Journal of International Technology and Information Management

Volume 19 | Issue 4 Article 1

2010

Use of a Fast Information Extraction Method as a Decision Support Tool

Mahmudul Sheikh
Rust College

Sumali Conlon
University of Mississippi

Use of a Fast Information Extraction Method as a Decision Support Tool

Mahmudul Sheikh
Rust College
USA

Sumali Conlon
University of Mississippi
USA

ABSTRACT

Ad-hoc extraction of information from documents can ensure the transparency of decisions made by an organization. Different Information Extraction methods have been applied to extract information from various domains. Most widely known methods use manually annotated training documents that require high development time. The automated training methods are not scalable to large application domains. We have developed a semi-automated knowledge-engineering method for building the knowledge-base with minimal efforts. Because our method reduces manual processing of the training data, the development process is very fast. We have developed a prototype application to extract information from the project-reports of the American Recovery and Reinvestment Act (ARRA) of 2009. The fast development process of our system, its scalability to large application domains, and its high extraction effectiveness will help the transparency of management decisions by extracting and mining relevant information.

INTRODUCTION

Most electronic documents are available in the free-text format. Analyzability of the free-text information can play a crucial role in decision making through pattern discovery (Lo & Hsieh, 2003). The enormous volume of available documents can be processed semi-automatically by a closed or an open system (Banko & Etzioni, 2008). Important applications such as electronic governance (Zhang, Lin, Lin, & Hsieh, 2008) may entail the use of extracted information from these documents to gain competitiveness and public trust by enforcing transparency. An effectively designed Information Extraction (IE) system can be useful in this regard. An IE system is designed to extract the user-specified items or pre-defined events (Srihari, Li, Niu, & Cornell, 2008) from the text documents of a specific domain.

The items to be extracted are specific word(s) of sentences of the input documents. An IE system can be used to fill the fields of a table of a relational database from text documents. Examples of the fields may include the name of a company or type of business. In order to interpret the extracted information, these items can be saved into a relational database. The extracted items can also be used to fill-out forms specified by a user. The information saved in the database can be further processed to identify the relevant correlations.

Compared to the extraction of structured data performed by, for example, an ERP system (Wu, Hsieh, Shin, & Wu, 2005), extracting information from a free-text domain is a challenging

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

problem. The reason is that only the contextually relevant target words or phrases have to be extracted. Because statistics does not consider the contexts of the words, the methods such as Naive Bayes Classifier or Average Mutual Information (Carven et al., 2000) are not effective in extracting information. Most of the state of the art IE systems use a combination of statistical and machine learning methods (Freitag, 2000). José Iria and Fabio Ciravegna (2006) developed an ontology learning (Suchanek, Sozio, & Weikum, 2009) and document classification method that represents language resources, such as syntactic and semantic parsers, in a way that is independent of the extraction process. Yildiz and Miksch (2007) proposed an unsupervised rule learning method (Downey, Schoenmackers, & Etzioni, 2007; Sekine & Oda, 2007) based on ontological structures. Syntactic and semantic analyses have been found to improve precision (Feldman, Aumann, Finkelstein-Landau, Hurvitz, Regev, & Yaroshevich, 2002). Xu, Uszkoreit, Li, and Felger (2008) designed a system that extracts linguistic grammar rules from a semantic seed. An alignment based pattern matching technique was developed by (Kim, Jeong, Lee, Ko, & Lee, 2008) that can extract relationships between two arguments. Jain, Ipeirotis, and Gravano (2008) developed a technique to process structured queries from the relations extracted from unstructured texts. Stevenson and Greenwood (2009) found that the IE models that use the relevant portions of a dependency tree (Wu & Weld, 2008) perform better. A rule-based decision tree was found effective in extracting information from online resumes (Bhargavi, Jyothi, Jyothi, & Sekar, 2008). The use of deep level ontology structure (Welty & Murdock, 2006) and the use of multiple ontologies from the same domain (Wimalasuriya & Dou, 2009) have been found effective in performance enhancement. Assigning weights to the syntactic features according to their co-occurrences in the related class has been found effective in determining ontologies about person-names and their geographic locations (Tanev & Magnini, 2008).

Enterprise applications of IE systems require higher accuracy and scalability (Chiticariu, Li, Raghavan, & Reiss, 2010). Similar to many other systems, Pennacchiotti and Pantel (2009) used a combination of a pattern match method with a distributional method. However, these systems do not address the challenge of extracting irregular target information. The irregularities usually arise from unpredictable exact words, irregular sequences of parts of speeches, and in some cases non-contiguity of the target words in a sentence. Automated training methods require manually annotated training documents and the annotation consumes huge human labors. The enormous training times required by the automated training methods make them inapplicable to large text domains. This study addresses these issues by investigating the effectiveness of a semi-automated (Milosavljevic, Grover, & Corti, 2007) knowledge-engineering method. The prototype system developed for this study has been applied to the extraction of information from the reports of the projects funded by the American Recovery and Reinvestment Act (ARRA) of 2009.

The ARRA of 2009 approved more than 40,000 projects to various public and private organizations. The first progress reports on these projects were collected from the website (www.recovery.gov). The reports contain tremendously irregular linguistic patterns. Another challenge is that, in some cases the target information is scattered across a sentence. In order for resolving these issues, we investigated the effectiveness of a semi-automated IE method.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

The usefulness of an IE system depends on the quality of information extracted by the system and its scalability to large text domains. In order to address these issues, this study attempts to answer the following questions:

1) How effectively can we use a knowledge-engineering approach to extract information from a domain of enormous pattern variety?

2) How successfully can a knowledge-engineering method be applied to the extraction of non-contiguous information from sentences?

3) Can a knowledge-engineering method be made scalable to a large text domain?

The IE system used in this study is based on a semi-automated knowledge-engineering method. The method exploits the synergy of a knowledge-base, a database, and an inference engine. The knowledge-base has been manually constructed to cover most of the relevant patterns. The database contains relevant target words and other surrounding words of the relevant sentences. The database was built on a sub-set of the documents of the data corpus. The inference engine comprises of a list of embedded SQL statements and an application module developed by Perl.

BUSINESS APPLICATIONS OF IE SYSTEMS

Effective Information Extraction (IE) systems can have very important business applications. An IE system can be used as an embedded module of an enterprise system. It can also be used as a supporting system for Semantic Web and Web Services (Lee et al., 2003) infrastructure. Government agencies are using IE systems as a tool for intelligence gathering. An IE system may also be used to improve the document retrieval performance of search engines (e.g. Google, AltaVista).

If information has to be interpreted by a computer, it should to be presented in Semantic Web (Hendler et al., 2002) format. In future, the Semantic Web will be able to express the relationships among the web entities (Berners-Lee et al., 2001). Semantic Web representation can be automated by extracting the semantically related information. The future IE systems may also benefit from using the entity-relationships (Kalyanpur et al., 2004, Albanese & Subrahmanian, 2007) and metadata provided by the Semantic Web to produce more accurate results (Iria & Ciravegna, 2005).

Different application domains of the Web Services include concept retrieval (Fu & Mostafa, 2004), e-commerce applications (Habegger & Quafafou, 2004), and Semantic Web applications (McIlraith et al., 2001). An IE module can be used to extract finer grained and context-relevant information by using ontology (Bratus et al., 2009).

Financial documents can be analyzed to extract important financial information (Grant & Conlon, 2006). Finding correlations among the extracted entities is termed as knowledge mining (Mooney, 2005). IE systems have been used for extracting enterprise contents from unstructured documents and for finding correlations among the extracted entities (Fensel, 2001).

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

In order to analyze a market, an IE system can be used to extract information about demands, designs, and prices of products by using ontology-based instance composition (Labsky et al., 2005). An IE system can be embedded into an intelligent web agent (Eliassi-Red & Shavlik, 2003) to perform customer profiling or business-to-business relationship management (Fensel, 2001). An IE system can also be used for the extraction of customer-opinions about products, for product classification (Dini & Mazzini, 2002) and for the extraction of product attributes (Ghani, et al., 2006). Other applications include legal case analysis (Chieze1 et al., 2010), extraction of crime information (Ku et al., 2008), and improvement of the contents of social media (Hoffmann et al., 2009).

IE systems are being used as a complementary technology to other natural language processing tasks such as text classification, ontology generation, knowledge representation (Sheth, 2005), and metadata extraction (Heidorn & Wei, 2008). A data mining method can ascertain uncertainty information from the output of an IE system (McCallum, & Jensen, 2003). The precision of text classification (Riloff & Lehnert, 1993) or text categorization (Betts et al., 2007) can be improved by using the output of an IE system. An IE system can be used to provide a virtual web service for the websites that do not provide any web service API (Han & Tokuda, 2008). Some of the biomedical applications include information retrieval from MedLine database (Uramoto et al., 2004), annotation of protein and gene names (Leitner & Valencia, 2008), medical text classification (Sotelsek-Margalef & Villena-Román, 2008), and classification of medical terms (Hsiao et al., 2009).

RELATED SYSTEMS

Both of the knowledge-engineering and automated training methods have been used for extracting information from various free-text domains. In early 1990s, JASPER (Journalist's Assistant for Preparing Earnings Report) system used syntactic and semantic knowledge to extract corporate earnings information. POETIC (Portable Extendable Traffic Incident Collator) was developed to extract traffic information by using a three-tier lexicon (Cahill, 1994). FACILE (Fast & Accurate Categorization of Information by Language Engineering) was a value-added application system (Chiravegna, 1999) that extracted financial information from a news domain. SCISOR (Information Summarization, Organization & Retrieval System) was designed to extract merger and acquisition information from news stories (Jacobs, 1990).

Considering important implications in the military, public administration, and business arenas, development of some of the IE applications was initiated by the Message Understanding Conferences (MUC). MUC had been sponsored by Defense Advanced Research Project Agency (DARPA). MUC-1(1987) and MUC-2(1989) focused on the development of IE systems for extracting information about naval operations. MUC-3(1991) and MUC-4(1992) were arranged to extract information from a terrorist domain. MUC-5 (1993) and MUC-6 (1995) were designed to extract information about new microelectronics products, joint venture, and management success. MUC-7(1997) focused on extracting information about air-shuttle launches from a similar corpus as that of MUC-6.

AutoSlog (Riloff, 1993) was used to build a dictionary for the MUC-4 domain of terrorist events. The system relies on CIRCUS (Lehnert, 1991) that performs domain-specific conceptual analysis

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

on input sentences. The LIEP system (Huffman, 1995) was used to extract management change events by using hand-built patterns. CRYSTAL (Soderland etal., 1995) was applied to extract disease symptom information from hospital reports by using grammatical patterns. HASTEN (Krupka, 1995) was built on a MUC-6 domain to extract management succession events by utilizing user-annotated examples.

PALKA (Parallel Automatic Linguistic Knowledge Acquisition) (Kim and Moldovan, 1995) used automatically created linguistic patterns to extract information from web documents. MITA (Glasgow et al., 1997) was developed to extract significant concepts for medical and occupational text fields from a domain of MetLife Insurance by using a knowledge-engineering method. The RAPIER (Robust Automated Production of Information Extraction Rules) system (Califf & Mooney, 1997) used Inductive Logic Programming (ILP) technique (Mooney, 1996) to extract information about computer job postings. Guarino (1997) introduced the concepts of semantic matching in information retrieval and extraction.

SRV (Freitag, 1998) utilized a relational rule-learning technique to extract course and seminar announcement information from university web pages and email messages respectively. WHISK (Soderland, 1999) used syntactic constituents and semantic labels to extract information from newswire stories. Maedche and Staab (2000) used manual and semi-automatic ontology engineering to extract non-taxonomic conceptual relations from a telecommunication domain. VargasVera et al., (2001) combined the process of ontology-based markup, rule learning, and information extraction component to extract ontological information from a news domain. Kosseim, Beauregard, and Lapalme (2001) combined IE with natural language generation technique for responding to email messages.

MUSE (Maynard et al., 2001) is a component of GATE (Cunningham et al., 2002) adapted to extract the named entities. Nemati et al., (2002) proposed a knowledge warehouse architecture that uses extracted information and knowledge documented by the knowledge workers. Ciravegna and Wilks (2003) developed an adaptive IE system to perform document annotation in the semantic web framework. Dingli, Ciravegna, and Wilks, (2003) combined information extraction, information integration, and machine learning techniques to automatically annotate domain-specific information. Manov et al., (2003) used publicly available location resources and combined the location information to extract domain-dependent named entities.

Rosendfeld et al., (2006) developed the TEG (Trainable Extraction Grammar) system that combines knowledge-based and statistical methods to extract entities and their relationships. Sekine (2006) combined pattern discovery, paraphrase discovery, and named entity recognition methods for the user-query based extraction of financial information. Hakkani-T¨ur, and Tur (2007) developed a statistical sentence extractor for information distillation by using discriminative classification method. XAR (Ashish et al., 2009) allows users to specify Datalog-style abstract expressive rules to extract information by enforcing semantic integrity constraints. A tree structured extraction system that uses conditional random fields was developed (Piao et al., 2010) to extract high level structured information in XML format.

THE DATA CORPUS

Our Semi-Automatic methods have been applied to extract project information from the first reports of about 40,000 projects funded by the ARRA of 2009. The recovery act was passed in 2009 to stimulate the economy by creating and saving jobs. The total amount of the bill was $787 billion. Due to the huge monetary size of the bill, the current administration felt that it would be better to let the general public know about the details of the projects funded by the Recovery Act. For that purpose of ensuring transparency, the funding data was uploaded to the government website (http://www.recovery.gov). The initial version of the corpus contains all of the project-reports in a single XML Document. Because the single XML document contains all of the first reports, it becomes clumsy to load and read he file in a personal computer.

Even though the XML format makes it easier for a computer program to extract a certain types of information such as project name and duration, a huge variety of other information such as project purpose, project type, and the types of jobs created is written in plain text format. A key purpose of this study was to develop a semi-automated information extraction model that can be used to extract a certain set of user-defined information from the XML document. Figure 1 shows a sample of the reports available in the initial version of the XML file.

Figure 1: A sample report from reports of the project funded by the Recovery Act 2009.

<award_type>G</award_type>
<award_date>2009-03-02T00:00:00-05:00</award_date>
<award_description>Highway Infrastructure Investment Grant: Available for Use in Any Area (flexible)</award_description>
<project_description>This project is a road widening in Richmond County. The project consists of the widening and reconstruction of Alexander Drive from Washington Road (SR 28) to Riverwatch Parkway (SR 104) in the City of Augusta. The existing two lane rural roadway will be widened to four lanes with a 20' raised concrete median, curb and gutter and sidewalks. The total project length is 0.87 miles.</project_description>

The XML document of the reports for the projects funded by the economic Recovery Act of 2009 was downloaded from the http://www.recovery.gov as a single document. The size of the XML document is 556 Mega Bytes.

Preparing the Corpus

The Sentence Extractor module of our system separates the XML tags from the initial file because; our purpose is to extract information that is not specified by any XML tag. The document contains various topics and only several of those topics might be of interest of any user. The parts of the document that are irrelevant in terms of the topics are excluded from the document by an automated sentence extractor module. Figure 2 shows a sample output of the sentence extractor module.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Figure 2: A sample output of Sentence Extractor module.

award_number: 08ST0401760
project_description:
Project 1 - The tribe will complete new landscaping and modernization of exteriors for 609 rental units in Districts
Project 2 - The tribe will complete major renovations to 13 of its vacant 1937 Housing Act rental stock in Districts 3 and 5 to bring them up to adopted codes and standards.
Project 3 - The tribe will extend infrastructure (electric, water, etc.)
to serve a minimum of 5 new homeownership units in District The units will be occupied by low-income Native American families and operated similarly to the 1937 Housing Act Mutual Help units including an MHOA.
job_creation:We have been able to employ up to 4 people to work on the fencing and other housing projects.

The Sentence Extractor module of our system extracts the sentences from the XML document by using the sentence delimiter (usually '.') and non-delimiter honorifics. The Sentence Extractor can extract any sentence even if it is contained in several lines. All of the sentences extracted from an original document were saved in 50 new text documents.

Considering the enormous size of the initial XML document, the Sentence Extractor module has been used to create 50 text files. We have randomly selected 40 of these text files to be used as the training files. These training files were used to build and refine the knowledge-base of our system. The remaining 10 files were used to test the performance of our system.

TOOLS AND SYSTEM ARCHITECTURE

We have designed, developed, and applied a knowledge-engineering method for the experimentation of this study. We have designed a knowledge-engineering framework that combines multiple tools and methods. Our knowledge-engineering framework includes a semi-atomically built knowledge-base, a domain-dependent database, an automated inference-engine, and a user input component.

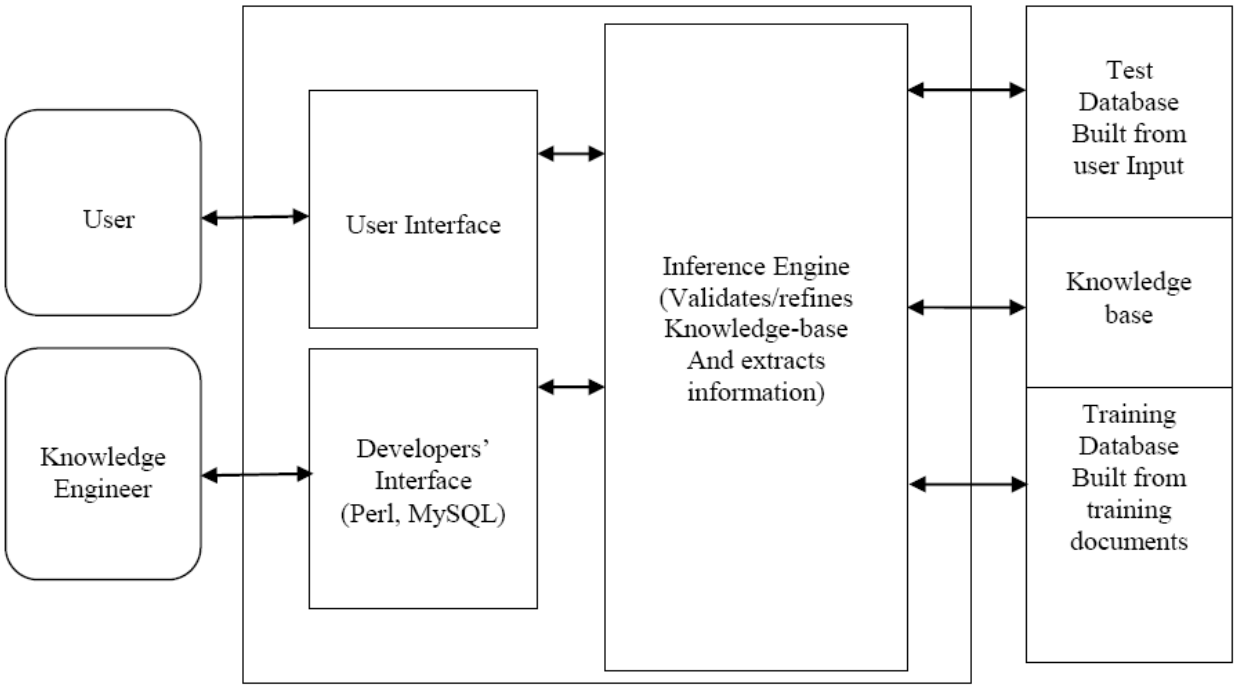
The knowledge-base comprises of a set of semi-atomically constructed rules. The rules developed for the purpose of our study are domain-dependent. The rules have been specifically tailored to fit the underlying corpus because application-specific rules tend to perform better (Negnevitsky, 2004). The database is built from the underlying corpus to validate the rules in the knowledge-base. The inference engine combines the rules from the knowledge-base and the facts from the database for producing its output. The user input takes the input documents from the users and feeds the document into the database system. The facts found in the database built from the user input are used to extract relevant information.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

The knowledge-base and the inference-engine modules have been developed by using Perl (Practical Extraction and Report Language). Perl combines features from high-level programming languages such as C or Basic. It contains several UNIX tools that make it flexible and adaptable to many other tools. A collection of Perl modules are also available for download at CPAN (Comprehensive Perl Archive Network). The database has been constructed by using MySQL that can be installed and run as a database server. It can be used by many users and it can handle large-scale databases. Perl includes a library of APIs that allows it to access data from a relational database management system such as MySQL.

Most of the modules of the system are fully automated or semi-automated. Building, validating, and refining the knowledge-base requires human intervention. The inference engine is fully automated. It uses the knowledge-base that was refined by using an iterative process until a satisfactory level of performance was achieved on the training data. The final extraction task is performed on the documents provided as input by the users. A separate test database is built from these user-provided documents to perform the extraction task. The inference engine directly uses this database to decide which information to extract. The decision reached by the inference engine depends on the knowledge-base and the database built from the user-provided input documents. Figure-3 shows the architecture of the system we have developed for this application.

Figure 3: The architecture of our knowledge-based information extraction system.



The following sub-sections describe each key component of the system architecture shown in figure 3.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Building database from training documents

Similar to the use of embedded extraction predicates in Datalog (Shen et al., 2007) we used a list of Perl modules with embedded SQL to feed the sentences of the training documents into the database. The purpose of building this training database was to use the key-word-in-context (KWIC) structure for searching the target words and their surrounding words. Figure 4 shows the KWIC example of a part of a sentence.

Figure 4: KWIC Entries for the sentence "All ARRA expenditures were to provide case services."

All	ARRA	expenditures	were	to	provide	case	services
ARRA	expenditures	were	to	provide	case	services	
expenditures	were	to	provide	case	services		
were	to	provide	case	services			
to	provide	case	services				
provide	case	services					
case	services						
services							

Searching and modifying the database by using SQL commands is much faster and easier than the speed of any programming language. For extracting information, a similar test database of words was also built from the input documents. The database contains the KWIC information for each sentence of the text documents.

Our system runs the KWIC index builder on every document of the data corpus. We have created 50 text files from the 40,000 projects funded by the ARRA of 2009. Each document contains the first reports of about 800 projects. Each table of the database contains the sentences from each of the 50 text files. Each of these files contains around 10,000 sentences. On an average, each table contains about n***10,000** records. Each of these records corresponds to any of these sentences where n is the average number of words in a sentence. The system sorts these records on various columns to learn about the structure of the sentences. Some sample entries from the reports, containing the word "created" in column 1, are shown in Table 1.

Table 1: Sample Entries in the KWIC Index File Containing the Term "created" in Column 1.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

W1	W2	W3	W4	W5	W6	W7	W8
created	a	part-time	Lab	Technician	and	a	part-time
created	a	Post-Doc	Scholar	position.			
created	a	new	full-time	position	at	the	Palo
created	a	Graduate	Student	Researcher	position.		
created	a	model	for	faith	based	org	to
created	an	entirely	new	FTE	job	position	in
created	an	on-line	application	form	for	our	program.
created	or	retained	are	service	coordination	and/or	child

Table 1 shows parts of the KWIC-indexed sentences that are sorted based on column 1. The table shows only the first 8 words of the rows of the KWIC index file.

Building the knowledge-base

The knowledge-base of our system is a set of semi-automatically constructed and refined rules. The extraction rules are built by analyzing the training documents. Our rule construction process is also helped by searching the training database by SQL commands. Only the records of the database where the contextual appearance of the target words and their synonyms (e.g., created, generated) found are considered for building the rules. The structure of our extraction rules are in the form of if-then-else statements. The following sentences are from the training documents. We used this kind of relevant sentences to construct the rules of our knowledge-base for each particular extraction slot.

> Sentences to construct the rules for the "Position Category" slot
> {Created} [two positions] {for} Graduate Research Assistants, one at 20 hours per week and one at 10 hours per week.
>
> This ARRA Supplement has allowed the [full-time] {employment of} [two] postdoctoral research associates.
>
> [Two full time] Family Service Workers have been {hired}.

The bracketed words of the above sentences represent the target values for the slot "Position Category." The braced words represent the surrounding words of the target values that may determine the contexts of the target values. Following are three initial rules created from the above three training sentences.

> Rules created for the knowledge-base of the "Position Category" slot
>
> Rule 1: IF w1=" Created" and (w4="for" or w4="to") and (w3="positions or w3=positions)
> Then output= w2 and w3
>
> Rule 2: If (w1=" full-time" or w1=" part-time") and w2=" employment" and w3="of"
> Then output= w4, and w1

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Rule 3: If (w2=”full’ or w2=”half”) and w3=”time” and (w4 or w5 or w6=”hired”)
Then output= w1, w2, and w3

The rules of the above structure are generalized by investigating the evidences from similar training sentences. A rule is generalized to a certain state where it covers a significant number of positive examples without covering a large number of negative examples.

Refining the knowledge-base

The knowledge-base of our system contains a set of rules for each extraction slot. Refining the knowledge-base is an iterative process that goes on until the performance level on the training data is satisfactory. The performance of a rule is defined in terms of the weighted average of the precision and recall obtained by the rule. The initially built rules are iteratively checked to determine the level of performance on the training documents. An excessively general rule usually covers many positive examples but the rule also covers a lot of negative examples. On the other hand, an excessively specific rule usually covers a few negative examples or no negative example at all but it does not cover a significant number of positive examples. We followed a specific-to-general (bottom-up) process of rule generalization to refine our knowledge-base. The following Rule 1 for the “Position Category” slot can be generalized to one step further by adding another option of value to the first word (w1).

Rule 1: IF w1=” Created” and (w4=”for” or w4=”to”) and (w3=”positions or
w3=positions)
Then output= w2 and w3

Refined Rule 1: IF (w1=” Created” or w1=” retained”) and (w4=”for” or w4=”to”) and
(w3=”positions or w3=positions)
Then output= w2 and w3

Because the expected slot values of Rule 1 in this case are the category (created or retained) and number of jobs, the above refinement of Rule 1 can extract more positive values for this slot. The refinement process for a rule is continued until a refined rule extracts more positive values for its underlying slot without extracting significantly large number of negative values.

Building database from the test document

In order to convert the user documents into the database records, our system uses the same list of SQL statements that was used to convert the training documents into database records. The purpose of building this test database was to use the KWIC structure for extracting the target words that the users need. Searching through the test database by using the SQL commands is very fast compared to the sequential search performed by any procedural or object-oriented program module.

Development of the Inference Engine

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

We have developed our inference engine by using the programming language Perl. The inference engine uses the knowledge-base and the test database to make decision about extracting relevant information. The knowledge-base used by the inference engine contains the final version of the extraction rules that were refined by using the training database. A separate knowledge-base is created for each individual extraction slot that is specified by the user. The test database used by the inference engine is built from the documents provided to the system by the user. The extraction rules try to find a match between the extraction rules and the database records built from the user documents. The matching in this regard means satisfaction of the antecedent conditions of a rule by the surrounding words of the target word(s) that the user wants to extract. The knowledge-base for each extraction slot is used to extract the target information for that slot from each input document from the user. Figure 5 shows how all target information related to an extraction slot is extracted by the inference engine from an input document provided by the user.

Figure 5: Information Extraction process used by the inference engine.

- For each rule R in the knowledge-base for slot #1
  - For each record r in the test database for input document #1
    - If record r satisfies all antecedent conditions of rule R
    - Then extract the target words from r specified by rule R
  - End For
- End For

The extraction process followed by the inference engine for each extraction slot from each input document is similar to the process shown in Figure 5. A separate program module that embeds the above extraction procedure was developed for each extraction slot.

RESULTS AND DISCUSSION

The effectiveness of our system has been tested on 10 randomly selected test documents that were created by the Sentence Extractor module from the original XML file. Each slot-value has been extracted by a separate knowledge-base generalized by using the 40 training documents. In order to extract more precise information regarding each of the slots, our system was designed to extract information related to several meaningful subfields related to each slot. The ability of our system to extract very fine-grained information demonstrates the effectiveness of our system. Table 2 shows the examples of the fine-grained output that were produced by our system from the test files.

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Table 2: A sample output for the extraction template.

Slot/Entity	Sub Fields and their extracted values
Project Purpose:	Project Purpose: Intensive/required professional development in curriculum Study Purpose: to evaluate a novel recombinant Lactobacillus vaccine against HIV Initial Purpose: (1) research activities and (2) public outreach activities Supplement Purpose: (1) directly test the function of Zic3 in node cells in vivo and (2) delineate the requirement of Zic3 in node cells and cilia
Project Location:	Project Location: within the Quinault Indian Nation Worker Location: 3 remote communities on the Rosebud Sioux Tribe Reservation Facility Location: adjacent to the Tribal Headquarters Site Location: 910 Wrightsboro Road, Augusta, GA
Project Progress:	Entire Project Completed: The project has been completed Partially Completed: Less than 50% completed Program Completed: completed the REU program Activities Completed: Disconnection of Utilities Demolition and removal of existing structure
Position Category : (number of positions part-time/full-time)	Part Time: 3 part-time Full Time: Two full time Hourly: two hourly jobs
Position Type: (Created and/or Sustained)	Created: added Sustained: sustained

Even though the corpus contains enormous pattern varieties, our system could extract most of the relevant information from the test data. The system was also successful in extracting the target information that contains non-contiguous words. Because the system is based on underlying database operations; it should be able to handle even a larger corpus.

CONCLUSION AND FUTURE DIRECTION

Our method reduces manual processing of the training and thus, the development process was very fast. We have applied our system to extract information from the project-reports of the American Recovery and Reinvestment Act (ARRA) of 2009. The effectiveness of our system indicates that it can be used to assure transparency and accountability of management decisions by extracting and mining relevant information. In future, we have a plan to perform correlation analysis on the extracted information. Automated training methods may not feasible due to computational complexities of a large data corpus that we used. But, it might be possible to use

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

statistical analysis such as Point-wise Mutual Information (PMI) to identify the extraction rules automatically.

REFERENCES

Albanese, M., & Subrahmanian V. (2007). T-REX: A domain-independent system for automated cultural information extraction. *Proceedings of the First International Conference on Computational Cultural Dynamics*.

Ashish, N., Mehrotra, S., & Pirzadeh, P. (2009). XAR: an integrated framework for information extraction. *WRI World Congress on Computer Science and Information Engineering.*

Banko, M., & Etzioni, O. (2008). The tradeoffs between open and traditional relation extraction. *Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Betts, T., Milosavljevic, M., & Oberlander, J. (2007). The utility of information extraction in the classification of books. *Proceedings of the 29th European conference on IR research*.

Bhargavi, P., Jyothi, B., Jyothi, S., & Sekar, K. (2008). Knowledge extraction using rule based decision tree approach. *International Journal of Computer Science and Network Security*, 8(7).

Blohm, S. & Cimiano, P. (2007). Using the web to reduce data sparseness in pattern-based information extraction. *Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases.*

Bratus, S., Rumshisky, A., & Magar, R. (2009). Using domain knowledge for ontology-guided entity extraction from noisy, unstructured text data. *Proceedings of the 3rd Workshop on Analytics for Noisy Unstructured Text Data.*

Carven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchel, T., Nigam, K., & Slattery, S. (2000). Learning to construct knowledge bases from the world wide web. *Artificial Intelligence - Special issue on Intelligent internet systems*, 118(1-2).

Chieze1, E., Farzindar, A., & Lapalme, G (2010). An automatic system for summarization and information extraction of legal information. *Lecture Notes in Computer Science*, 6036, 216-234.

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Chiticariu L., Li, Y., Raghavan, S., & Reiss, F. (2010). Enterprise information extraction: recent developments and open challenges. *Proceedings of the international conference on Management of data.*

Ciravegna, F. & Wilks, Y. (2003). Designing adaptive information extraction for the semantic web in amilcare: annotation for the semantic web. *Frontiers in Artificial Intelligence and Applications, IOS.*

Dingli, A., Ciravegna, F., & Wilks, Y. (2003). Automatic semantic annotation using unsupervised information extraction and integration. *K-CAP Workshop on Knowledge Markup and Semantic Annotation.*

Dini, L., & Mazzini, G. (2002). Opinion classification through information extraction. *Data Mining III, WIT Press,* 299-310.

Downey, D., Schoenmackers, S., & Etzioni, O. (2007). Sparse information extraction: unsupervised language models to the rescue. *Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Eliassi-Rad, T., & Shavlik, J. (2003). Intelligent web agents that learn to retrieve and extract information. *Intelligent exploration of the web, Physica-Verlag GmbH, Heidelberg, Germany.*

Feldman, R., Aumann, Y., Finkelstein-Landau M, Hurvitz, E., Regev, Y., & Yaroshevich, A. (2002). A comparative study of information extraction strategies. *Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing, London, UK.*

Freitag, D. (2000). Machine learning for information extraction in informal domains. *Machine Learning, 39,* 169-202.

Fensel, D. (2001). Challenges in content management for B2B electronic commerce. *Proceedings of the 2nd International Workshop on User Interfaces to Data Intensive Systems.*

Fu, Y., & Mostafa, J. (2004). Toward information retrieval web services for digital libraries. *Proceedings of the Joint ACM/IEEE Conference on Digital Library.*

Ghani, R., Probst, K., Liu, Y., Krema, M., & Fano, A. (2006). Text mining for product attribute extraction. *ACM SIGKDD Explorations Newsletter, 8 (1).*

Glasgow, B., Alan, M., Binney, D., Ghemri, L., & Fisher, D. (1997). MITA: an information extraction approach to analysis of free-form text in life insurance applications. *Proceedings of the fourteenth national conference on artificial intelligence and ninth conference on Innovative applications of artificial intelligence*.

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Grant, G., & Conlon, S. (2006). Edgar extraction system: an automated approach to analyze employee stock option disclosures. *Journal of Information Systems*.

Guarino, N. (1997). Semantic matching: formal ontological distinctions for information organization, extraction, and integration. *Proceeding SCIE International Summer School on Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology.*

Habegger, B., & Quafafou, M. (2004). Web services for information extraction from the web. *Proceedings of the IEEE International Conference on Web Services.*

Hakkani-Tur D., & Tur, G. (2007). Statistical sentence extraction for information distillation. *IEEE International Conference on Acoustics, Speech and Signal Processing.*

Han, H., & Tokuda T. (2008). A method for integration of web applications based on information extraction. *Proceedings of the 2008 Eighth International Conference on Web Engineering, EEE Computer Society Washington, DC, USA.*

Heidorn, P., & Wei, Q. (2008). Automatic metadata extraction from museum specimen labels. *Proceedings of the Int'l Conference on Core and Metadata Applications, Dublin.*

Hendler, J., Berners-Lee, T., & Miller, E. (2002). Integrating applications on the semantic web. *Journal of the Institute of Electrical Engineers of Japan*, 122(10), 676-680.

Hoffmann, R., Amershi, S., Patel, K., Wu, F., Fogarty, J., & Weld, D. (2009). Amplifying community content creation with mixed-initiative information extraction. *Proceedings of the 27th international conference on Human factors in computing systems.*

Hsiao, M., Chen, C., & Chen J. (2009). Using UMLS to construct a generalized hierarchical concept-based dictionary of brain functions for information extraction from the FMRI literature. *Journal of Biomedical Informatics*, 42, 912–922.

Iria, J., & Ciravegna, F. (2006). A methodology and tool for representing language resources for information extraction. *Proceedings of Language Resources and Evaluation (LREC).*

Iria, J., & Ciravegna, F. (2005). Relation extraction for mining the semantic web. *Proceedings of Machine Learning for the Semantic Web, Dagstuhl, DE.*

Jain, A., Ipeirotis, P., & Gravano, L. (2008). Building query optimizers for information extraction, the SQoUT project. *Association of Computing Machineries SIGMOD Record,* 37(4).

Kalyanpur, A., Jennifer, G., Jay, B., & James, H. (2004), OWL: capturing semantic information using a standardized web ontology language. *Multilingual Computing and Technology Magazine,* 15(7).

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Kim, S., Jeong, M., Lee, G., Ko, K., & Lee Z. (2008). An alignment-based approach to semi-supervised relation extraction including multiple arguments. *Proceedings of the 4th Asia information retrieval conference on Information retrieval technology Springer-Verlag Berlin, Heidelberg.*

Kosseim, L., Beauregard, S., & Lapalme, G. (2001). Using information extraction and natural language generation to answer e-mail. *Data & Knowledge Engineering,* 38, 85-100.

Ku, C., Iriberri, A., & Leroy, G. (2008). Crime information extraction from police and witness narrative reports. *IEEE International Conference on Technologies for Homeland Security.*

Labsky, M., Svatek, S., Praks, P., & Svab, S. (2005). Information extraction from html product catalogues: coupling quantitative and knowledge-based approaches. *Proceedings of Dagstuhl Seminar on Machine Learning for the Semantic Web*.

Lee, H., Etnyre, V., & Chen, L. K. (2003). A study of .net framework, XML web services and supply chain management. *Journal of International Technology and Information Management, 12(1),* 137-153.

Leitner, F., & Valencia A. (2008). A text-mining perspective on the requirements for electronically annotated abstracts. *Federation of European Biochemical Societies Letters, 582,* 1178–1181.

Lo, M., & Hsieh, C. (2003). Mining the fx electronic inter-dealer market. *Journal of International Technology and Information Management, 12(1),* 61-76.

Maedche, A., & Staab, S. (2000). Mining ontologies from text. *Proceedings of the 12th European Workshop on Knowledge Acquisition, Modeling and Management.*

Manov, D., Kiryakov, A., & Popov, B. (2003). Experiments with geographic knowledge for information extraction. *Proceedings of the HLT-NAACL workshop on Analysis of geographic references, 1.*

McCallum, A., & Jensen, D. (2003). A note on the unification of information extraction and data mining using conditional-probability relational models. *IJCAI Workshop on Learning Statistical Models from Relational Data.*

McIlraith, S., Son, T., & Zeng, H. (2001). Semantic Web Services. *IEEE Intelligent Systems, 16(2),* 46-53.

Milosavljevic, M., Grover, C., & Corti, L. (2007). Smart qualitative data (SQUAD): information extraction in a large document archive. *RIAO Proceedings on Large Scale Semantic Access to Content (Text, Image, Video, and Sound).*

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Mooney, R., & Bunescu, R. (2005). Mining knowledge from text using information extraction. *ACM SIGKDD Explorations special issue on NLP and Text Mining*.

Negnevitsky, M. (2004). *Artificial Intelligence: A Guide to Intelligent Systems*, 2nd edition, November 12.

Nemati, R., Steiger, M., Iyer, S., & Herschel, T. (2002). Knowledge warehouse: an architectural integration of knowledge management, decision support, artificial intelligence, and data warehousing. *Decision Support Systems - Special issue: Directions for the next decade archive, 33(2).*

Pennacchiotti, M., & Pantel P. (2009). Entity extraction via ensemble semantics. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, (1).

Piao, Y., Zou, S., Wang, X., & Wang, Z. (2010). XML structure extraction from plain texts based on conditional random fields. *Journal of Computational Information Systems, 6 (8),* 2683-2690.

Riloff, E. (1993). Automatically constructing a dictionary for information extraction Task. *Proceedings of the 11th National Conference on Artificial Intelligence*, 811-816.

Rosendfeld, B., Feldman, R., & Fresko, M. (2006). TEG-a hybrid approach to information extraction. *Knowledge Information Systems, 9(1).*

Shen, W., Doan, A., Naughton, J., & Ramakrishnan, R. (2007). Declarative information extraction using datalog with embedded extraction predicates. *Proceedings of the 33rd International Conference on Very Large Data Bases.*

Sheth, A. (2005). Enterprise applications of semantic web: the sweet spot of risk and compliance. *Proceedings of IFIP International Conference on Industrial Applications of Semantic Web, Finland.*

Sekine, S. (2006). On-demand information extraction. *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, Sydney*, 731–738.

Sekine, S., and Oda, A. (2007). System demonstration of on-demand information extraction. *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*.

Sotelsek-Margalef, A., and Villena-Román, J. (2008). MIDAS: an information-extraction approach to medical text classification. *Natural language processing*, 41, 97-104.

Srihari, R., Li W., Niu, C., & Cornell, T. (2008). InfoXtract: a customizable intermediate level information extraction engine. *Natural Language Engineering*, 14(1).

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

Stevenson M., & Greenwood, M. (2009). Dependency pattern models for information extraction. *Research on Language & Computation*, 7(1), 13-39.

Suchanek, F., Sozio, M., & Weikum, G. (2009). SOFIE: a self-organizing framework for information extraction. *WWW Proceedings of the 18th international conference on World Wide Web.*

Tanev, H., & Magnini, B. (2008). Weakly supervised approaches for ontology population. *Proceeding of the 2008 conference on Ontology Learning and Population: Bridging the Gap between Text and Knowledge*.

Uramoto, N., Matsuzawa, H., Nagano, T., Murakami, A., Takeuchi, H., & Takeda, K. (2004). A text-mining system for knowledge discovery from biomedical documents. *IBM Systems Journal*, 43(3).

VargasVera, M., Motta, E., Domingue, J, Shum, S., & Lanzoni, M. (2001). Knowledge extraction by using an ontology based annotation tool. *K-CAP workshop on Knowledge Markup and Semantic Annotation.*

Welty C., & Murdock, J. (2006). Towards knowledge acquisition from information extraction. *Proceedings of the 5th International Semantic Web Conference*.

Wimalasuriya, D., & Dou, D. (2009). Using multiple ontologies in information extraction. *Proceeding of the 18th ACM conference on Information and knowledge management.*

Wu, F., & Weld, D. (2008). Open information extraction using wikipedia, *ACM SIGMOD Record*, 37(4).

Wu, J., Hsieh, C., Shin, S., & Wu, C. (2005). A methodology for evaluating data and output misfits in commercial off-the-shelf erp systems, *Journal of International Technology and Information Management*, 14(4), 27-44.

Xu, F., Uszkoreit, H., Li, H., & Felger, N (2008). Adaptation of relation extraction rules to new domains. *Proceedings of the Poster Session of the Sixth International Conference on Language Resources and Evaluation, Marrekech, Morocco*.

Yildiz, B., & Miksch, S. (2007). Motivating ontology-driven information extraction, international conference on semantic web and digital libraries, *Indian Statistical Institute Platinum Jubilee Conference Series*, 45-53.

Zhang, J., Y., Lin, Z., Lin, Q., & Hsieh, C. (2008). The readiness for and current status of e-goverenment in China. *Journal of International Technology and Information Management*, 17(1), 75-84.

  ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy

This Page Left Intentionally Blank

ISSN: 1543-5962-Printed Copy ISSN: 1941-6679-On-line Copy