IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 27-31
www.iosrjournals.org

A Survey on Approaches to Text Mining using Information Extraction

Mrs. Rashmi Dukhi[1], Ms. Antara Bhattacharya[2]
[1]*(rashmi.dukhi@raisoni.net, MCA,GHRIIT/ RTMNU, India)*
[2]*(antarab@rediffmail.com ,CSE,* GHRIETW */ RTMNU, India)*

Abstract**:** *Information extraction is an important text mining problem and has been extensively studied in areas such as natural language processing, information retrieval and Web mining. Information Extraction (IE) techniques aim to extract the names of entities and objects from text and to identify the roles that they play in event descriptions. IE systems generally focus on a specific domain or topic, searching only for information that is relevant to a user's interests. Information extraction technology focuses on only the relevant parts of the text and ignores the rest. Information Extraction is the mapping of natural language texts (such as newswire reports, newspaper and journal articles, electronic mail, World Wide Web pages, any textual database, etc.) into predefined, structured representation, or templates, which, when filled, represent an extract of key information from the original text. The information concerns entities of interest in the application domain (e.g. companies or persons), or relations between such entities, usually in the form of events in which the entities take part (e.g. company takeovers, management successions etc.). Once extracted, the information can then be stored in databases to be queried, data mined, summarized in natural language, etc. In this paper various steps involved in information extraction is discussed. In this paper we describe Named entity recognition (NER) which is one of the most common uses of information extraction technology. Named entity recognition aims at finding names of entities such as people, organizations and locations while Relation extraction is the task of finding the semantic relations between entities from text.*

Keywords- *Automatic Content Extraction, corpus , Named entity recognition ,semantic, tokenizer.*

I. Introduction

Information Extraction (IE), is one of the most prominent techniques currently used in Text Mining. In particular, by combining Natural Language Processing tools, lexical resources and semantic constraints, it can provide effective modules for mining the biomedical literature, or to help in preventing terrorism. Complementary visualization tools enable the user to explore, check (and correct if required) the results of the Text Mining process effectively.

As a first step in tagging documents, each document is processed to find (extract) Entities and Relationships that are likely to be meaningful and content-bearing. In "Relationships" we refer to Facts or Events involving certain Entities. A possible "Event" may be that a company has entered into a joint venture to develop a new drug. A "Fact" may be that a gene causes a certain disease. Facts are static in nature and usually do not change; events are more dynamic in nature and have a specific time stamp associated with them. The extracted information provides more concise and precise data for the mining process than the more naive word-based approaches such as those used for text categorization, and tends to represent concepts and relationships that are more meaningful and relate directly to the examined document's domain. For **unstructured data** ,we first convert the **unstructured data** of natural language sentences into the structured data.Then we reap the benefits of powerful query tools such as SQL. This method of getting meaning from text is called **Information Extraction**.

Information extraction from text is an important task in text mining. The general goal of information extraction is to discover structured information from unstructured or semi-structured text. For example, given the following English sentence,
In 1998, Larry Page and Sergey Brin founded Google Inc.
we can extract the following information,
FounderOf(Larry Page, Google Inc.),
FounderOf(Sergey Brin, Google Inc.),
FoundedIn(Google Inc., 1998)

Such information can be directly presented to an end user, or more commonly, it can be used by other computer systems such as search engines and database management systems to provide better services to end users. Information extraction has applications in a wide range of domains. The specific type and structure of the information to be extracted depend on the need of the particular application. Traditionally information extraction tasks assume that the structures to be extracted, e.g. the types of named entities, the types of relations, or the template slots, are well defined. In some scenarios, we do not know in advance the structures of the information we would like to extract and would like to mine such structures from large corpora. For example, from a set of earthquake news articles we may want to automatically discover that the date, time, epicenter, magnitude and casualty of an earthquake are the most important pieces of information reported in news articles. There have been some recent studies on this kind of unsupervised information extraction problems but overall work along this line remains limited.

II. Information Extraction Architecture

Figure 2.1 shows the architecture for a simple information extraction system. It begins by processing a document using several of the procedure: first, the raw text of the document is split into sentences using a sentence segmenter, and each sentence is further subdivided into words using a tokenizer. Next, each sentence is tagged with part-of-speech tags, which will prove very helpful in the next step, **named entity detection**. In this step, we search for mentions of potentially interesting entities in each sentence. Finally, we use **relation detection** to search for likely relations between different entities in the text

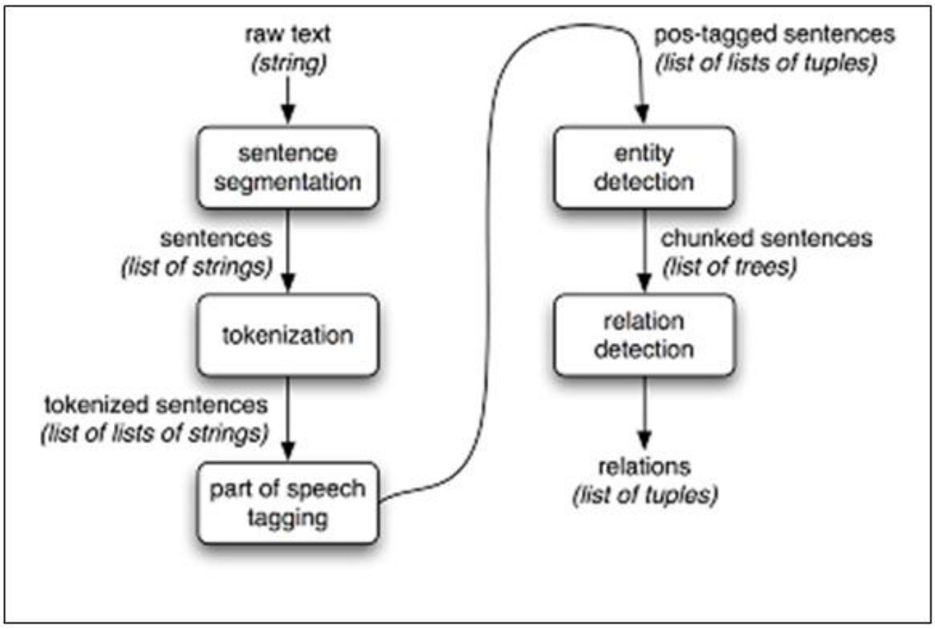

Fig 2.1: Simple Pipeline Architecture for an Information Extraction System.

2.1 Tokenizer

Tokenizers is used to divide strings into lists of substrings. For example, Sentence tokenizer can be used to find the list of sentences and Word tokenizer can be used to find the list of words in strings.

2.2 Part-of-speech tagging

Part-of-speech tagging is one of the most important text analysis tasks used to classify words into their part-of-speech and label them according the tagset which is a collection of tags used for the pos tagging. Part-of-speech tagging also known as word classes or lexical categories.In corpus linguistics, part-of-speech tagging (POS tagging or POST), also called grammatical tagging or word-category disambiguation, is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition, as well as its context—i.e. relationship with adjacent and related words in a phrase, sentence, or paragraph. A simplified form of this is commonly taught to school-age children, in the identification of words as nouns, verbs, adjectives, adverbs, etc.

Once performed by hand, POS tagging is now done in the context of computational linguistics, using algorithms which associate discrete terms, as well as hidden parts of speech, in accordance with a set of descriptive tags. POS-tagging algorithms fall into two distinctive groups: rule-based and stochastic. E. Brill's tagger, one of the first and most widely used English POS-taggers, employs rule-based algorithms.

2.3 Named Entity Recognition

A named entity is a sequence of words that designates some real world entity, e.g. “California,” “Steve Jobs” and “Apple Inc.” The task of named entity recognition, often abbreviated as NER, is to identify named entities from free-form text and to classify them into a set of predefined types such as person, organization and location. Oftentimes this task cannot be simply accomplished by string matching against pre-compiled gazetteers because named entities of a given entity type usually do not form a closed set and therefore any gazetteer would be incomplete. Another reason is that the type of a named entity can be context-dependent. For example, “JFK” may refer to the person “John F. Kennedy,” the location “JFK International Airport,” or any other entity sharing the same abbreviation. To determine the entity type for “JFK” occurring in a particular document, its context has to be considered. Named entity recognition is probably the most fundamental task in information extraction. Extraction of more complex structures such as relations and events depends on accurate named entity recognition as a preprocessing step. Named entity recognition also has many applications apart from being a building block for information extraction. In question answering, for example, candidate answer strings are often named entities that need to be extracted and classified first [6]. In entity-oriented search, identifying named entities in documents as well as in queries is the first step towards high relevance of search results [5, 3].

The most commonly studied named entity types are person, organization and location, which were first defined by MUC-6. These types are general enough to be useful for many application domains. Extraction of expressions of dates, times, monetary values and percentages, which was also introduced by MUC-6, is often also studied under NER, although strictly speaking these expressions are not named entities. Besides these general entity types, other types of entities are usually defined for specific domains and applications. For example, the GENIA corpus uses a fine-grained ontology to classify biological entities [9]. In online search and advertising, extraction of product names is a useful task.

2.3.1 Rule-based Approach

Rule-based methods for named entity recognition generally work as follows: A set of rules is either manually defined or automatically learned. Each token in the text is represented by a set of features. The text is then compared against the rules and a rule is fired if a match is found. A rule consists of a pattern and an action. A pattern is usually a regular expression defined over features of tokens. When this pattern matches a sequence of tokens, the specified action is fired. An action can be labeling a sequence of tokens as an entity, inserting the start or end label of an entity, or identifying multiple entities simultaneously. For example, to label any sequence of tokens of the form “Mr. X” where X is a capitalized word as a person entity, the following rule can be defined:

(token = “Mr.” orthography type = FirstCap) → person name.

The left hand side is a regular expression that matches any sequence of two tokens where the first token is “Mr.” and the second token has the orthography type FirstCap. The right hand side indicates that the matched token sequence should be labeled as a person name.

This kind of rule-based methods has been widely used [1, 8, 2, 7, 4]. Commonly used features to represent tokens include the token itself, the part-of-speech tag of the token, the orthography type of the token (e.g. first letter capitalized, all letters capitalized, number, etc.), and whether the token is inside some predefined gazetteer. It is possible for a sequence of tokens to match multiple rules. To handle such conflicts, a set of policies has to be defined to control how rules should be fired. One approach is to order the rules in advance so that they are sequentially checked and fired.

Manually creating the rules for named entity recognition requires human expertise and is labor intensive. To automatically learn the rules, different methods have been proposed. They can be roughly categorized into two groups: top-down (e.g. [7]) and bottom-up (e.g. [2, 4]). With either approach, a set of training documents with manually labeled named entities is required. In the top-down approach, general rules are first defined that can cover the extraction of many training instances. However, these rules tend to have low precision. The system then iteratively defines more specific rules by taking the intersections of the more general rules. In the bottom-up approach, specific rules are defined based on training instances that are not yet covered by the existing rule set. These specific rules are then generalized.

2.3.2 Statistical Learning Approach

More recent work on named entity recognition is usually based on statistical machine learning. Many statistical learning-based named entity recognition algorithms treat the task as a sequence labeling problem. Sequence labeling is a general machine learning problem and has been used to model many natural language processing tasks including part-of-speech tagging, chunking and named entity recognition. It can be formulated as follows. We are given a sequence of observations, denoted as x = (x1, x2,...,xn). Usually each observation is represented as a feature vector. We would like to assign a label yi to each observation xi. While one may apply standard classification to predict the label yi based solely on xi, in sequence labeling, it is assumed that the label yi depends not only on its corresponding observation xi but also possibly on other observations and other labels in the sequence. Typically this dependency is limited to observations and labels within a close neighborhood of the current position i. To map named entity recognition to a sequence labeling problem, we treat each word in a sentence as an observation. The class labels have to clearly indicate both the boundaries and the types of named entities within the sequence

2.4 Relation Extraction

Another important task in information extraction is relation extraction. Relation extraction is the task of detecting and characterizing the semantic relations between entities in text. For example, from the following sentence fragment, Facebook co-founder Mark Zuckerberg we can extract the following relation, FounderOf(Mark Zuckerberg, Facebook). Much of the work on relation extraction is based on the task definition from the Automatic Content Extraction (ACE) program. ACE focuses on binary relations, i.e. relations between two entities. The two entities involved are also referred to as arguments. A set of major relation types and their subtypes are defined by ACE. Examples of ACE major relation types include physical (e.g. an entity is physically near another entity), personal/social (e.g. a person is a family member of another person), and employment/affiliation (e.g. a person is employed by an organization). ACE makes a distinction between relation extraction and relation mention extraction. The former refers to identifying the semantic relation between a pair of entities based on all the evidence we can gather from the corpus, whereas the latter refers to identifying individual mentions of entity relations. Because corpus-level relation extraction to a large extent still relies on accurate mention-level relation extraction, in the rest of this chapter we do not make any distinction between these two problems unless necessary. Various techniques have been proposed for relation extraction. The most common and straightforward approach is to treat the task as a classification problem: Given a pair of entities co-occurring in the same sentence, can we classify the relation between the two entities into one of the predefined relation types? Although it is also possible for relation mentions to cross sentence boundaries, such cases are less frequent and hard to detect. Existing work therefore mostly focuses on relation extraction within sentence boundaries.

III. Applications

Some example applications of information extraction below:

- Biomedical researchers often need to sift through a large amount of scientific publications to look for discoveries related to particular genes, proteins or other biomedical entities. To assist this effort, simple search based on keyword matching may not suffice because biomedical entities often have synonyms and ambiguous names, making it hard to accurately retrieve relevant documents. A critical task in biomedical literature mining is therefore to automatically identify mentions of biomedical entities from text and to link them to their corresponding entries in existing knowledge bases such as the FlyBase.
- Financial professionals often need to seek specific pieces of information from news articles to help their day-to-day decision making. For example, a finance company may need to know all the company takeovers that take place during a certain time span and the details of each acquisition. Automatically finding such information from text requires standard information extraction technologies such as named entity recognition and relation extraction.
- Intelligence analysts review large amounts of text to search for information such as people involved in terrorism events, the weapons used and the targets of the attacks. While information retrieval technologies can be used to quickly locate documents that describe terrorism events, information extraction technologies are needed to further pinpoint the specific information units within these documents.
- With the fast growth of the Web, search engines have become an integral part of people’s daily lives, and users’ search behaviors are much better understood now. Search based on bag-of-word representation of documents can no longer provide satisfactory results. More advanced search problems such as entity search, structured search and question answering can provide users with better search experience. To facilitate these search capabilities, information extraction is often needed as a preprocessing step to enrich document representation or to populate an underlying database.

IV. Conclusion

Information extraction is an important text mining problem and has been extensively studied in areas such as natural language processing, information retrieval and Web mining. In this chapter we reviewed some representative work on information extraction, in particular work on named entity recognition and relation extraction. Named entity recognition aims at finding names of entities such as people, organizations and locations. State-of-the-art solutions to named entity recognition rely on statistical sequence labeling algorithms such as maximum entropy Markov models and conditional random fields. Relation extraction is the task of finding the semantic relations between entities from text.

With the fast growth of textual data on the Web, it is expected that future work on information extraction will need to deal with even more diverse and noisy text. Weakly supervised and unsupervised methods will play a larger role in information extraction. The various user-generated content on the Web such as Wikipedia articles will also become important resources to provide some kind of supervision.

REFERENCES

Journal Papers:

[1]. Chia-Hui Chang, Mohammed Kayed, Moheb Ramzy Girgis, and Khaled F. Shaalan. A survey of Web information extraction systems. *IEEE Transactions on Knowledge and Data Engineering*,18(10):1411–1428, October 2006.

Books:

[2]. C.C. Aggarwal and C.X. Zhai (eds.), *Mining Text Data* (Springer Science+ Business Media, LLC 2012)

Proceedings Papers:

[3]. Douglas E. Appelt, Jerry R. Hobbs, John Bear, David Israel, and Mabry Tyson. FASTUS: A finite-state processor for information extraction from real-world text. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1993.

[4]. Mary Elaine Califf and Raymond J. Mooney. Relational learning of pattern-match rules for information extraction. In *Proceedings of the 16th National Conference on Artificial Intelligence and the 11th Innovative Applications of Artificial Intelligence Conference,* pages 328–334, 1999.

[5]. Tao Cheng, Xifeng Yan, and Kevin Chen-Chuan Chang. Supporting entity search: a large-scale prototype search engine. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, pages 1144–1146, 2007.

[6]. Fabio Ciravegna. Adaptive information extraction from text by rule induction and generalisation. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, pages 1251–1256, 2001.

[7]. Guoping Hu, Jingjing Liu, Hang Li, Yunbo Cao, Jian-Yun Nie, and Jianfeng Gao. A supervised learning approach to entity search. In *Proceedings of the 3rd Asia Information Retrieval Symposium*, pages 54–66, 2006.

[8]. Xin Li and Dan Roth. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics*, pages 1–7, 2002.

[9]. Stephen Soderland. Learning information extraction rules for semistructured and free text. *Machine Learning*, 34(1-3):233–272, February 1999.

[10]. Stephen Soderland, David Fisher, Jonathan Aseltine, and Wendy Lehnert. CRYSTAL inducing a conceptual dictionary. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1314–1319, 1995.

[11]. Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. The GENIA corpus: an annotated research abstract corpus in molecular biology domain. In *Proceedings of the 2nd International Conference on Human Language Technology Research*, pages 82–86, 2002.

[12]. Fei Wu and Daniel S. Weld. Open information extraction using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 118–127, 2010.