UNIVERSITY OF MANCHESTER

SCHOOL OF COMPUTER SCIENCE

COMP 60990: Research Methods and Professional Skills

INFORMATION EXTRACTION

Initial background report

Student Name: Madina Ipalakova
Programme: ACS and ITM
Supervisor: Dr. John McNaught

2010
# TABLE OF CONTENTS

ABSTRACT .......................................................................................................................... 3  
INTRODUCTION ................................................................................................................. 4  
CHAPTER 1. BACKGROUND ............................................................................................. 5  
  1.1 Text Mining and Information Extraction ................................................................... 5  
  1.2 Defining Information Extraction ................................................................................ 6  
  1.3 Information Extraction Systems Evaluation ............................................................... 8  
  1.4 Two Approaches ..................................................................................................... 11  
  1.5 The Overall Process of Information Extraction ......................................................... 13  
  1.6 Comparison of the Information Extraction Systems .................................................... 15  
  1.7 Software Architectures for Information Extraction Systems Design ....................... 17  
  1.8 Summary ................................................................................................................. 18  
CHAPTER 2. RESEARCH METHODS AND PROJECT PLAN ......................................... 19  
  2.1 The Aim of the Project, Its Objectives and Deliverables ........................................... 19  
  2.2 Methodology ........................................................................................................... 20  
  2.3 Project Plan ............................................................................................................. 22  
CONCLUSION ................................................................................................................... 23  
LIST OF REFERENCES ..................................................................................................... 24
ABSTRACT

During the last two decades with the accelerated Internet development a great amount of data have 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 realisable.

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 analysed. 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.
INTRODUCTION

With a huge amount of data available on the Web it is important to have some technologies and tools to analyse 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.

This is the initial report for the project and it is organised as follows. Chapter 1, Background, provides an overview of the information extraction area. An attempt to define information extraction for the purposes of the current project taking into consideration its features and limitations is made. The questions touched upon above are closely investigated. Chapter 2 concentrates on the research methods employed. The list of objectives which will help to achieve the main aim of the project is formulated. The methodology that will be used to accomplish the current project is described. Finally, the project plan in the form of Gantt chart is presented.
CHAPTER 1. BACKGROUND

In this chapter the examination 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.

1.1 Text Mining and Information Extraction

Nobody will contradict the statement that we live in the Information Age. Information per se, information technologies in general, the Web and the Internet in particular have totally changed the way we work, study and communicate. There is an enormous amount of information on the Web which is available now for almost anyone. 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 yottabytes (1 yottabyte is equal to $2^{80}$ 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:

1. 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.

2. 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.

3. Presentation which provides a user interface with a query editor and visualisation tools.


4. Refinement which includes optimisation 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 first group 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 analysing 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).

Thereby, 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.

1.2 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 situations when information must be analysed somehow but it is available primarily only in the form of natural text, such as technical reports, scientific articles, log records, news, etc. 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 analysed 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 analysing 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:

- 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.

- 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.

- Not only natural language text is considered as unstructured information. Video 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, p. 4).

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” (Grishman, 1997, p. 10).

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, p.2).

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 particular report we 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 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 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 characterised 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).

### 1.3 Information Extraction Systems Evaluation

Message Understanding Conferences or Message Understanding Competitions (MUCs) have played an important role in the development of information extraction as a field of study. This conference was initiated by the United States Naval Ocean Systems Centre (NOSC) and was sponsored by the Defence Advanced Research Project Agency (DARPA). MUCs took place seven times from 1987 until 1998. Although the event is called a “conference”, it can be described with other words like “competition” between information extraction research groups or “evaluation” of their systems’ performances (Grishman and Sundheim, 1996; Cowie and Lehnert, 1996; Turmo et al., 2006).

The major aim of these conferences was the evaluation of the state-of-the-art in the information extraction area, discovery and promotion of the new approaches in this field. However, Grishman and Sundheim (1996) claimed that MUCs differed from any other conferences in the way how the research groups were selected in order to take part in those conferences. The evaluation procedure started approximately 6 months before each conference. The research teams were given the same task to extract particular information from the sample texts. From conference to conference the tasks and domains to extract
information from were changed. The research groups had to develop information extraction systems to accomplish those tasks. Just before the conference the participants were given a number of test texts to be processed using their systems. Then the obtained results from researchers were evaluated and compared with the pattern which had been gained previously by hand (Grishman and Sundheim, 1996; Turmo et al., 2006).

Since MUCs were launched by NOSC, the general domain for the first conference was Naval Tactical Operations. However, the selection process described above was not the case for MUC-1. The exact task and format for the results were not specified preliminarily and the systems were not evaluated on the common base. Starting from MUC-2, the selection scheme was followed precisely and different domains were explored, such as reports about Joint Ventures, Terrorist Attacks and Airline Crashes. For the second conference, the same area of military messages as for the first one was chosen but the particular task identified by the organisers was to fill in a template with 10 slots for information to be extracted. From conference to conference, new tasks were introduced and they became more complex; the number of slots to be filled in increased constantly and texts in Japanese language were used alongside documents in English (Turmo et al., 2006).

Another issue in which the second MUC played an important role concerns establishing the evaluation measures for information extraction systems. The conferences showed that information extraction is not an easy task, as it is very difficult to create a system with an accuracy level of 100%. This means there is always relevant information in the text which is not extracted and extracted entities in the slots which are not relevant to the task. Rodriguez-Esteban (2009) compares the process of information extraction with manual search for needles in a haystack. After one attempt there are some needles as well as straws in a hand and there are some needles left in the haystack. To evaluate information extraction processes there are two metrics, namely, Precision and Recall. In simple terms, Precision ($P$) is the proportion of correctly extracted entities ($N_{correct}$) to the total number of extracted entities ($N_{response}$) (the ratio between number of needles in a hand and number of needles and straws in the hand). Recall ($R$) is the proportion of correctly extracted entities ($N_{correct}$) to the total number of entities which are extracted manually ($N_{key}$) (the ratio between number of needles in the hand and total number of them in the haystack). Thus,

$$P = \frac{N_{correct}}{N_{response}},$$

$$R = \frac{N_{correct}}{N_{key}}.$$  

Another way of representation of information extraction systems evaluation is based on the notion of true and false positives and true and false negatives. It can be said that correctly extracted entities are true positives, whereas false positives are wrongly extracted information. Similarly, false negatives are relevant but not extracted information which is left in the text; true negatives are the information which is not extracted and not relevant to the task (Grishman, 1997; Appelt and Israel, 1999).

Buckland and Gey (1994) studied the relationship between precision and recall in the information retrieval field. However, their findings can be applied to the information extraction area as well. As they stated, recall can be described as the measure of extraction effectiveness, whereas precision is the measure of extraction purity. Both of them are desired to be high. However, they are mutually dependent. If one of the metrics is increasing it leads
to the another metric decreasing and this trade-off is unavoidable. For the information retrieval process Buckland and Gey (1994) examined a two-stage strategy when the second retrieval step is performed on the subset which was obtained in the first step. The initial step is aimed at high recall, while the next one improves precision.

However, the approach described above is not suitable for information extraction. It is impossible to apply the same rules a second time to once extracted information, in other words, to filled templates. And there is no point to apply the same rules twice on the initial texts, since the result will not change.

In order to combine precision and recall, the $F$ measure was introduced in one of the MUCs. Thus,

$$F = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}, \quad 0 < \beta \leq 1.$$

It is the harmonic mean of two metrics which allows comparing and assessing different information extraction systems using one common base. Different values for $\beta$ are used to, e.g., favour precision over recall (Appelt and Israel, 1999; Turmo et al., 2006).

The work that had been done through all the MUCs led to the formulation and introduction of basic extraction tasks. MUC-6 and MUC-7 contributed the most to this process. The Named Entity Recognition (NER) task is the first step of any information extraction system which involves proper names and quantities identification. The techniques used to accomplish this task are well-understood now and NER can be considered as a more or less “solved problem”. The Template Element task (TE) is the next step to identify not only names but the descriptions of those names as well. The Template Relationship (TR) task implies finding the relationships between the entities extracted during the previous tasks. The Scenario Template (ST) task is based on the extraction according to the description of the particular event. The goal of the final Coreference task (CO) is to determine all the nouns, pronouns and noun phrases that refer to the same entity (Grishman and Sundheim, 1996; Turmo et al., 2006; Feldman and Sanger, 2007).

After the last Message Understanding Conference 7 in 1998, the evaluation of information extraction systems has not stopped. MUC has been followed by the Automatic Content Extraction programme since 1999. However, ACE is not just a copy of MUC; it differs from its predecessor in the following several ways (Doddington et al., 2004):

- ACE defined 3 main extraction tasks different from those of MUC. The tasks are: Entity Detection and Tracking, Relation Detection and Tracking, Event Detection and Characterisation. It must be mentioned that the first task involves the extraction not only of the name of an entity but anything that refers to that name, such as a description or a pronoun. That is why it is possible to say that the Entity Detection and Tracking task has combined the Named Entity Recognition and the Coreference Tasks of MUC.

- Apart from the text sources in English language, texts in Arabic and Chinese languages are processed as well.

- Not only text documents but audio and image data are used to extract information from.
• Until 2008 the evaluation results had not been published. In 2008 the official results of ACE were made publicly available for the first time.

• The systems are evaluated using a Value measure which shows the correctly detected and recognised objects and their attributes. It is applied for all of the tasks and target objects, namely, entities, relations and events.

Analysing the last available official ACE results some conclusions can be drawn. There were four tasks introduced both in English and Arabic languages, namely Local Entity Detection and Recognition (LEDR), Global Entity Detection and Recognition (GEDR), Local Relation Detection and Recognition (LRDR), Global Relation Detection and Recognition (GRDR). “Local” defines the task within one document, whereas “global” implies the same task but across all the documents processed. Approximately 11,000 documents were available in English language from 7 domains and 10,000 documents in Arabic language from 6 domains. 10 research groups were registered to participate in the evaluation, but none of them fulfilled all the tasks introduced. Therefore, it is difficult to compare the performance of particular systems across all the tasks. However, the general conclusions can be made.

• The overall performance of the systems in English language is higher than in Arabic. It might be explained by the fact that much more research has been done on English than on Arabic in the information extraction area.

• Entity extraction both local and global was performed more successfully than relation extraction. This is natural since entity recognition task is more straightforward.

• It cannot be said that results achieved on the local tasks were higher than on the global ones. However, in GRDR in Arabic language there was no positive evaluation among all sites which tried to carry out that task.

The best performance was obtained by BBN Technologies in GEDR task in English language on Multi-domain with Value measure of 66.4%. The best result (Value = 53%) across all the domains was gained by BBN Technologies in GEDR task in English language.

1.4 Two Approaches

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. Analysing 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 unlabelled 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. Such considerations have led to the development of evaluation frameworks such as U-Compare (http://u-compare.org).

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 analysing 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 does not involve a stage of choosing an approach. Since one of the main goals of the project is writing a set of extraction rules for a specific domain the question which method to prefer does not arise.

1.5 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, analysing those different approaches the general pipeline of the information extraction process can be summarised. In the current work six main stages were determined as following:

1. Initial processing.
2. Proper names identification.
3. Parsing.
4. Extraction of events and relations.
5. Anaphora resolution.
6. 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 tokenisers, text zoners, segmenters or splitters. As Appelt and Israel (1999) stated, tokenisation 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 specialised 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 organisations, 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 recognised 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 recognised 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 realised 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 realised 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 labelled 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.** Despite the fact that this problem was firstly introduced and evaluated on the MUC-6 as the coreference task (CO), before the MUC-6 coreference presented as a challenge and research groups tried to resolve it, although implicitly.

Any 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 coreference resolution is performed. Coreference or anaphora resolution is the stage when for noun phrases it is determined if they refer to the same entity or not. There are several types of coreference, but the most common types are pronominal and proper names coreference, 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 normalisation operations for dates, time, currencies, etc. For instance, a round-off procedure for percentages can be executed and a real number 75.96 will be turned into integer 76 (Hobbs, 1993; Turmo et al., 2006).

Not all of the tasks must me 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:

- **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.

- **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.

- **Extraction task.** For an easy task like names recognition the parsing and anaphora resolution modules might not be needed at all.

### 1.6 Comparison of the Information Extraction Systems

Information Extraction is a comparatively new field of study. However, during this short period there are many information extraction systems have been created by different research groups. Moreover, the fact that the IE systems have been assessed in the past in MUC conferences and are now constantly evaluated in ACE competitions pushes them to refine their systems or develop new ones. At some point a need to look at all those systems as a whole and compare them emerges. There have been several attempts to conduct a survey of the information extraction systems (Muslea, 1999; Kuhlins and Tredwell, 2002; Laender et al., 2002; Kaiser and Miksch, 2005; Seifkes and Siniakov, 2005; Chang et al., 2006). All of them use a wide range of different criteria to make comparisons, like an approach that is used to extract information or a type of output results that systems produce.

One of the comparison criteria of frequent occurrence is a type of source data that a system can handle. On the basis of that characteristic Muslea (1999) divides the IE systems in those which extract information from free text, from online documents and wrapper induction systems which process Web pages as a source. Free text is considered to contain plain grammatical sentences. The systems that take such a text as an input (AutoSlog, LIEP, PALKA, CRYSTAL) are able to use only the natural language processing techniques to create extraction rules either manually or from the learning processes. Wrappers (WIEN, SoftMealy, STALKER), in contrast, are not constrained by only linguistic approaches and can take advantage of predefined HTML templates which implicitly classify the data found in a
Online documents are defined as a combination of free, grammatical and ungrammatical texts. Therefore, the systems which process that type of source data (WHISK, RAPIER, SRV) use a mixture of techniques based on the linguistics and structural features of documents.

Web information extraction is a study area that has emerged within the framework of traditional information extraction. It considers only Web pages as a source data and applies delimiter-based rules while extracting entities. This field has gained a lot of interest recently and that is why several authors survey only Web information extraction systems (Kuhlins and Tredwell, 2002; Laender et al., 2002; Chang et al., 2006).

Another commonly used criterion to system categorisation is an approach for rule development. As defined earlier there are two main methods for rule creation, namely, knowledge engineering and machine learning or, in another interpretation, manual and automatic pattern discovery respectively. The latter can be divided into supervised, semi-supervised and unsupervised approaches. As Kaiser and Miksch (2005) argue, manual pattern discovery was used at the earlier periods of information extraction era. The representatives of this class of systems are FASTUS, GE NLTSET, PLUM, PROTEUS, etc. Now, according to them, information extraction automation has become more popular due to some restrictions of the previous approach, like time and effort consumption. Among the automated systems are WHISK, RAPIER, WIEN, SRV (supervised); IEPAD, OLERA (semi-supervised); DeLa, RoadDunner, DEPTA (unsupervised) (Kaiser and Miksch, 2005; Chang et al., 2006).

Automation degree and user expertise are the parameters that strongly correlate with the approach used for pattern discovery. Thus, in the manual IE systems there is no automation performed and user’s rule writing skills are required. Supervised and semi-supervised systems are semi-automatic and, therefore, a user needs to have the text annotation experience. Finally, unsupervised systems are fully automatic and only a pattern selection task is left to be done by a user (Laender et al., 2002; Chang et al., 2006).

Seifkes and Siniakov (2005) discuss in more detail the exact methods for rule creation within each approach. For instance, they distinguish horn clauses, ontology-based and thesaurus-based techniques which are related to the knowledge engineering approach. Pattern and template creation, case-based methods, relational and covering algorithms are considered to be among the rule learning approaches.

Type of task handled is another criterion to classify IE systems. Single-slot extraction is the first step of the overall process which means filling the slots within the templates correctly. Multi-slot extraction implies templates generation which requires coreference resolution to be performed. Most of the systems stop at the first step providing only the single-slot extraction, since it is an easier task (SRV, STALKER, RAPIER, LIEP, AutoSlog etc.). Others try to execute discourse analysis and perform templates creation at the sentence level (Crystal, WHISK, TIMES, etc.) or within the whole text (IE², SIFT, SNoW-IE) (Laender et al., 2002; Seifkes and Siniakov, 2005; Kaiser and Miksch, 2005).

XML now is as a standard for data representation. That is why whether an extraction system produces results in an XML format or not is regarded as another important classifier for IE systems by many authors (Kuhlins and Tredwell, 2002; Laender et al., 2002; Chang et al., 2006). Those systems that have such a feature are Minerva, XWrap, NoDoSE, SoftMealy, RoadRunner, LAPIS, DEByE. Other possible output formats for extraction process are text, object exchange model (OEM) and SQL database output format. Some systems support
several formats like NoDoSE (XML, OEM), DEByE (XML, SQL DB, and text), RoadRunner (XML, text).

In addition to the parameters mentioned above there are many other characteristics that can be used to compare information extraction systems. However, at this point some issues emerge that must be taken into consideration. Firstly, as Chang et al. (2006) state, IE systems are usually created to perform different extraction tasks; therefore a straightforward comparison cannot be applied to IE system analysis. Secondly, the major aim of the IE system design is to extract the information relevant to a client’s query. Thereby, one of the most significant comparison criteria must be the quality of the task performed by those systems. Under the circumstances the competitions like MUC and ACE which evaluate IE systems on the basis of common tasks and measures seem to be a reasonable ground for making those comparisons.

1.7 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 realised 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:

- Reuse the tools for natural language processing and text mining which have been previously created by other developers.

- Quickly combine different tools and thereby analyse 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 Initialiser 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, a 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.

### 1.8 Summary

Information extraction is a relatively new area of study. However, as any information technology it advances quite quickly and a great progress has been made from the time it appeared. Texts from different domains were processed within the MUC and ACE competitions and the performance, for example, in the named entity recognition task has reached higher than 90% level. The process of the information extraction system design has changed from independent development of a system for a particular task from scratch to application of architectures like UIMA and GATE which allow using the previously created components and combining them easily.

However, there are still many unsolved problems. Event extraction task, for instance, cannot be executed as yet with as high level of performance as named entity recognition. And domain-independent information extraction systems are still one of the big research issues.
CHAPTER 2. RESEARCH METHODS AND PROJECT PLAN

This chapter introduces the main concepts according to which the development of the information extraction systems will be carried out. The aim and the objectives are defined, the research methodology is chosen and the project plan is presented.

2.1 The Aim of the Project, Its Objectives and Deliverables

The current information extraction project is placed among those which involve writing extraction rules according to the knowledge engineering approach. Those rules are used to execute the task to extract information corresponding to a user’s need from a set of texts. Therefore, the high level aim of this project can be stated as follows: to gain a deep understanding of information extraction field by creating a system which extracts information relevant to the user’s need from a number of unstructured texts.

The following are the limitations involved in the project:

1. 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.

2. The data source for the information to be extracted from is the free unstructured texts with plain, grammatical sentences in English language.

3. The time allocated for the project to be completed is four months part-time and three months full-time. The project is done by a novice in the information extraction area.

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

1. Study the state of the art in the information extraction field, the approaches for system design and evaluation methods.

2. Choose the domain of texts the information to be extracted and define the template(s) with a number of slots to be filled in.

3. Familiarise ourselves with the formalism of the system which will be used to develop extraction rules.

4. Explore the gazetteers provided by the system and create the new ones if needed.

5. Write and test the extraction rules.

6. Evaluate the level of performance calculating Precision, Recall and F measure.

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

- Produce only the outline of the system’s specifications which can be modified later but still serves as a guide for developers.
- Develop the first prototype of the software according to those initial requirements.
- 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).

Within the current project 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](image)

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.

2.2.1 System requirements

Information extraction can be applied to a wide range of text domains. As we can see from the MUC experience domains vary from Joint Ventures from business news 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 Earthquakes News Reports domain has been chosen. The text will be taken from the http://news.yahoo.com/topics/earthquakes site. The information extracted can be used then, for instance, to analyse areas with the most frequent earthquakes, the periodicity of the earthquakes in a particular region, or the magnitude. The example of the texts which will be processed is given below.

“Minor earthquake hits Tennessee,KNOXVILLE, Tenn., April 20 (UPI)

A 3.3-magnitude earthquake hit Blount County, Tenn., Tuesday morning, the U.S. Geological Survey said. The temblor's depth was at 3.1 miles with its epicenter southwest of downtown Maryville, Tenn., WBIR-TV, Knoxville, reported. No damage was reported. The rumble was felt across far western North Carolina, eastern Tennessee and northern Georgia, the Charlotte (N.C.) Observer said. Temblors in western North Carolina and eastern Tennessee are known to occur but the majority have magnitudes of less than 4.0, the Observer said.”

The names entities that will be extracted are place, date, time, magnitude, number of people affected, damage caused.

2.2.2 Implementation and Testing

CAFETIERE is a system which will be used within the current project. CAFETIERE is an abbreviation for Conceptual Annotations for Facts, Events, Terms, Individual Entities and Relations. It is a Web-based system which follows the knowledge engineering approach. Each text goes through the following stages of processing within the information extraction chain, which are depicted on the Figure 3.

![Figure 3 – The CAFETIERE stages of information extraction process](image)

Plain text as well as marked up in HTML or SGML documents can taken 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
Gazetteer lookup means words are labelled 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 coreference resolution (Black et al., 2005).

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 fulfil 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.

### 2.2.3 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.

### 2.3 Project Plan

Figure 2 depicts the Gantt chart of the dissertation project plan.

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Days</th>
<th>Start</th>
<th>Finish</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
<th>July</th>
<th>August</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dissertation Project</td>
<td>160</td>
<td>01/02/10</td>
<td>10/09/10</td>
<td>1</td>
<td>8</td>
<td>15</td>
<td>22</td>
<td>1</td>
<td>8</td>
<td>15</td>
</tr>
<tr>
<td>Background research</td>
<td>43</td>
<td>01/02/10</td>
<td>31/03/10</td>
<td>0.5</td>
<td>0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write up of background report</td>
<td>50</td>
<td>01/03/10</td>
<td>07/03/10</td>
<td>0.16</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Report submission</td>
<td>5</td>
<td>10/05/10</td>
<td>14/05/10</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exam period</td>
<td>20</td>
<td>17/05/10</td>
<td>31/05/10</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learning the system's formalism</td>
<td>13</td>
<td>14/06/10</td>
<td>05/06/10</td>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Writing extraction rules</td>
<td>37</td>
<td>01/07/10</td>
<td>20/07/10</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing the system</td>
<td>37</td>
<td>01/07/10</td>
<td>20/07/10</td>
<td>0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Write up of dissertation</td>
<td>42</td>
<td>01/07/10</td>
<td>03/09/10</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>System evaluation</td>
<td>20</td>
<td>23/08/10</td>
<td>02/09/10</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dissertation submission</td>
<td>5</td>
<td>06/09/10</td>
<td>10/09/10</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2 – Project plan**

The project can be divided by the examination period into two parts. The first part of the project is comprised of the background research and the current report. The second part is the system implementation which includes the rest of the tasks. The “Days” column shows the number of days that will be allocated to the execution of the particular task. The numbers in yellow blocks specify the effort in persons per week that will be put to fulfil that task. Before the examination period the half of the time students spend working on rest of the modules. That is why the time allocated for the project during that period is 0.5 person per week.
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 history of the MUC and ACE conferences has been traced with providing the information extraction systems evaluation framework. 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. All the stages of the development are described and project plan in the form of Gantt chart is given.
LIST OF REFERENCES


