International Journal Of Mechanical Engineering And Information Technology
An Official Publication Of IGM Publication

Survey of Stages of Developing the Information Extraction Systems from the Web

DOI: http://dx.doi.org/10.18535/ijmeit/v3i11.01

Author
Asmaa Ahmed Hamed Khalil Elsaeidy
9/5 Young Street, Queanbeyan, NSW 2620 Australia
Email- *asmaa.hamed0@yahoo.com*

ABSTRACT:
Extracting useful information from World Wide Web is an important and challenging problem. Information Extraction (IE) task is an interesting area that is used in getting a useful information. The traditional information extraction systems are focused on satisfying precise, narrow and pre-specified requests from small homogenous corpora. Applying these systems on another domain or a large scale heterogeneous corpus is a complex task. Information Extraction has traditionally relied on extensive human involvement in the form of hand-crafted extraction rules or hand-tagged training examples. The main contribution is how to help the user to extract relevant information from different and changeable web pages and integrate this extracted information into a single structured file automatically. Social networks play an important role in the semantic web. According to the intention of utilizing the social networks for the semantic web, several studies have been examined the automatic information extraction from social networks. This survey explores the different information extraction methods, tasks, applications, system development and how to evaluate their performance. In addition to introduce a view on the most used approaches in extracting information from social networks.

INTRODUCTION

Information extraction task can be viewed from several different views. Some researchers are classified the extraction task according to the information type. The information source can be classified into three main types: free text, structured text and semi-structured text [(1), (2)]. Natural Language Processing (NLP) is used to extract the unrestricted or unregulated type of information like free text and SQL is used to query the structured data which usually is stored in databases [(1)]. The wrapper induction system is operated only on highly structured documents [(3)]. Web page is an example of semi structured text that this survey will explore. The field of information extraction from the web has emerged with the growth of the web. The web is constructed of a huge amount of pages and most of them are generated dynamically from structured data which provides a rich source of usable information to be extracted. The different presentations of information and different formats on each web site is performing a challenge in extraction process and integrating these different sources and formats in one single structured file. Two main markup languages are commonly used in structuring the web content, the Hyper Text Markup Language and Extensible Markup Language (XML). HTML is used to structure the web page content and the XML separates the data structure from layout and provides a much more suitable data representation. Some systems are constructed using XML instead of HTML due to the different presentations of data and different formats used in each web page such as Lixto [(4)]. Another important use of XML is for building the web services [(5)].

The key feature of the web is the redundancy of information which the information can be represent in several different formats on the web

from different sources. Most of current methodologies are based on human centred annotation and are often completely manual, so convincing users to annotate documents for the Web is difficult and required a world-wide action of uncertain outcome because the manual annotation is difficult, time consuming and expensive [6]. Recent researches have focused on extracting information with minimum user intervention. Static annotation associated to a document can be incomplete, incorrect, obsolete (not aligned with pages' updates) or irrelevant for some users. Although the information extraction activity is very complex task, decomposing it into several subtasks can be beneficial for IE main objective. The Information Extraction can be customized according to an application's needs by reordering, selecting and composing some of its tasks. There are some considered tasks like: segmentation, classification, association, normalization and co-reference resolution [7]. The IE system performance is depended on a set of criteria such as type of data (structured, semi structured or unstructured), the selected technique used (supervised, semi-supervised or unsupervised) [8], [2], [9] and the methodology that followed. Early researches have used rule-based approach which based on writing the code manually. This means the rules are generated by humans, and then the computer extracts the information from these generated rules. These hand coded methods cannot meet the increasing need of information extraction. Methods which are able to learn the rules using the machine learning approaches flood into the IE fields. However, researches still found that these rule-based IE methods are not adaptive enough if the input is a noisy unstructured data. Another trend is appeared that uses the statistical learning methods in IE. There are two main models which developed and used: Hidden Markov Models (HMM) and Conditional Models Wrapper Induction (CMWI). These are a special group of methods based on the fact that most web pages are generated by templates. Each method has its advantages and drawbacks, and a specific domain which is good at [10], [9].

Search engines is the common example for an information extraction process which used by the user. Search engine can retrieve a set of documents with a specific percent of precision and recall but it can't extract facts or assess confidence or fuse information from multiple documents. The goal of IE is to rank or select documents, to extract salient facts from the documents based on pre-specified types of events, entities, or relationships in order to build more meaningful rich representations of their semantic content. This extracted information can be used to populate databases that provide structured input for mining more complex patterns like trends and summaries from text collections [11]. Information extraction systems are automatically identified and extract factual information related to the events of interest. Also information extraction techniques may be used to learn informative clues of subjectivity [12]. Information extraction systems are targeted towards specific domains of interest and use either manual or semi-automatic learning of the target examples involved. In contrast, the goal of automatic information extraction is to discover the relations among data items of interest and similar data items on a large scale and independent from domain without any needed training [13]. There are several examples of such systems which are developed based on this concept like KNOWITALL [3], TEXTRUNNER [14] and the second generation of Open Information Extraction (OIE) [15]. Finally, several studies have addressed the information extraction from social networks automatically using artificial intelligence techniques and semantic web concepts [16]. In this paper a set of a popular approaches that have been used to extract information social networks are explored which plays a tremendous role in semantic web. This paper will provide a brief survey of IE, including definition, tasks, methods, general applications,

systems of IE and how it is evolved from traditional use to the open IE and the metrics in which these systems can be evaluated. The reminder of this paper is organized as follows, In Section 2, the definition of IE is discussed, In Section 3 IE tasks are explored, In Section 4 IE methods are explored, In Section 5 shows how IE task performance is evaluated, In Section 6 list of IE applications are explained, In Section 7 the components of IE system are explained, In Section 8 the role of IE in semantic web is defined. In Section 9 the work is concluded and what is the suggested future work can be.

WHAT IS INFORMATION EXTRACTION?

Information extraction is the task of automatically extracting structured information such as entities, relationships between entities, and attributes describing entities from unstructured and/or semi-structured machine-readable documents. In most of the cases this activity is focusing on processing human language texts by means of natural language processing [(17)]. Extracting structured information from unstructured text is a challenging problem; it attracts large amounts of researchers involving in this area. In addition to, information extraction is engaging many different fields including information retrieval, web and document analysis, database, and machine learning [(9).] IE technology didn't reach the market yet but it could be a great enhancement if the other organizations use it especially the financial companies, banks, publishers and governments. Named entity recognition (NER) is one of the most common uses of information extraction technology. NER system identifies different types of proper names, such as person and company names, and sometimes special types of entities such as dates and times that can be easily identified using surface-level textual patterns. NER is important in biomedical applications mainly in terminology problem. It is important to note that the information extraction is much more than just named entity recognition, but it is also concerned with recognition of events and their participants[(18)]. NER is probably the most fundamental task in information extraction. Extraction of complex structures such as relations and events is depending on accurate named entity recognition as a preprocessing step. NER has many applications apart from being a building block for information extraction. It is used in question answering. For example, candidate answer strings are often named entities that is needed to be extracted and classified first [(10), (11).]

Much of the technology in information extraction was developed in response to a series of evaluations and associated conferences called the Message Understanding Conference (MUC), held between 1987 and 1998. IE research has been stimulated by the Automatic Content Extraction (ACE) evaluations. The ACE evaluations have focused on identifying named entities, extracting isolated relations, and co reference resolution. The Linguistic Data Consortium (LDC) is developing annotation guidelines, corpora and other linguistic resources to support the ACE Program which are used to evaluate IE systems in a similar manner to conducting MUC competitions. Both MUC and ACE initiatives are of central importance to the IE field, since they provided a set of corpora that are available to the research community for the evaluation of IE systems and approaches [(11)].

INFORMATION EXTRACTION TASKS

A key component of any IE system is its set of extraction patterns or extraction rules that are used to extract information from each document that is relevant to a particular extraction task [(19)]. There are some fundamental distinctions that cut across different information extraction tasks such as, the text is structured or semi-structured, the information is extracted from single or multi-document, and a set of assumptions about the incoming document is built [(18)]. An Information Extraction activity can be very complex. Thus, it is common to decompose it into several tasks. This decomposition offers some advantages. First,

it is possible to choose, for each task, the techniques and algorithms that better fit the objective of a particular application. Second, it is easy to locally debug an IE program since the module that responsible for each task is completely independent from the others. Finally, an IE can be customized activity according to an application's needs, by reordering, selecting and composing some of the tasks [7]. The considered tasks are: segmentation, classification, association, normalization and co reference resolution. The Segmentation task divides the text into atomic elements, called segments or tokens.

The classification task determines the type of each segment obtained in the segmentation task. In other words, it determines the field of the output data structure where the input segment fits. The result of this task is the classification of a set of segments as entities which are elements of a given class potentially relevant for the extraction domain. The association task seeks for finding how the different entities found in the classification task can be related. Many techniques in the association task are rule-based. The simplest approach uses a set of patterns to extract a limited set of relationships. Normalization and co reference resolution are the less generic tasks of the information extraction process because they are using heuristics and rules that are specific to the data domain. The normalization task transforms information to a standard format defined by the user. Co reference arises whenever the same real world entity is referred in different ways in a text fragment. Rule-based approaches for co reference usually take into account semantic information about entities [11], [2].

INFORMATION EXTRACTION METHODS

Information extraction methods are categorized into two classes, rule-based methods and statistical method. In rule-based methods hard predicates are easier to interpret and develop. Any rule consists of pattern and 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 label a sequence of tokens like an entity by inserting the start or end of entity label or identifying multiple entities simultaneously [10]. On the other hand, statistical methods are based on a weighted sum of predicate firings and robust to noise in the unstructured data. A lot of early works have focused on the rule-based methods which are mostly useful in the closed domains which much of human work is involved. Some works have focused on statistical methods which are useful in the free-domain with less human involvement like extract opinion from online passage [9].

Wrapper induction is another method for IE. In the web environment, a wrapper can be defined as a processor that converts information that is implicitly stored in a document into information explicitly stored as a data structure for further processing. Wrapper induction is a special group of methods based on the fact that most web pages are generated by templates. Although there are several methods regarding to solve information extraction problem, there is no real winner among them for any type of problem. Each method has its advantages, drawbacks, and the specific domain it is suitable for. Rule-based approaches and statistical approaches are still improving to be important topics in IE field according to what reported in [9], [6], [4] and [20].

INFORMATION EXTRACTION EVALUATION

The extracted output from information extraction task is presented as a hierarchical attribute-value structure. Human annotators provide a set of key templates for the training data and the test data that is compared to the system's output. Values can be correct or incorrect according to its matching with the right target output. Attributes with non-empty values that not aligning with a key attribute are considered over-generation attributes [2], [11]. It is possible to define recall and precision scores for the output of an IE system

given the total possible correct responses (P), number of correct values (C), number of incorrect values (I), and over-generated values (O) as the following equations:

$$\text{Recall} = \frac{C}{P}$$

$$\text{Precision} = \frac{C}{C+I+O}$$

Recall measures the ratio of correct information extracted from the texts against all the available information present in the text and the precision measures the ratio of correct information that was extracted against all the information that was extracted. It is difficult to optimize both parameters at the same time. Besides these measures there is an F-score which use a metric to compare various IE systems by only one value. This metric uses weighted recall and precision scores depending on which value are giving more importance [2], [11].

$$F = \frac{(\beta^2+1)PR}{\beta^2 P+R}$$

The F measure is a geometric mean between recall and precision. By means of the parameter β it can be determined whether the recall (R) or the precision (P) score is weighted more heavily. Recall, precision, and F measure are the most frequently used metrics used in referring to an IE system's performance.

INFORMATION EXTRATION APPLICATIONS

The world is full of unstructured data that cannot be understood by machines. Information Extraction is an important task in text mining and has been extensively studied in various research communities including natural language processing, information retrieval and web mining. It has a wide range of applications in domains such as biomedical literature mining and business intelligence [10]. Extracting rules to be filled in a formatted data tables is considered as an application of IE. For example, the automated information extractor for a message is supposed to extract the name of the sender, topic of the message, date, important keywords, and content of the message [9]. Another application of IE is called Named Entity Recognition (NER) which refers to detect of reference for a particular objects like names of people, places, and companies. Today's enterprises are generating and consuming rapidly increasing quantities of unstructured textual data like emails, web pages, news articles, blog posts, online reviews and comments and call centre records. These types of enterprise applications should meet a set of requirements such as, scalability, accuracy, and usability. Prominent examples of these applications include [21]:

Semantic Search which extracts structured information from text documents and use this information as a metadata to enhance the accuracy of keyword searches.

Data as a Service which extracts and cleans the useful information hidden in publicly available documents by creating a valuable collection of structured data that can be rented or shared over the Internet.

Business Intelligence which mines the text data streams such as blog entries or call center records for information about consumer sentiment, product pipeline problems, or important economic events.

Data-driven Mashups which extracts structured information from unstructured feeds and join this structured information with other enterprise data to build a new data mashups.

The output of information extraction systems has been applied in several different types of applications like question-answering, review and opinion mining. Typically, these applications are search-oriented which a human user is a significant part of the process and can help to overcome mistakes or noise in the extraction process. Furthermore, the text-like extractions can be integrated into such applications relatively easily [22]. There are some other examples on the application where the information extraction can be applied such as, biomedical research, financial professionals, and intelligent analysis [10].

COMPONENTS OF INFORMATION EXTRACTION SYSTEMS

A typical information extraction system has several phases, these phases are input tokenization, lexical and morphological processing, some basic syntactic analysis, and identifying the information being sought in the particular application as shown in figure 1 [(2), (11), (20)]. Depending on the nature of task some phases may not be required. In addition to the modules in the left-hand column, IE systems may include some modules from the right-hand column that depends on the particular requirements of the application.

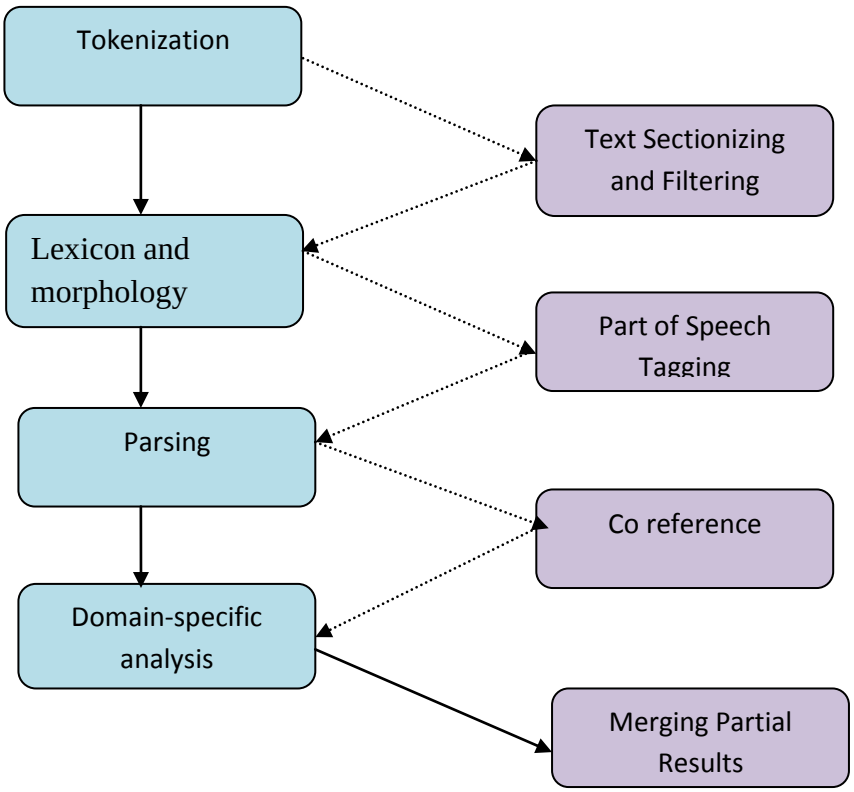

Figure 1 Modules of an Information Extraction System (2)

TRADITIONAL INFORMATION EXTRACTION

As mentioned before information extraction is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. In most of the cases this activity concerns processing human language texts by means of natural language processing (NLP). Traditional information extraction developed to mainly decrease the user intervention and automate the whole process. There is a set of challenges in information extraction task according to the semi structured type of information that found in web pages such as, ill formatting, schema lacking, high updated frequency and the semantic heterogeneity of the information [(1)]. Other challenges are described in [(14)] as the following:

Automation to launch the training process. The creation of suitable training data required substantial expertise as well as non-trivial manual effort for every relation extracted, and the relations have to be specified in advance.

Corpus heterogeneity: traditional IE systems are able to rely on heavy linguistic technologies tuned to the domain of interest, such as dependency parsers and Named-Entity Recognizers (NERs). Those systems were not designed to scale relatively to the size of the corpus or the number of relations extracted where both parameters were fixed and small.

Efficiency: according to the increasing number of web pages and the information that are changed dynamically, the extraction system must confirm on scalability and decreasing the time of the extracting all the pages that include the relevant information.

Another set of models are focused on the use of supervised learning techniques such as HMM, rule learning, and conditional random fields [(3), (12)]. These techniques are learning a language model or a set of rules from a set of hand-tagged training documents, and then apply this model or rules to new texts. Models learned in this manner are effective on documents which similar to the set of training documents, but the extraction became quite poor if it is applied to documents with a different genre or style. As a result, this approach has difficulty on scaling to the web due to the diversity of text styles and genres on the web and the prohibitive cost of creating an equally diverse set of hand tagged documents. Most of the technologies used in reducing the annotation burden in some annotation tools are based on supervised learning and they are requiring user-defined annotated corpora. When the task becomes complex and the documents to cope with present high variability in type increased (free texts are mixed with more or less rigidly structured pages), the amount of annotated material grows and supervised learning become unfeasible. To achieve the main contribution of the information extraction task with respect to web services offered by the websites which used to collect relevant information for a specific domain, to integrate it to be usable for the user and to make the technology scalable to a large number of cases, it is necessary to define a generic portable architecture in an easy way. In [(6)] architecture based on web services are proposed where each task is divided into subtasks. Each subtask is performed by a server which in turn uses other servers for implementing parts of the subtask. Each server exposes a declaration of an input, output and a set of working parameters. Servers are reusable in different contexts and applications. In this model the extracted information used to bootstrap more complex modules such as wrappers which will in turn collect more information used in training more sophisticated IE engines. Redundancy of information is very important in getting other pieces of information. When known information is found in different sources, then they can be used to bootstrap recognizers. The more the task become complex, the more information needed for training, the more reliable input data becomes difficult to identify. [(5)] Has proposed to compose existing web services with information extraction predefined operators in order to build new information extraction web services. This framework is composed of two phases. The first phase, identify the sub tasks of the information extraction task and define them as operators which some of them are generic such as querying, fetching or parsing, while the others are specific to the task. The second phase, an XML language specification is defined for the general purpose web service. This method still limited because building of the proposed information extraction web service is done manually according to the user needs by writing the web service XML based on the operators might needed. The Lixto tool [(4)] is based on the wrapper technology to extract the relevant information from HTML document and translate it into XML which can be easily queried and further processed. Once the wrapper is built, it can be applied automatically to continually extract relevant information from permanently changing web pages. Lexito toolkit consists of the following modules:

- Interactive pattern builder.
- Extractor
- Controller.
- Transformer.

Lixto wrapper is built interactively by creating patterns in a hierarchical order. The user interface is extremely simple and the entire wrapper construction process can be learned by an average

user in very short time. The user is guided through a supervised pattern generation and by simply marking relevant information items on-screen and visually setting constraints filters and patterns are created. Lixto offers two basic mechanisms of data extraction, tree and string extraction. Lixto tool characterized with a set of benefits that make this tool a popular one. Lixto is easy to learn and use because a fully visual and interactive user interface is provided. Lixto uses a straightforward region marking and selection procedures that allow all users even those not familiar with HTML to work with the wrapper generator. Lixto permits a wrapper designer to work directly and solely on a browser-displayed example pages. It is also allowing for extraction of target patterns based on surrounding landmarks on the contents itself on HTML attributes in order of appearance, on semantic and on syntactic concepts. Lixto also allows for more advanced features such as disjunctive pattern definitions, crawling on other pages during extraction and recursive wrapping and the extracted data structures do not have to strictly obey the input HTML structure.

KNOWITALL is another system that aims to automate the tedious process of extracting large collections of facts from the web in an autonomous, domain-independent, and scalable manner. This system can overcome the previous systems challenges from using the supervised learning technique and domain-dependent. KNOWITALL unlike wrapper induction system which operates only on highly structured documents or TREC systems which extract information from relatively small corpora of newswire and newspaper articles (3), (1) and (22). KNOWITALL is an autonomous system that extracts facts, concepts, and relationships from the web. KNOWITALL is seeded with an extensible ontology and a small number of generic rule templates from which it creates text extraction rules for each class and relation in its ontology. The system relies on a domain- and language-independent architecture to populate the ontology with specific facts and relations. KNOWITALL is designed to support scalability and high throughput. The KNOWITALL's consists of four main modules as illustrated in Figure 2:

- The Extractor.
- Search engine interface.
- Assessor
- Database.

In order to scale readily to a new classes, which requires minimizing the amount of hand-entered training data, this system rely on a bootstrapping technique that induces seeds from generic extraction patterns and automatically- generated discriminator phrases. The KNOWITALL can be characterized by:

Dealing with unstructured text such as web pages that have large scale of information.

It begins with a domain-independent set of generic extraction patterns which induces a set of seed instances.

It's using of Turney's PMI-IR methods to assess the probability of extractions using web-scale statistics.

It is relied on the scale and redundancy of the web for an ample supply of simple sentences that are relatively easy to process.

Domain independent and highly automated.

Employing unsupervised learning methods that extract facts by using search engines to home in on easy-to-understand sentences scattered throughout the Web.

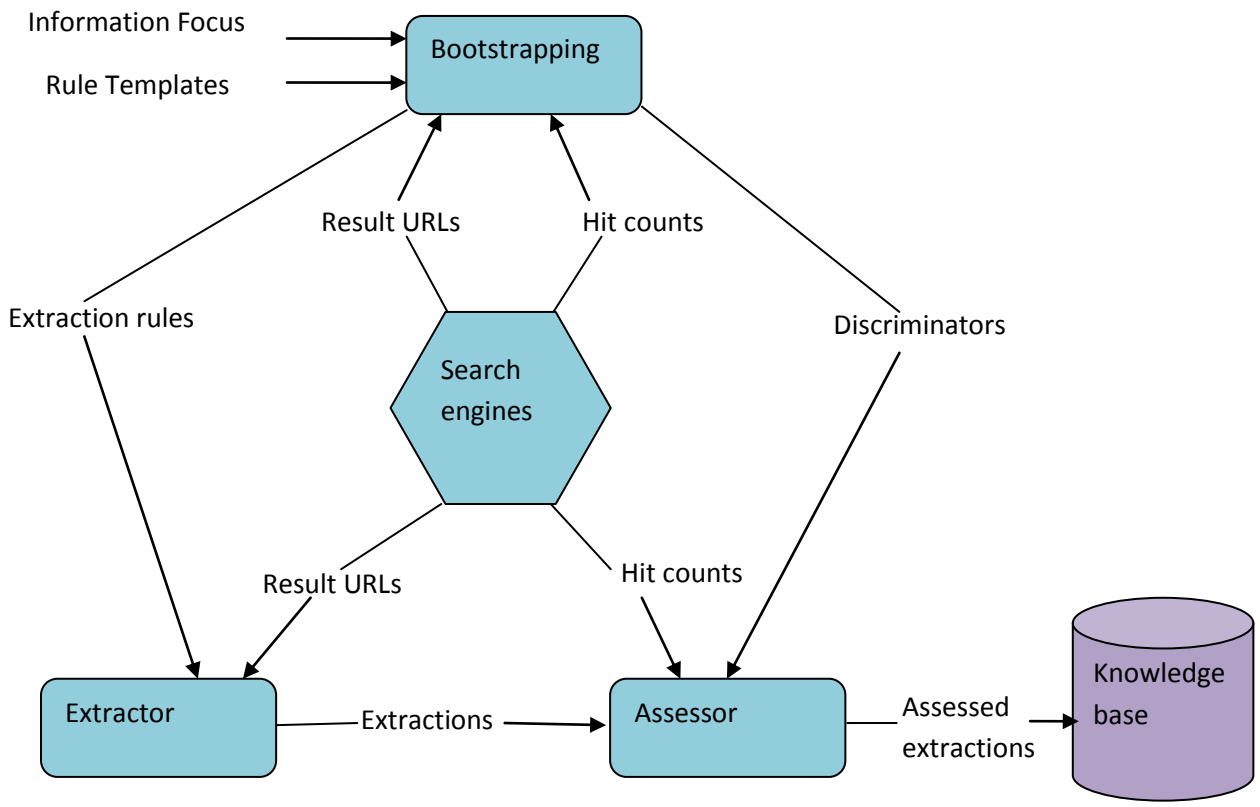

Figure 2 Flowchart of the main components in Know It All [22]

OPEN INFORMATION EXTRACTION

The traditional information extraction systems are focused on satisfying precise, narrow and pre-specified requests from small homogenous corpora. Applying the system on another domain requires the user to name the target relation and manually create new extraction rules or hand-tag new training examples. The efficiency problem with using the KNOWITALL system is it is time consuming, but it can be addressed by Open Information Extraction (OIE) because KNOWITALL requires large numbers of search engine queries and web page downloads. KNOWITALL takes relation names as an input. Thus, the extraction process has to be run and rerun each time a relation of interest is identified. OIE is a novel extraction paradigm that facilitates domain-independent discovery of relations extracted from text and readily scales to the diversity and size of the Web corpus [14], [15]. TEXTRUNNER is the first scalable, domain-independent OIE system. TEXTRUNNER is a fully implemented system that extracts relational tuples from text. The tuples have assigned probability and indexed to support efficient extraction and exploration via user queries [22]. TEXTRUNNER consists of three key modules and their interaction is illustrated in Figure 3:

Self-supervised learner which contains two phases, first phase is automatically labelling its own training data as positive or negative ones automatically. In second phase, the labelled data is used to train a Naive Bayes classifier which is used by the extractor module.

Single-pass extractor: the extractor makes a single pass over its corpus automatically by tagging each word in each sentence with its most probable part-of-speech.

Redundancy-based assessor: TEXTRUNNER automatically merges tuples which both entities and normalized relation are identical and counts the number of distinct sentences from which each extraction was found.

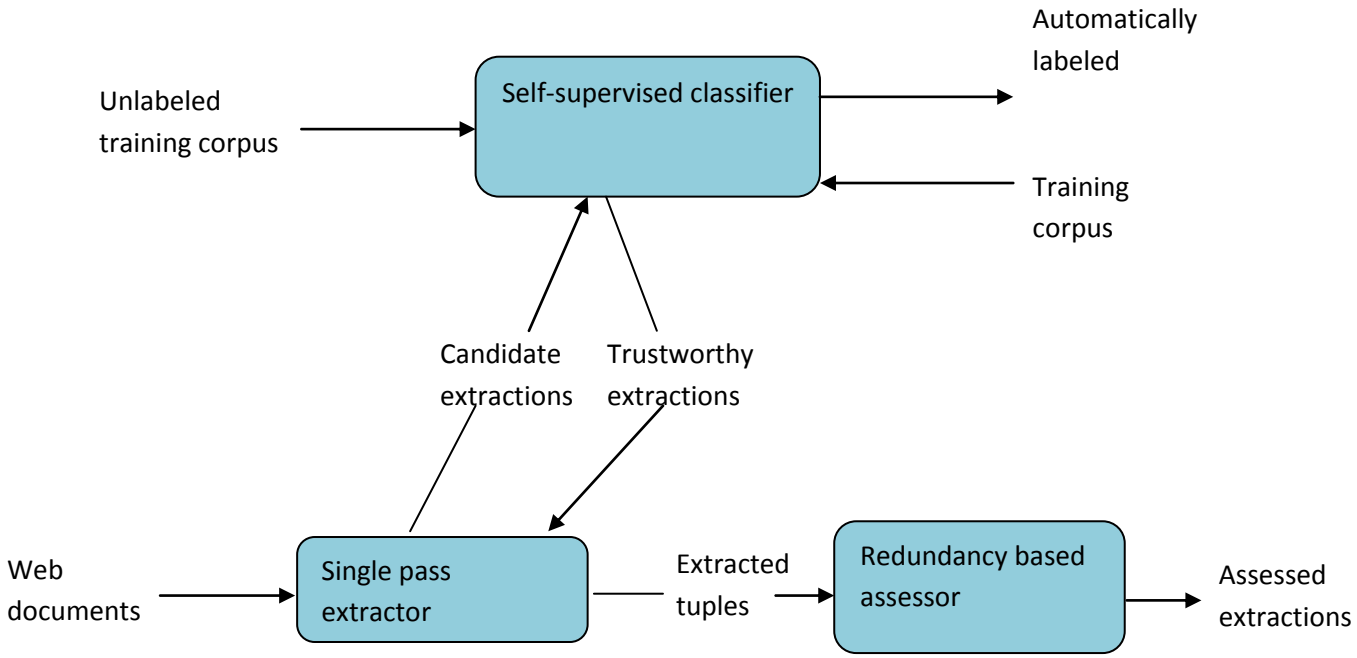

Figure 3: Flowchart of the main components in Text Runner [22]

TEXTRUNNER has a set of advantages compared with the other traditional IE systems by proving theoretically and practically that it is excel over the KNOWITALL in the efficiency, speed and scalability. Standard IE systems can only operate on relations given to it a priori by the user, and are only practical for a relatively small number of relations. In contrast, Open IE operates without knowing the relations a priori, and extracts information from all relations at once. TEXTRUNNER achieves an error reduction of 33% on a comparable set of extractions. OIE form the web is an unsupervised extraction paradigm that eschews relation-specific extraction in favor of a single extraction pass over the corpus during which relations of interest are automatically discovered and efficiently stored. To convey the quality and type of extractions that TextRunner extraction, the following analysis classifies TextRunner's extractions in four dimensions:

Correctness, whether the extraction has the same truth value as conveyed by the sentence it was extracted from.

Abstractness, whether the truth of the extraction is grounded in particular entities (a concrete extraction), or the extraction is general and underspecified (an abstract extraction).

Well-formed relation, relations are judged to be well-formed if there exists some pair of arguments for which the relation holds.

Well-formed arguments, arguments to a relation r are said to be well formed for r if they are of the correct type.

Open IE systems [15] make a single (or constant number of) passes over a corpus and extract a large number of relational tuples (Arg1, Pred, Arg2) without requiring any relation specific training data. For instance, given the sentence "McCain fought hard against Obama, but finally lost the election," an Open IE system should extract two tuples, (McCain, fought against, Obama), and (McCain, lost, the election). The strength of Open IE systems is in their efficient processing as well as ability to extract an unbounded number of relations. The second generation from Open IE systems was developed to overcome two important problems in previous Open IE such as TEXTRUNNER and WOE. The two significant problems are incoherent extractions and uninformative extractions [23].

Incoherent extractions are cases where the extracted relation phrase has no meaningful interpretation. Incoherent extractions arise because the learned extractor makes a sequence of decisions about whether to include each word in the relation phrase, often resulting in incomprehensible relation phrases. The second problem in uninformative extractions occurs when extractions omit critical information. For example, consider the sentence "Hamas claimed responsibility for the Gaza attack". Previous Open IE systems will return the uninformative: (Hamas, claimed, responsibility) instead of (Hamas, claimed responsibility for, the Gaza attack). Based on these two problems in previous Open IE the ReVerb extractor is introduced which implements a general model of verb-based relation phrases, expressed as two simple constraints (Syntactic Constraint and Lexical Constraint). ReVerb is a novel open extractor based on the two simple constraints. ReVerb first identifies relation phrases that satisfy the syntactic and lexical constraints and then finds a pair of NP arguments for each identified relation phrase. The resulting extractions are then assigned a confidence score using a logistic regression classifier trained on 1,000 random web sentences with shallow syntactic features. This algorithm differs in three important ways from previous methods. First, the relation phrase is identified holistically rather than word-by-word. Second, potential phrases are filtered based on statistics over a large corpus (the implementation of lexical constraint). Finally, ReVerb is relation first rather than arguments first, which enables it to avoid a common error made by previous methods—confusing a noun in the relation phrase for an argument like the noun responsibility in claimed responsibility for [15]. ClausIE is a novel clause-based approach to open information extraction which extracts relations and their arguments from natural language text. Virtually all of existing OIE techniques make use of handcrafted extraction heuristics or automatically constructed training data to learn extractors or estimate the confidence of propositions. Some approaches such as Text Runner, WOEpos, Reverb, and R2A2 focus on efficiency by restricting syntactic analysis to part-of-speech tagging and chunking. These fast extractors usually obtain high precision for high-confidence propositions, but the restriction to shallow syntactic analysis limits maximum recall and/or may lead to a significant drop of precision at higher points of recall. Other approaches such as Wanderlust, WOEparse, KrakeN, OLLIE, and additionally uses dependency parsing. These extractors are generally more expensive than the extractors above because they trade efficiency for improved precision and recall. Each of these approaches makes use of various heuristics to obtain propositions from the dependency parses (24). ClausIE also makes use of dependency parsing. ClausIE fundamentally differs from previous approaches in that it separates the detection of useful pieces of information expressed in a sentence from and its representation in terms of one or more propositions. The detection of clauses is based on the dependency parse and to detect clause types use a small set of domain-independent lexica. ClausIE separates the detection of clauses and clause types from the actual generation of propositions. This allows ClausIE to obtain more and higher-precision extractions than alternative methods, but also enables flexible generation of propositions. ClausIE can be seen as the first step towards clause-based open information extraction.

OTHER METHODS FOR INFORMATION EXTRACTION

Methods used in IE classified the information extraction task according to the information source type: free text, structured text, or semi structured text. The patterns of information are classified into static and non-static structures and use different techniques to extract the relevant information. This method can overcome the challenges that the web pages with semi-structured format have as illustrated before. This

can be done by transforming the page format into extensible hypertext mark-up language (XHTML). Then, make use of the DOM tree hierarchy of a web page and lexical dictionary and called this approach with advanced approach of DOM tree by enhancing the keyword set. The system consisting of two phases as in Figure 4, the pre-processing phase and extraction phase (1). In pre-processing phase all pages transformed into XHTML format to overcome the weak representation of HTML documents. The objective of the training process is to mine out patterns and rules for target information. In extraction phase which based on human training results, the system chooses suitable extraction methods for different information fields.

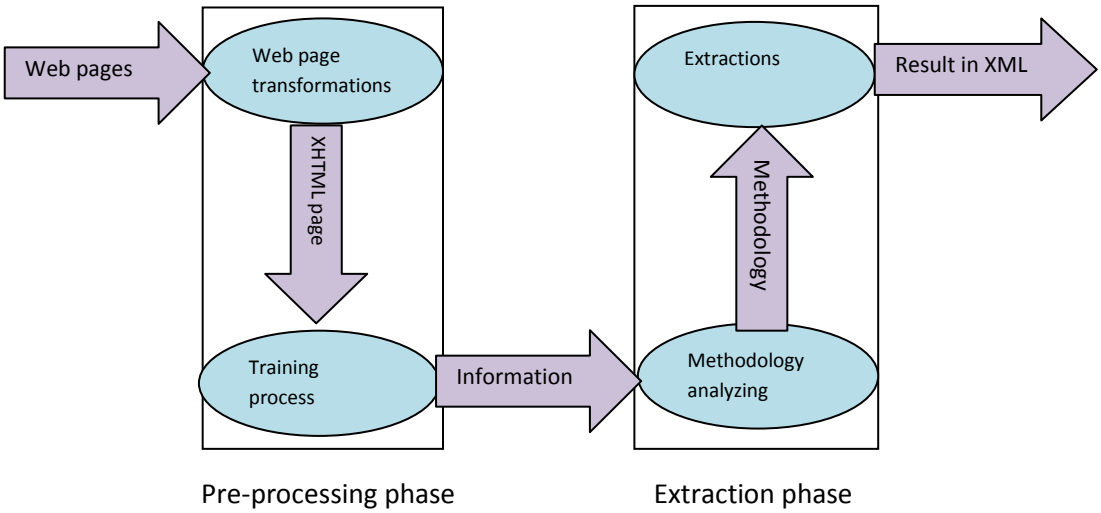

Figure 4: the System Architecture (1)

The advanced approach of using DOM tree is showing an improved in the results than using the basic approach by precision and recall as a metric evaluation method. The major advantages of this approach is that after enhancing the key word set a more direction of the information is able to be captured out at the same time. Since the added words are come from a lexical dictionary, only the synonymous with the same word type will be considered to be added which make more noise to the key word set. In addition, by using the advanced approach, it is possible to omit the training process. After the user enters a keyword, a set of synonymous can be found out from dictionary. By using those words, it is enough to extract some information. The limitation of this method is that the extraction task is only individual page based.

The Role of Information Extraction in Semantic Web Social networks play an important role in our daily lives. People conduct communications and share information through social relations with others such as friends, family, colleagues, collaborators, and business partners. Our lives are influenced by social networks without our knowledge of the implications. In the context of the semantic web, social networks are crucial to realize a web of trust, which enables the estimation of information credibility and trustworthiness because anyone can say anything on the web. For that reason, the web of trust helps humans and machines to discern, reliably, which contents are credible, and to determine which information is useful. Ontology construction is also related to a social network. For example, if numerous people share two concepts, the two concepts might be related [25]. Several means can be existed to demarcate social networks and one approach is to make a user describe relations to others. In the social sciences, network

questionnaire surveys are often performed to obtain social networks. Automatic detection of relations is also possible from various sources of information such as e-mail archives, schedule data, and web citation information. In some studies, social networks are extracted by measuring the co-occurrence of names on the web.

SOCIAL NETWORK EXTRACTION SYSTEMS

In the mid-1990s, Kautz et al. developed a social network extraction system from the web called Referral Web [25]. The system addresses co-occurrence of names on web pages using a search engine. It estimates the strength of relevance of two persons X and Y by putting a query "X and Y" to a search engine: If X and Y share a strong relation, we can find much evidence that might include their respective homepages, lists of co-authors in technical papers, citations of papers, and organizational charts. Interestingly, a path from a person to a person (from Henry Kautz to Marvin Minsky) is obtained automatically using the system. Later, with development of the WWW and semantic web technology, more information on our daily activities has become available online. Automatic extraction of social relations has much greater potential and demand now than when referral web is first developed. Several researchers have used that technique to extract social networks. Recently, Mika developed a system for extraction, aggregation and visualization of online social networks for a semantic web community called Flink [26]. Social networks are obtained using analyses of Web pages, e-mail messages, and publications and self-created profiles Friend-of-a-Friend (FOAF files). The Web mining component of Flink, similarly to that in Kautz's work, employs a co-occurrence analysis. Given a set of names as input, the component uses a search engine to obtain hit counts for individual names as well as the co-occurrence of those two names. The system targets the semantic web community. Therefore, the term "semantic web OR ontology" is added to the query for disambiguation. McCallum and co-workers [27], [28] present an end-to-end system that extracts a user's social network. That system identifies unique people in e-mail messages, finds their homepages, and fills the fields of a contact address book as well as the other person's name. Links are placed in the social network between the owner of the web page and persons discovered on that page. A newer version of the system targets co-occurrence information on the entire web, integrated with name disambiguation probability models. Other studies have used co-occurrence information. [29] Developed a system to extract names and also person-to-person relations from the web. [30] Obtained a social network of 15 million persons from 500 million web pages using their co-occurrence within a window of 10 words. A notable characteristic of such NLP studies compared to studies of social network mining is that they try to recognize the relation among general words, or classes, rather than a particular set of named-entities. They try to recognize synonyms, hyponyms–hyponyms, or coordinates (words with the same hyponyms). However, the difference is minor at least from algorithmic perspective; if we apply one method to more concrete entities. A network of named entities can be extracted if we apply the method to be more abstract concepts, we can extract a network of concepts [25]. A social network of participants is displayed in POLYPHONET to illustrate a community overview. Various types of retrieval are possible on the social networks. Researchers can be sought by name, affiliation, keyword, and research field, the related researchers to a retrieved researcher are listed and a search for the shortest path between two researchers can be made. Even more complicated retrievals are possible. For example, search for a researcher in the social network among researchers in a certain field. POLYPHONET is incorporated with a scheduling support system and a location information display system in the ubiquitous

computing environment at the conference sites [25]. Three kinds of social networks are addressed in POLYPHONET: user-registered knows network, web-mined collaborated network, and face-to-face meet social networks. Social networks are important for the semantic web. Integration of multiple networks, or spinning social networks, is becoming increasingly necessary [31]. Recent important approaches of a web mining toward the semantic web uses the web as a huge language corpus and combines with a search engine. The underlying concept of these methods is that it uses globally available web data and structures to annotate local resources semantically to bootstrap the Semantic Web. The previous surveyed approaches use a superficial approach instead of profound assessment to determine the type of relation. They adopt a supervised machine learning method which requires a large annotated corpus which costs great deal of time and effort to construct and administer. In addition, it is necessary to gather domain-specific knowledge a priori to define the extracted relations.[16] Proposed a method that automatically extracts the labels that describe relations among entities in social networks. They obtain a local context in which two entities co-occur on the web and accumulate the context of the entity pair in different Web pages. Given the collective contexts of each entity pair the key idea is clustering all entity pairs according to the similarity of their collective contexts. This method is entirely unsupervised and domain independent and it is easily incorporated into existing extraction methods of social networks.

CONCLUSION AND FUTURE WORK

The ultimate goal from IE is to extract the relevant information to the user automatically in a single file from different data sources with high precision and recall. Systems differ between them in the technique used, the learning strategy and the methodology applied. Some systems depend upon their specific extraction system. These systems don't care to the other domain; however, in search engine application the domain differs according to the user query. Thus, the need to design IE system deal with domain-independent is an interest research area where KNOWITALL, WOE, TEXTRUNNER, and OIE systems are developed. OIE systems provided more performance than previous systems. Clause-based approach the ClausIE obtains higher recall and higher precision than existing approaches, both on high-quality text as well as on noisy text as found in the web. Recent important approaches of web mining toward the semantic web are using the web as a huge language corpus and combine with a search engine. The underlying concept of the methods used to extract social networks is that it uses globally available web data and structures to annotate local resources semantically to bootstrap the semantic web. In the future, a plan to collect more information about other systems in this area and conduct a comparison among them practically is suggested. And it is planned to survey more large area in the topic of semantic web and information extraction.

REFERENCES

1. Lam MI, Gong Z, Muyeba M. A method for web information extraction. Progress in WWW Research and Development: Springer; 2008. p. 383-94.
2. Kaiser K, Miksch S. Information extraction-a survey. 2005.
3. Etzioni O, Cafarella M, Downey D, Kok S, Popescu A-M, Shaked T, et al., editors. Web-scale information extraction in knowitall:(preliminary results). Proceedings of the 13th international conference on World Wide Web; 2004: ACM.
4. Baumgartner R, Flesca S, Gottlob G, editors. Visual web information extraction with lixto. VLDB; 2001.
5. Habegger B, Quafafou M, editors. Web services for information extraction from

the Web. Web Services, 2004 Proceedings IEEE International Conference on; 2004: IEEE.
6. Ciravegna F, Dingli A, Guthrie D, Wilks Y, editors. Integrating Information to Bootstrap Information Extraction from Web Sites. IIWeb; 2003.
7. Simoes G, Galhardas H, Coheur L, editors. Information Extraction tasks: a survey. Proc of INForum; 2009.
8. Kehler A, Hobbs JR, Appelt D, Bear J, Caywood M, Israel D, et al., editors. Information extraction research and applications: current progress and future directions. Proceedings of a workshop on held at Baltimore, Maryland: October 13-15, 1998; 1998: Association for Computational Linguistics.
9. Tang H, Ye J. A Survey for Information Extraction Method.
10. Jiang J. Information extraction from text. Mining text data: Springer; 2012. p. 11-41.
11. Piskorski J, Yangarber R. Information extraction: Past, present and future. Multi-source, Multilingual Information Extraction and Summarization: Springer; 2013. p. 23-49.
12. Wiebe J, Riloff E. Finding mutual benefit between subjectivity analysis and information extraction. Affective Computing, IEEE Transactions on. 2011;2(4):175-91.
13. Gatterbauer W, Bohunsky P, Herzog M, Krüpl B, Pollak B, editors. Towards domain-independent information extraction from web tables. Proceedings of the 16th international conference on World Wide Web; 2007: ACM.
14. Banko M, Cafarella MJ, Soderland S, Broadhead M, Etzioni O, editors. Open information extraction for the web. IJCAI; 2007.
15. Etzioni O, Fader A, Christensen J, Soderland S, Mausam M, editors. Open information extraction: The second generation. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume One; 2011: AAAI Press.
16. Mori J, Ishizuka M, Matsuo Y, editors. Extracting Keyphrases to Represent Relations in Social Networks from Web. IJCAI; 2007.
17. Sarawagi S. Information extraction. Foundations and trends in databases. 2008;1(3):261-377.
18. Hobbs JR, Riloff E. Information extraction. Handbook of natural language processing. 2010;2.
19. Muslea I, editor Extraction patterns for information extraction tasks: A survey. The AAAI-99 Workshop on Machine Learning for Information Extraction; 1999.
20. Chang CH, Kayed M, Girgis MR, Shaalan KF. A survey of web information extraction systems. Knowledge and Data Engineering, IEEE Transactions on. 2006;18(10):1411-28.
21. Chiticariu L, Li Y, Raghavan S, Reiss FR, editors. Enterprise information extraction: recent developments and open challenges. Proceedings of the 2010 ACM SIGMOD International Conference on Management of data; 2010: ACM.
22. Yates A. Information extraction from the web: Techniques and applications: University of Washington; 2007.
23. Velardi P, D'Antonio F, Cucchiarelli A, editors. Open domain knowledge extraction: inference on a web scale. Proceedings of the 3rd International Conference on Web Intelligence, Mining and Semantics; 2013: ACM.
24. Del Corro L, Gemulla R, editors. ClausIE: clause-based open information extraction. Proceedings of the 22nd international conference on World Wide Web; 2013: International World Wide Web

Conferences Steering Committee.

25. Matsuo Y, Mori J, Hamasaki M, Nishimura T, Takeda H, Hasida K, et al. POLYPHONET: an advanced social network extraction system from the web. Web Semantics: Science, Services and Agents on the World Wide Web. 2007;5(4):262-78.
26. Mika P. Flink: Semantic web technology for the extraction and analysis of social networks. Web Semantics: Science, Services and Agents on the World Wide Web. 2005;3(2):211-23.
27. Culotta A, Bekkerman R, McCallum A. Extracting social networks and contact information from email and the web. 2004.
28. Bekkerman R, McCallum A, editors. Disambiguating web appearances of people in a social network. Proceedings of the 14th international conference on World Wide Web; 2005: ACM.
29. Harada M, Sato S-y, Kazama K, editors. Finding authoritative people from the web. Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries; 2004: ACM.
30. Faloutsos C, McCurley KS, Tomkins A, editors. Fast discovery of connection subgraphs. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining; 2004: ACM.
31. Matsuo Y, Hamasaki M, Nakamura Y, Nishimura T, Hasida K, Takeda H, et al., editors. Spinning multiple social networks for semantic web. Proceedings of the National Conference on Artificial Intelligence; 2006: Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.