect suspicious messages in chats.
iii. To extract the exact meaning of the conversations in a terrorist chat.

3. Background to the Problem

The concept of information extraction is presented. The history of its development can be traced through the discussion. The main aspects of the information extraction field like major approaches, evaluation techniques and design issues are investigated.

Text Mining and Information Extraction

According to Moens (2006) there have been several attempts to estimate how much information the Web contains. Even though it is obvious that such kinds of measurements are very rough and approximate, they allow us to gain general understanding of the volume of available data and predict that if the trend remains the same we will have to estimate the information in millions of bytes in the near future.

However, the amount of accessible information would not be of much use if there were no suitable techniques to process it and extract knowledge from it. Thus, text mining

is one of the technologies which are employed for those purposes. It can be described as a process of identifying the unknown information from a variety of unstructured data sources with a goal of further analysis of the derived facts.

It is possible to draw a parallel between data mining and text mining technologies. Both of them obtain useful information from the available data sources by searching for and discovering patterns. However, data mining operates on structured data in the form of database records, whereas text mining investigates unstructured or semi-structured content of textual documents. This difference affects the way a text mining system is designed forcing it to have special subsystems to deal with unstructured information (Ben-Dov and Feldman, 2005; Feldman and Sanger, 2007).

According to Feldman and Sanger (2007) the architecture of any text mining system contains the following four main components:

i. Pre-processing which includes activities to prepare data to the next step. Typically, they involve the process of converting the raw data from original source into the format which is suitable for applying core mining operations.
ii. Core mining operations which are the essence of the text mining technology. They provide algorithms for pattern discovery in the data extracted from documents by the first component. The most widespread of them are distributions, frequent and near frequent sets and associations.
iii. Presentation which provides a user interface with a query editor and visualization tools.
iv. Refinement which includes optimization operations with the resulting data.

The pre-processing operations are divided into two broad categories which are techniques according to their task and according to the algorithms and frameworks they employ. The FirstGroup of approaches provides the structuring of the source documents and presenting them as the task requires. The second group contains the approaches which imply the application of formal methods for analyzing available data. However, different techniques from both categories can be used in conjunction to solve many text mining tasks.

Information extraction is considered as a part of the task-oriented pre-processing approaches alongside with preparatory processing and other natural language processing (NLP) techniques. While the other NLP and preparatory tasks can be defined as domain-independent, information extraction itself is a highly domain-dependent technology (Ben-Dov and Feldman, 2005; Feldman and Sanger, 2007). Therefore, in the context of text mining technology information extraction can be classified as one of the pre-processing tasks which are used in order to make data ready for applying major data mining techniques. These pre-processing operations involve processing the input, unstructured information in the form of documents, and presenting it in a more structured way to make further post-processing analysis possible.

Defining Information Extraction

Despite the fact that information extraction is generally considered as a link in the chain of text mining techniques, it is a powerful technology itself. Even within the text mining operations, Ben-Dov and Feldman (2005) mention information extraction as the most important pre-processing technique which significantly increases the text mining potential. But moreover it is used as a self-dependent technology to settle the particular issues concerning the processing of the text information.

There are a lot of problem when the information to be analyzed is available primarily only in the form of natural text, such as technical reports, scientific articles, log records, news and chats. For instance, a hospital wants to produce its own statistics about the most commonly encountered diseases within the age and gender groups of patients. But the data they need is mostly stored in medical records in textual form. Another example can be provided from a business area. A particular company or business agency wishes to know the tendency of enterprises' bankruptcies by industries. That kind of information can be taken only from news reports. In both cases the information extraction is able to help and accomplish those kinds of tasks avoiding people to process large amounts of text documents by hand. It reduces the amount of information to be analyzed by extracting useful facts and ignoring irrelevant ones. Derived data is presented then in a more structured database way when it is easily accessible for applying different analyzing techniques (Grishman, 1997; Grishman, 2003).

To explain the term information extraction, definitions from different authors are cited further. Moens (2006) discusses different definitions of the term from such authors like Riloff and Lorenzen, Cowie and Lehnert. She points out the limitations of those examples and according to them suggests the factors which must be taken into consideration while defining information extraction. Some of those factors are listed below:

i. An information extraction system's independency from a specific domain. In general, information extraction is highly domain dependent. A particular system is built to solve a particular kind of extraction problem. However, the overall aim in the development of information extraction as a field of study is to design systems which can easily switch from one area to another and can be applied to different extraction tasks without much effort. That is why according to Moens (2006) this issue must be considered in the long-term definition of information extraction.
ii. Information extraction deals with identifying not only named entities but relationships between those entities and events as well. This fact must be explicitly stated in the definition.
iii. Not only natural language text is considered as unstructured information. Video, chat conversations and image can be classified in that way as well.

> Fulfilling the conditions above Moens (2006) introduces her definition of information extraction which is used within the context of her work. "Information extraction is the identification, and consequent or concurrent classification and structuring into semantic classes, of specific information found in unstructured data sources, such as natural language text, making the information more suitable for information processing tasks" (Moens, 2006).

Grishman (1997) explains the meaning of information extraction quite similar to the way Moens (2006) does. His definition indicates relationship and event identification and clearly specifies what kind of result of applying this technique will be. According to Grishman information extraction is "[...] the identification of instances of a particular class of events or relationships in a natural language text, and the extraction of the relevant arguments of the event or relationship. Information extraction therefore involves the creation of a structured representation (such as a data base) of selected information drawn from the text".

Turmo et al. (2006) present their own vision on formulating the definition of information extraction by providing its major goal. According to them "The objective of information extraction is to extract certain pieces of information from text that are related to a prescribed set of related concepts, namely, an extraction scenario" (Turmo et al., 2006).

As soon as there is no classical definition for information extraction every author defines it in the way which he or she believes explains information extraction in the better way. That is why for this research work will try to define information extraction technique as well, taking into consideration the conditions and limitations involved in the project.

Firstly, we agree with Moens' (2006) remark that an ideal information extraction system should not depend on the specific domain of knowledge to be extracted. However, in the case of this project a terrorist domain has been determined from the very beginning and there is no need to interpret information extraction in a larger context. Another Moens' statement that makes her definition too wide for the current work is about considering image and video as unstructured information as well alongside text. Despite the fact that the observation itself is true, image and video will not be regarded as the source of unstructured information in the project which is currently being implemented. That is why talking about other data sources apart from text and chat documents in the definition here would be unreasonable.

Moens' (2006) comment about mentioning in the definition the extraction not only of entities but of relationships between them and events seems very credible and will be taken into account in the definition below. Finally, in our opinion, the aim of name and event extraction from texts must be explicitly stated in the definition since it might not be clear for an untrained user from the very beginning.

Here is the definition of information extraction we have come up with taking into account everything mentioned above. It is defined in a more simplified way but without losing its core idea and aims. Information extraction is the identification and selection of the named entities relevant to the specific task, of the relationships between them and events in which they participate in the natural language text in order to make them more accessible for further manipulations.

Apart from the definition of information extraction, the difference between information extraction and information retrieval must be explained, since these two techniques are often mutually confused. Information retrieval can be characterized as the operation previous to information extraction within the text mining framework. The aim of information retrieval is to filter the available documents and find those which correspond to the queries representing the user's information need. After this process information extraction derives names and events from the texts provided by the information retrieval mechanism.

Another way to distinguish between these two techniques is to look at their output results. In the case of information retrieval, the output is the collection of documents relevant to the user's information need, although he must then read these in order to obtain precise information; whereas after information extraction, a user has a collection of records with different entities, relations and events which have been derived from those documents (Cowie and Lehnert, 1996; Wilks, 1997; Appelt and Israel, 1999; Ben-Dov and Feldman, 2005). Usually an information extraction system supports one of the two basic approaches of extraction, namely, Knowledge Engineering Approach and Automatic Training Approach.

Knowledge Engineering Approach

In order to extract information from available texts using a system which supports a knowledge engineering approach a set of extraction rules must be written manually. A person who creates such a type of system, or is responsible for writing those rules (i.e., a knowledge engineer) must be an expert in the knowledge domain chosen for extraction or at least must be closely familiar with it. Apart from that, a designer must know the formalism for writing those rules for the particular system used. Usually the knowledge engineer has a number of texts which are related to the chosen domain. Analyzing those texts, the designer finds common patterns in them and writes the rules using his or her intuition, which according to Appelt and Israel (1999) is a very important factor in creating a system with a high level of performance.

The rules are then interpreted by the components of the information extraction system and useful facts are found and extracted from the texts. It is worth mentioning that creating an information extraction system using this approach is a highly time and effort consuming iterative process. Firstly, the knowledge engineer writes a particular rule. Then he applies it to the available texts and checks whether it works correctly or not. Modifications are done

if needed and the rule is examined again until a desirable result is achieved. Since this approach involves writing rules, in some sources it is called as a rule-based approach.

Automatic Training Approach

In this case there is no need to design extraction rules manually. Therefore, a person who is responsible for the information extraction process does not have to know how to write rules and how a system works. A machine learning algorithm implemented in the information extraction system creates those rules. In order to do that the algorithm must have access to a large number of training texts related to the chosen domain. Those texts must be annotated manually in advance to provide examples on which the algorithm can learn and produce extraction rules. Thereby, the engineer must provide the set of training documents and be able to annotate them. Among algorithms that can be used for the automatic training approach there are decision trees, maximum entropy models and hidden Markov models (Appelt and Israel, 1999). In many sources this approach is named as the machine learning approach. The development of this method allows the information extraction area to become less domain-independent since the same machine learning algorithm can be applied to different domains as long as corpora of domain-related texts are available.

According to Moens (2006) a machine learning process can be supervised or unsupervised. Supervised learning is described above when a number of documents is used to help the algorithm to learn about the information to be extracted. Unsupervised learning means an annotated corpus is not used to improve the system's level of performance. As a type of unsupervised learning a weakly supervised approach exists when the algorithm uses a limited number of annotated texts and a large number of unlabeled documents. However, it is not necessary to create all the components of an information extraction system using only one particular approach. It is quite possible to interchange these two approaches while building different components of the system. One of the reasons of having such a possibility is that one can never say objectively which approach is better. Both of them have their advantages and disadvantages.

As Appelt and Israel (1999) stated, the systems which use a knowledge engineering approach show a higher performance compared to the other ones. However, they require a lot of effort and time and depend on the knowledge engineer's skills and experience and availability of linguistic resources. The very important advantage of a machine learning based system is that it can be transferred to a different domain easily as long as specific texts and a person who can annotate them are available. But sometimes those texts are problematic or expensive to obtain or there is a lack of useful documents on which an algorithm can learn, and manual (or even machine-aided) annotation on the scale needed to provide reasonable levels of performance may be expensive. On the basis of analyzing the benefits and drawbacks of both approaches it is possible to conclude with the criteria which determine the choice of one of them. The most important condition to choose the automatic training approach is the presence of a set of suitable texts which can be used to train the algorithm. In the case of the knowledge engineering approach the availability of a person who is experienced in writing extraction rules is the most crucial criterion. Other aspects which can be considered are the specifications and the level of performance. If the specifications are subject to change and the level of performance is desired to be as higher as possible it is more reasonable to apply the rule-based approach, otherwise machine learning mechanisms can be employed. However, the current project will make use of the automatic training approach.

The Overall Process of Information Extraction

Different authors divide the process of information extraction in different steps of different granularity, combining them into bigger stages and assigning the components of the information extraction systems to accomplish the tasks involved (Hobbs, 1993; Cowie and Lehnert, 1996; Grishman, 1997; Appelt and Israel, 1999; Turmo et al., 2006; Feldman and Sanger, 2007). However, analyzing those different approaches the general pipeline of the information extraction process can be summarized. In the current work six main stages were determined as following: Initial processing, Proper names identification, Parsing, Extraction of events and relations, Anaphora resolution, Output results generation.

Initial Processing

There are several operations which usually compose the primary step of the information extraction process. The first of them is the splitting a text into the fragments which are defined differently throughout the papers from different researchers like zones, sentences, segments or tokens. This procedure can be performed by the components named as tokenizers, text zoners, segmenters or splitters. As Appelt and Israel (1999) stated, tokenization is a quite straightforward task for the texts in any European language, where the blank space between characters and punctuation indicate the boundaries of a word and a sentence respectively. But, for example, for Chinese or Japanese texts, where the boundaries are not so obvious this operation is not the simple one and requires much more effort to fulfill it.

The next task within the initial processing stage is usually the morphological analysis which includes part-of-speech tagging and phrasal units (noun or verb phrases) identification. Part of-speech tagging might be helpful to the next step which is the lexical analysis. It handles unknown words and resolves ambiguities, some of them by identifying part-of-speech of the words which cause those ambiguities. In addition, the lexical analysis involves working with the specialized dictionaries and gazetteers, which are composed of different types of names: titles, countries, cities, companies and their suffixes, positions in a company, etc. If a word in a document is found in a gazetteer it is tagged with the semantic class the word belongs to. For example, a word "Mr" will be tagged with the semantic class "Titles". Some authors add a filtering

task to the pre-processing stage which implies selecting only those sentences which are relevant to the extraction requirements (Hobbs, 1993; Turmo et al., 2006).

Proper Names Identification

One of the most important operations in the chain of information extraction is the identification of various classes of proper names, such as names of people or organizations, dates, currency amounts, locations, addresses, etc. They can be encountered in almost all types of texts and usually they constitute the part of the extraction scenario. These names are recognized using a number of patterns which are called regular expressions (Feldman and Sanger, 2007). However, usually authors do not classify this operation as a separate task within the whole information extraction process.

Parsing

During this stage the syntactic analysis of the sentences in the documents is performed. After the previous step, where the basic entities were recognized the sentences are parsed to identify the noun group around some of those entities and verb groups. This parsing stage must be done in order to prepare the ground for the next stage of extraction relations between those entities and events in which they participate. The noun and verb groups are used as sections to begin to work on at the pattern matching stage. The identification of those groups is realized by applying a set of specially constructed regular expressions (Grishman, 1997; Feldman and Sanger, 2007).

However, the full parsing is not an easy task; therefore, it requires expensive computations to be involved which in its turn slow down the whole process of information extraction. Since it is a difficult problem, the full parsing is prone to introduce errors. In contrast, sometimes the full syntactic analysis might not be needed at all. Thereby, more and more information extraction research groups tend to use so called partial or shallow parsing instead of full one. Using only local information the shallow parsing creates partial, not overlapping syntactic fragments which are identified with a higher level of confidence. At the beginning of the evaluation process all of the MUC's participants used the full parsing. And the group that came up with the new idea of shallow parsing was Lehnert et. al. during MUC-3 in 1991. As a result of applying the partial syntactic analysis, they showed a better performance than the rest of the sites which tried to create full syntactic structures (Grishman, 1997; Appelt and Israel, 1999; Turmo et al., 2006).

Extraction of Events and Relations

Everything which is done previously is basically the preparation for the major stage of extraction of events and relations, which are particularly related to the initial extraction specifications given by a client. This process is realized by creating and applying extraction rules which specify different patterns. The text is matched against those patterns and if a match is found the element of the text is labeled and later extracted. The formalism of writing those extraction rules differs from one information extraction system to another (Grishman, 1997; Appelt and Israel, 1999; Feldman and Sanger, 2007).

Anaphora Resolution

A given entity in a text can be referred to several times and every time it might be referred differently. In order to identify all the ways used to name that entity throughout the document co-reference resolution is performed. Co-reference or anaphora resolution is the stage when fornoun phrases it is determined if they refer to the same entity or not. There are several types of co-reference, but the most common types are pronominal and proper names co-reference, when a noun is replaced by a pronoun in the first case and by another noun or a noun phrase in the second one (Appelt and Israel, 1999; Feldman and Sanger, 2007).

Output Results Generation

This stage involves transforming the structures which were extracted during the previous operations into the output templates according to the format specified by a client. It might include different normalization operations for dates, time, currencies, etc. For instance, a round-off procedure for percentages can be executed and areal number 75.96 will be turned into integer 76 (Hobbs, 1993; Turmo et al., 2006). Not all of the tasks must be necessarily accomplished within one information extraction project. Therefore, a particular information extraction system does not have to have all of those possible components. According to Appelt and Israel (1999) there are several factors that affect the choice of systems' components, like:

i. Language. As it was mentioned earlier for processing texts in Chinese or Japanese languages with not clear word and sentence boundaries or texts in German language with words of a difficult morphological structure some modules are definitely necessary compared to working with English documents.
ii. Text genre and properties. In transcripts of informal speech, for example, spelling mistakes might occur in addition to implicit sentence boundaries. If information must be extracted from such texts those issues must be taken into consideration and addressed while designing a system by adding corresponding modules.
iii. Extraction task. For an easy task like names recognition the parsing and anaphora resolution modules might not be needed at all.

Software Architectures for Information Extraction Systems Design

At the earliest stages of the development of information extraction as a field of study research groups designed information extraction systems from scratch every time they faced a different extraction problem. That was partly because at that time the major task was to solve the extraction problem and reusability of the tools created was not considered at all. Later, when the need for the integration of the tools developed by different groups was

realized it was almost impossible to accomplish that task because of the diverse programming platforms used and the fact that the tools were not meant to be used in another application (Kano et al., 2008).

Since then several architectures have been developed to facilitate the process of the information systems development by providing the common platform for systems' components design, integration and reuse. Among them are the Unstructured Information Management Architecture (UIMA), the General Architecture for Text Engineering (GATE), the Architecture and Tools for Linguistic Analysis Systems (ATLAS), the Automated Linguistic Processing Environment (ALPE) (Dietl et al., 2008). Employing either of them it is possible to:

i. Reuse the tools for natural language processing and text mining which have been previously created by other developers.
ii. Quickly combine different tools and thereby analyze possible approaches to design of the language processing software.

The first two architectures (UIMA and GATE) are the most prominent and provide almost the same capabilities. UIMA was created by IBM and then became an Apache open-source project. Both Java and C++ frameworks are available. One of the major distinguishing features of UIMA is a Common Analysis Structure (CAS) which represents an original document and its stand-off annotations.

Thus, the UIMA processing engine works as following. A CAS Initialize acquires raw documents through the Collection Reader interface and produces the initial CASs. Then Text Analysis Engines (such as language translators, grammatical parsers or document classifiers) perform the document-level analysis, modify the CASs and transfer them to the CAS Consumers. The latter in their turn execute the collection-level analysis. It can be said that the main interface within the UIMA processing engine takes CASs as input and returns them as output (Ferrucci and Lally, 2004).

GATE is an open-source architecture written in Java which was created by the University of Sheffield. One of the main elements of GATE is the GATE Document Manager (GDM). The GDM model includes three elements: a collection with documents which contain texts and annotations upon them. Thus, the GDM stores all the information about the texts which is produced by the system. All the components of the system interact with each other only through GDM which decreases the number of communication interfaces to one. CREOLE, collection of Reusable Objects for Language Engineering, is the GATE element which performs all the tasks of text analysis (Cunningham, 2002).

In the case of UIMA the unstructured data sources can be not only just plain text or HTML page, an audio or video streams can be processed as well. GATE in its turn supports XML, HTML, RTF, SML formats and plain texts (Dietl et al., 2008).

Both GATE and UIMA have the graphical user interface for tools searching, browsing and integration. In order to upload an existing text analysis tool to the collection of predefined components existing within the both architectures a wrapping procedure must be performed. To be integrated into UIMA a tool must be written in C++, Java, Perl Python or TCL. The C/C++, Java, TCL, Prolog, Lisp and Perl tool's implementations are right for GATE (Cunningham, 2002; Kano et al., 2008).

Thus, with the advent of such common frameworks as UIMA and GATE a huge step forward has been made in the development of the text mining technologies in general and in the information extraction area in particular. The latter has become more efficient since the researchers can draw on the other researchers' successful experience and have a platform for quick systems design.

The Aim of the Project, Objectives, Limitation and Deliverables

The main aim of this research work is to analyze communication chats by designing and implementing a machine learning algorithm that can access terrorist chats inform of texts and annotate it in order to develop extraction rules that can be used for natural language processing and information extraction.

To achieve the aim of the project a list of objectives was set which takes into consideration the limitations mentioned above:

i. Study the state of the art in the information extraction field, the approaches for system design and evaluation methods.
ii. Choose the domain of texts the information to be extracted and define the template(s) with a number of slots to be filled in.
iii. Formalization of the system which will be used to develop extraction rules.
iv. Explore the gazetteers provided by the system and create the new ones if needed.
v. Test the extraction rules.
vi. Evaluate the level of performance calculating Precision, Recall and F measure.

The following are the limitations involved in the project:

i. In the case of the project performed we act as developers as well as users. This means we establish the requirements for information to be extracted and then create rules to meet those requirements.
ii. The data source for the information to be extracted from is the free unstructured texts with plain, grammatical sentences in English language.

The main project deliverables will be: a gazetteer or a set of them, a set of extraction rules, and a project report.

4. Methodology

There are several approaches to the information systems development. The oldest one is the waterfall model which was introduced in 1960s. Before that period there were no predefined formal procedures that must be followed during the software design. The waterfall model brought an order to the development process and formalized it.

According to this model the development process must go through several consequent stages including identifying requirements, design, implementation, testing, operation and maintenance. The output of a previous stage becomes the input for a next stage. The main idea of the waterfall model is that the system's specifications are defined in the beginning of the process and the rest of the phases are accomplished based on those. However, this approach has been criticized because of the issues that it does not take into account. First of them is that for end-users it is very difficult to define and formulate their real requirements for the system at the beginning of the starting point. Another problem is that the requirements might change after a significant amount of work has been done. Finally, there is lack of communication with end-users during the development process and some design errors, for instance, are discovered later, at the testing stage (MacCormack et al., 2003; Sommerville, 1996).

Another alternative to decide on the software development method is the prototyping model. It includes the following stages:

i. Produce only the outline of the system's specifications which can be modified later but still serves as a guide for developers.
ii. Develop the first prototype of the software according to those initial requirements.
iii. Test the system with the end-users involved.

The crucial aspect of this approach is that it implies a feedback to the previous stages. It happens if the system does not meet the users' needs. Some changes are made in the requirements and a second prototype is developed. The process is repeated until the users are satisfied with the product (MacCormack et al., 2003; Sommerville, 1996).The current project is a combination of the two models mentioned will be employed. The Figure 1 depicts the adapted development process.

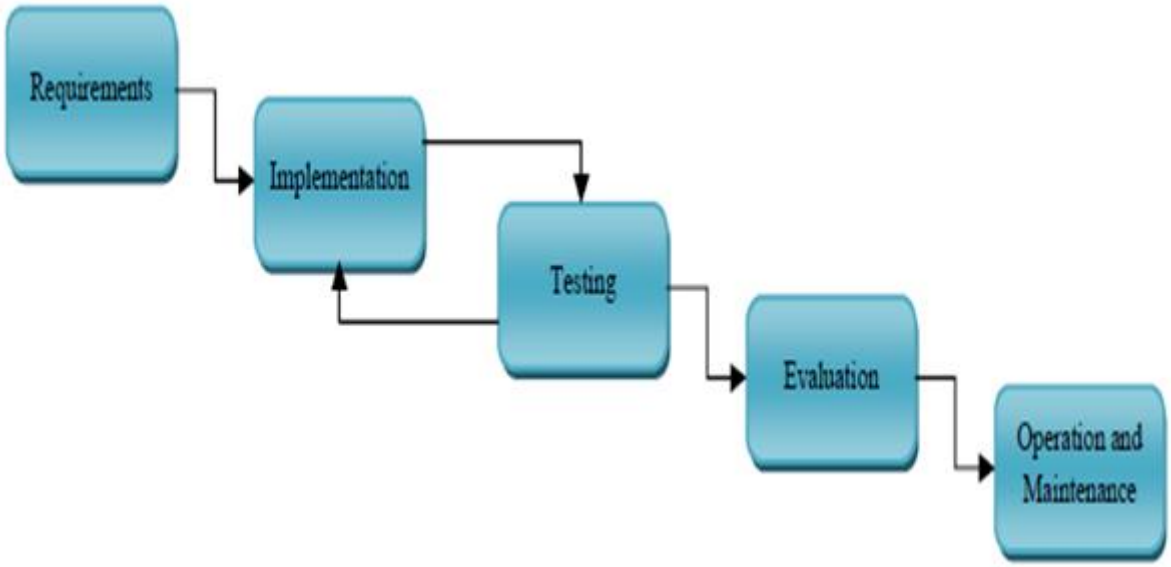

Figure 1: Adapted model for information extraction system development

The reason for using the mixture of the two models is hidden in the nature of any information extraction project. In general, the whole project will be carried out based on the waterfall model. However, some elements of the prototyping will be included. This is done because the extraction rules are created one by one and the testing procedure must be performed straight after the rule is written in order to check if it works or not. That is why there will be a cycle between the implementation and testing stages. At the same time, it is not a pure prototyping model since the actual requirements; in this case – the entities to be extracted – remain the same.

System Requirements

Information extraction can be applied to a wide range of text and chat domains. As we can see domains vary from Joint Ventures from business news to communication chats to Airline Crashes Reports. If a particular domain must be processed within the information extraction framework it must meet some requirements. The major of them is that the names, relations and events that need to be extracted must be present in all of the texts. Ideally the texts should correspond to the common structure, but it is not the necessary condition.

For the current project Terrorist communication chats has been chosen. The text will be taken from the yahoo messenger. The information extracted can be used then, for instance, to analyze areas with the most frequent terrorist attack, the periodicity of the terrorist attack in a particular region, or the magnitude. The names entities that will be extracted are place, date, time, magnitude, number of people affected, damage caused.

Implementation and Testing

GATE is a system which will be used within the current project. GATE is an abbreviation for General Architecture for text engineering. It is a standalone system which follows the machine learning approach. Each text goes through the following stages of processing within the information extraction chain, which are depicted on the Figure 3.Chats, Plain text; HTML documents can be accessed as input. The very first step of the process performs separation of the cover of the document from its body and dividing the text into paragraphs. After that, at the tokenization stage the text within each paragraph is split up into different segments like words, numbers, punctuation, etc. which are referred to as tokens. Tagging stage is responsible for specifying a part of speech for each Tokenization Tagging Gazetteer lookup token. Gazetteer lookup means words are labeled if they are found in the special dictionaries– gazetteers. And the final stage within this chain is rule application for named entities, relations and events recognition and co reference resolution (Black et al., 2005).

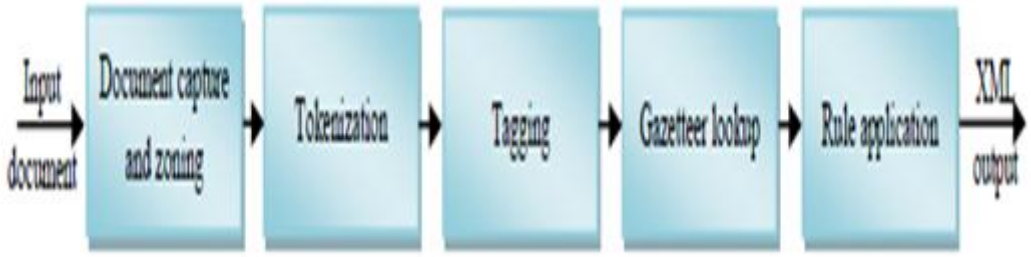

Figure 3: The GATE stages of information extraction process

The first four stages are already implemented in the system. And within this project we must create a set of extraction rules in order to process a predefined collection of texts and fulfill the rule application stage. In addition, a gazetteer can be expanded if needed. Thus, the implementation step of the project development process implies the execution of these tasks. As it was discussed earlier the implementation and testing steps will be carried out according to the prototyping model. This means as soon as a single extraction rule is written, it is tested straight away on a number of texts, and if the result is unsatisfying the rule is rewritten and tested again. This cycle continues until the rule provides the results needed.

5. Evaluation

The evaluation stage is an important part of any project undertaken, since this process rates the quality of the work that has been done. In the case of an information extraction project the level of performance is determined by calculating Precision, Recall and F measure. Within the current project the MUC evaluation scheme will be adopted as follows. A number of texts from the chosen domain will be picked out and extraction rules will be developed using that text corpus available. Then after the stage of grammar design and testing will be finished another group of texts will be selected. The entities from the target template will be extracted from those texts twice, namely manually and using the system developed.

Precision measures the reliability of the information extracted that shown below. Recall measures the amount of the relevant information that the natural processing language system correctly extracts from the test dataset.

$$precision = \frac{\#\ correct\ slot\ fillers\ in\ output\ templates}{slot\ fillers\ in\ output\ templates}$$

$$recall = \frac{\#\ correct\ slot\ fillers\ in\ output\ templates}{slot\ fillers\ in\ answer\ keys}$$

6. Conclusion

The current work is the initial background report for the information extraction project. The aim of the report is to provide a literature review of the information extraction field and give a framework according to which the project itself will be carried out. Within this paper the area of information extraction has been carefully studied. The definition of the term information extraction which reflects the features of the current project has been given and its place in the sequence of text mining techniques has been determined. The two approaches for the information extraction system design have been examined and the factors which influence the choice of one of them have been listed. The discussion about the stages of the extraction process has been presented and an attempt to classify the characteristics of the IE systems and compare the systems according to them has been made.

Finally, the two architectures, namely UIMA and GATE, which provide a common platform for the information extraction systems design, have been examined. Regarding the current project itself the methodology that will be used for the development of the information extraction system is described. It is the combination of waterfall and prototyping models which shows the character of the project.

References

[1] Appelt, D. and Israel, D. (1999) *Introduction to Information Extraction Technology*: IJCAI-99tutorial <http://www.ai.sri.com/~appelt/ie-tutorial/IJCAI99.pdf> (Accessed on 06/05/10).

[2] Ben-Dov, M. and Feldman, R. (2005)"Text Mining and Information Extraction". In: Maimon, O. and Rokach, L. (eds.) *Data Mining and Knowledge Discovery Handbook*. Springer Science + Business Media, Inc., pp. 801-831.

[3] Black, W.J., McNaught, J., Vasilakopoulos, A., Zervanou, K., Theodoulidis, B., and Rinaldi, F. (2005) "CAFETIERE: Conceptual Annotations for Facts, Events, Terms, Individual Entities, and RElations", Parmenides Technical Report TR-U4.3.1 <http://www.nactem.ac.uk/files/phatfile/cafetiere-report.pdf> (Accessed on 06/05/10).

[4] Buckland, M. and Gey, F. (1994) "The Relationship between Recall and Precision", *Journal of the American Society for Information Science*, 45(1), pp. 12-19.

[5] Chang, C.-H., Kayed, M., Girgis, M.R., and Shaalan, K.F (2006) "A Survey of Web Information Extraction Systems", *Proceedings of the IEEE Transactions on Knowledge and Data Engineering*, 18(10), pp. 1411-1428.

[6] Cowie, J. and Lehnert, W. (1996) "Information Extraction", *Communication of the ACM*, 39(1), pp. 80-91.

[7] Cunningham, H. (2002) "GATE, A General Architecture for Text Engineering", *Computers and Humanities*, 36(2), pp. 223-254.

[8] Dietl, R., Hoisl, B., Wild, F., Richter, B., Essl, M. and Doppler, G. (2008) Project Deliverable Report. Deliverable D2.1 – Services Approach & Overview General Tools and Resources <http://www.ltfll-project.org/tl_files/Dokus_Flash/LTFLL_D2.1.pdf> (Accessed on 06/05/10)

[9] Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. (2004) "The Automatic Content Extraction (ACE) Programme – Tasks, Data and Evaluation", *Proceedings of the Conference on Language Resources and Evaluation.*

[10] Feldman, R. and Sanger, J. (2007) *The Text Mining Handbook: Advanced Approaches In Analyzing Unstructured Data*. New York: Cambridge University Press.

[11] Ferrucci, D. and Lally, A. (2004) "UIMA: an Architectural Approach to Unstructured Information Processing in the Corporate Research Environment", *Natural Language Engineering*, 10(3/4), pp. 327-348.

[12] Grishman, R. and Sundheim, B. (1996) "Message Understanding Conference – 6: A Brief History", *Proceedings of the 16th conference on Computational Linguistics*, 1, pp. 466-471.

[13] Grishman, R. (1997) "Information Extraction: Techniques and Challenges". In: Pazienza, M.T. (ed.) *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*. Berlin, Heidelberg: Springer-Verlag, pp. 10-27.

[14] Grishman, R. (2003) "Information Extraction". In: Mitkov, R. (ed.) *The Oxford Handbook of Computational Linguistics*. Oxford: Oxford University Press, pp. 545-559.Hobbs, J.R. (1993) "The Generic Information Extraction System", *Proceedings on the 5th Conference on Message Understanding*, pp. 87-91.

[15] Kaiser, K. and Miksch, S. (2005) "Information Extraction: A Survey"<http://ieg.ifs.tuwien.ac.at/techreports/Asgaard-TR-2005-6.pdf> (Accessed on 06/05/10).

[16] Kano, Y., Nguyen, N., Sætre, R., Yoshida, K., Miyao, Y., Tsuruoka, Y., Matsubayashi, Y., Ananiadou, S., and Tsujii, J. (2008) "Filling the Gaps between Tools and Users: A Tool Comparator, Using Protein-Protein Interaction as an Example", *PSB 2008 Online Proceedings* <http://psb.stanford.edu/psb-online/proceedings/psb08/kano.pdf> (Accessed on06/05/10).

[17] Kuhlins, S. and Tredwell, R. (2002) "Toolkits for Generating Wrappers: A Survey of Software Toolkits for Automated Data Extraction from Web Sites". In: Aksit, M., Mezini, M., and Unland, R. (eds.) *Objects, Components, Architecture, Services, and Applications for a Network World*. Berlin, Heidelberg: Springer-Verlag, pp. 184-198.

[18] Laender, A.H.F., Ribeiro-Neto, B.A., Da Silva, A.S., and Teixeira, J.S. (2002) "A Brief survey of Web Data Extraction Tools", *ACM SIGMOD Records*, 31(2), pp. 84-93.

[19] MacCormack, A., Kemerer C.F., Cusumano, M., and Crandall, B. (2003) "Trade-offs between Productivity and Quality in Selecting Software Development Practices", *IEEE Software*, 20(5), pp. 78-85.

[20] Moens, M.-F. (2006) *Information Extraction: Algorithms and Prospects in a Retrieval Context*. Springer Netherlands.

[21] Muslea, I. (1999) "Extraction Patterns for Information Extraction Tasks: A Survey", *Proceedings of the AAAI Workshop on Machine Learning for Information Extraction.*

[22] Rodriguez-Esteban, R. (2009) "Biomedical Text Mining and Its Applications", *PLoS Computational Biology*, 5(12).

[23] Seifkes, C. and Siniakov, P. (2005) "An Overview and Classification of Adaptive Approaches to Information Extraction". In: Spaccapietra, S. (ed.) *Journal on Data Semantics IV*, Berlin, Heidelberg: Springer-Verlag, pp. 172-212.

[24] Sommerville, I. (1996) "Software Process Models", *ACM Computing Surveys*, 28(1), pp. 269-271.

[25] Turmo, J., Ageno, A., and Catala, N. (2006) "Adaptive Information Extraction", *ACM Computing Surveys*, 38(2), pp. 1-47.

[26] Wilks, Y. (1997) "Information Extraction as a Core Language Technology". In: Pazienza, M.T. (ed.) *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*. Berlin, Heidelberg: Springer-Verlag, pp. 1-9.NIST 2008 Automatic Content Extraction Evaluation (ACE08). Official Results. Date of Release: September 29, 2008<http://www.itl.nist.gov/iaui/894.01/tests/ace/2008/doc/ace08_eval_official_results_20080929.html> (Accessed on 06/05/10)