Chapter 2

ADVANCES IN INFORMATION EXTRACTION

Jakub Piskorski
German Research Center for Artificial Intelligence
Saarbrücken
Germany

Abstract: Nowadays, knowledge relevant to business of any kind is mainly transmitted through free-text documents. Latest trends in information technology such as Information Extraction (IE) provide dramatic improvements in conversion of the overflow of raw textual information into valuable and structured data. This chapter gives a comprehensive introduction to information extraction technology including design, processing natural language, and evaluation issues of IE systems. Further, we present a retrospective overview of IE systems which have been successfully applied in real-world business applications which deal with processing vast amount of textual data, and we discuss current trends. Finally, we demonstrate an enormous indexing potential of lightweight linguistic text processing techniques in other areas of information technology closely related to IE.

Keywords: information extraction, free text processing, natural language processing

1. INTRODUCTION

For years possessing sound information at the right time has been an essential factor in the strategic planning and facilitating decision making in the area of business of any kind. Today's companies, governments, banks, and financial organizations are faced with monitoring a vast amount of information in digital form in a myriad of data repositories on Intranets and Internet. Unfortunately, the major part of electronically available information like newswire feeds, corporate reports, government documents or litigation records is transmitted through free-text documents und thus hard

to search in. One of the most difficult issues concerning applying search technology for retrieving information from textual data collections is the process of converting such data into a shape for searching. Hence, an ever-growing need for effective, efficient and intelligent techniques for analyzing free-text documents and discovering valuable and relevant knowledge from them in form of structured data can be observed.

Conventional Information Retrieval (IR) techniques used in WWW search engines (e.g., boolean queries, ranked-output systems) applied even to homogenous collections of textual documents fall far from obtaining optimal recall and precision simultaneously. For this reason, more sophisticated tools are needed that go beyond simple keywords search and are capable of automatically analyzing unstructured text documents in order to build expressive representations of their conceptual content and determine their relevance more precisely. Recent trends in information technology such as Information Extraction (IE) provide dramatic improvements in conversion of the overflow of raw textual information into structured data which could be further used as input for data mining engines for discovering more complex patterns in textual data collections.

The task of IE is to identify predefined set of concepts in a specific domain and ignoring other irrelevant information, where domain consists of a corpus of texts together with a clearly specified information need. Even in a specific domain it is a non-trivial task due to the phenomena and complexity of natural language. For instance, there are obviously many ways of expressing the same fact, which on the other side could be distributed across several sentences. Furthermore, implicit information is not easy to discern and an enormous amount of knowledge is needed to infer the meaning of unrestricted natural language. The process of understanding language comes quite naturally to most humans, but it is very difficult to model this process in a computer, which has been frustrating the design of intelligent language understanding software.

The potential value of automatically structuring natural language data has been already recognized in the 1950's and 1960's. Currently, Natural Language Processing[1] (NLP) techniques are applied in both IR and IE in order to build rich representations of the analyzed texts. However, it has been shown that from technical point of view realizing high accurate general full text understanding system is at least today impossible. Recent advances

[1] "NLP is a branch of computer science that studies computer systems for processing natural languages. It includes the development of algorithms for parsing, generation and acquisition of linguistic knowledge; the investigation of time and space complexity of such algorithms; the design of computationally useful formal languages (such as grammar and lexicon formalisms) for encoding linguistic knowledge; the investigation of appropriate software architectures for various NLP tasks; and consideration of the types of non-linguistic knowledge that impinge on NLP" [Gazdar, 1996].

in NLP concerning new robust, efficient and high-coverage shallow text processing techniques instead of fully-fledged linguistic analysis contributed to the size in the deployment of IE techniques in real-world business information systems dealing with huge amount of textual data. The use of light-weight linguistic analysis tools instead of full text understanding systems may be advantageous since it might be sufficient for the extraction and assembly of the relevant information and it requires less knowledge engineering, which means a faster development cycle and fewer development expenses.

The rest of this chapter is organized as follows. Section 2 introduces the IE task, issues concerning design and evaluation of IE systems, and utilization of shallow text processing. The IE systems successfully applied in the financial, insurance and legal domain are presented in section 3. Section 4 focuses on presenting some fields in information technology closely related to IE, such as Text Mining which benefit from applying similar light-weight linguistic analysis techniques. Finally, we end with a summary in section 5.

2. INFORMATION EXTRACTION

2.1 Information Extraction Task

The task of *Information Extraction* (IE) is to identify instances of a particular pre-specified class of entities, events and relationships in natural language texts, and the extraction of the relevant arguments of the identified events or relationships [SAIC, 1998]. The information to be extracted is pre-specified in user-defined structures called templates (e.g., product or company information, management succession event), each consisting of a number of slots, which must be instantiated by an IE system as it processes the text. The slots are usually filled with: some strings from the text, one of a number of pre-defined values or a reference to other already generated template. One way of thinking about an IE system is in terms of database construction since an IE system creates a structured representation (e.g., database entries) of selected information drawn from the analyzed text. From the viewpoint of NLP information extraction is very attractive since its task is well defined, it uses real-world texts and possesses difficult and interesting NLP problems.

In the last decade IE technology has progressed quite rapidly, from small-scale systems applicable within very limited domains to useful systems which can perform information extraction from a very broad range of texts.

IE technology is now coming to the market and is of great significance to publishers, banks, financial companies, and governments. For instance, a financial organization want to know facts about company take-overs, executive successions and foundations of international joint-ventures happening in a given time span. In figure 2-1 an example of an instantiated template for joint-venture foundation event is presented.

"Munich, February 18, 1997, Siemens AG and The General Electric Company (GEC), London, have merged their UK private communication systems and networks activities to form a new company, Siemens GEC Communication Systems Limited."

$$\begin{bmatrix} \text{VENTURE} : \textit{Siemens GEC Communication Systems Limited} \\ \text{PARTNERS} : \textit{Siemens AG}, \textit{The General Electr}\text{ic} \\ \text{TIME} : \textit{February}\ 18, 1997 \\ \text{PRODUCT} : \textit{communication systems}, \textit{network activities} \end{bmatrix}$$

Figure 2-1. An instantiated template for joint-venture foundation event

The process of extracting such information involves locating names of companies and finding linguistic relations between them and other relevant entities (e.g., locations, temporal expressions). However, in this scenario an IE system requires some specific domain knowledge (understanding the fact that ventures generally involve at least two partners and result in the formation of a new company) in order to merge partial information into an adequate template structure. Generally, IE systems rely always to some degree on domain knowledge. Further information such as appointment of key personnel, announcement of new investment plans, or other market related data could also be reduced to instantiated templates.

The templates generated by an IE system may be incomplete (partially instantiated templates), but incomplete information is still better than no information at all since it could be used to focus users attention on the passage of text containing targeted information.

2.2 Designing IE Systems

There are two basic approaches to designing IE systems: Knowledge Engineering Approach and Learning Approach [Appelt and Israel, 1999]. In the knowledge engineering approach the development of rules for marking and extracting sought-after information is done by a human expert through inspection of the test corpus and his or her own intuition. Therefore, the skill of the knowledge expert is a major factor which impacts the level of performance of an IE system built in this way. Constructing a set of high-coverage extraction rules is usually done iteratively. Initially, a set of rules is written and tested against a test corpus in order to check whether they

undergenerate or overgenerate. Subsequently, the rule set is appropriately modified and the process iterates till no significant improvement can be achieved.

In the learning approach the rules are learned from an annotated corpus and interaction with the user. This involves utilization of machine learning techniques based on Hidden Markov Models, Maximum Entropy Modeling, and Decision Trees [Manning and Schütze, 1999]. In this approach, large quantities of training data are a major prerequisite for achieving high accuracy. Obviously, the annotation of texts for the information being extracted requires less skill than manual construction of extraction patterns, but it is still a laborious task. Once a suitable training corpus is provided, the system can be ported to a new domain in a relatively straightforward manner. A debate on advantages and disadvantages of both approaches is given in [Appelt and Israel, 1999]. Generally, higher performance can be achieved by handcrafted systems, particularly when training data is sparse. However, in a particular scenario automatically trained components of an IE system might outperform their handcrafted counterparts. Approaches to building hybrid systems based on both approaches are currently being investigated.

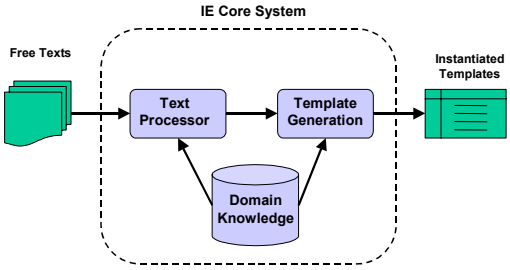

Figure 2-2. A coarse-grained architecture of an IE system

IE systems built for different tasks often differ from each other in many ways. Nevertheless, there are core components shared by nearly every IE system, disregarding the underlying design approach. The coarse-grained architecture of a typical IE system is presented in figure 2-2. It consists of two main components: text processor and template generation module. The task of the text processor is performing general linguistic analysis in order to extract as much linguistic structure as possible [Piskorski and Skut, 2000].

The scope of information computed by the text processor may vary depending on the requirements of a particular application. Usually, following steps are performed:

– Segmentation of text into a sequence of sentences, each of which is a sequence of lexical items representing words together with their lexical attributes,

– Recognition of small scale-structures (e.g., abbreviations, core nominal phrases, verb clusters and named entities),

– Parsing, which takes as input a sequence of lexical items and small-scale structures and computes a dependency structure of the sentence, the so called parse tree.

FRAGMENT	**TYPE**
Munich,	LOCATION
February 18, 1997	DATE
Siemens AG	COMPANY NAME
and	CONJUNCTION
The General Electric Company (GEC), London	COMPANY NAME
have merged	VERB GROUP
their UK private communication systems and networks activities	NOUN PHRASE
to form	VERB GROUP
a new company	NOUN PHRASE
Siemens GEC Communication Systems Limited	COMPANY NAME

Figure 2-3. Recognition of small-scale structures for the sentence presented in the figure 2-1

Depending on the application scenario it might be desirable for the text processor to perform additional tasks, such as: part-of-speech disambiguation, word sense tagging, anaphora resolution or semantic interpretation, i.e., translating parse tree or parse fragments into a semantic structure or logical form. A benefit of the IE task orientation is that it helps to focus on linguistic phenomena that are most prevalent in a particular domain or a particular extraction task. An example of a result of recognition of small-scale structures for the sentence in figure 2-1 is given in the table in figure 2-3. Recently, a general tendency towards applying partial text analysis instead of computing all possible interpretations could be observed. We explore this issue in the next subsection.

The task of the template generation module is to merge the linguistic structures computed by the text processor and to derive scenario-specific relations in form of instantiated templates using domain knowledge (e.g., via domain-specific extraction patterns and inference rules). Since linguistic analysis, except anaphora resolution operates within a scope of individual sentences, and due to the fact that filling certain templates requires merging structures extracted from different sentences, the process of template

generation is usually divided into: (a) creation of templates representing event and entity description, and (b) template merging. In the first step, domain-specific pattern/action rules map linguistic structures into corresponding templates, whereas in the second step, similarity between templates triggers appropriate template merging procedures in order to assemble scattered information pieces. Similarity measures usually rely on the distance and the degree of the overlap between the templates considered [Kehler, 1998]. In figure 2-4, we give an example of template merging.

"Mr. Diagne would leave his job as a vice-president of Yves Saint Laurent, Inc. to become operations director of Paco Raban. John Smith **replaced** Diagne."

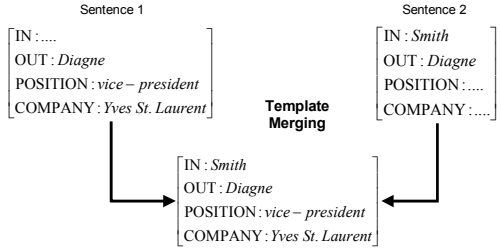

Figure 2-4. Template Merging

In practice, the boundary between text processor and template generation component may be blurred.

2.3 Shallow vs. Deep Text Processing

The main problem one has to cope with when processing free texts is the fact that natural languages are massively ambiguous. The problem of ambiguity pervades all levels of natural language processing. Individual words may have often number of meanings and are used to refer to different things on different occasions of utterance. Further, words or phrases contained in a sentence may be related to one another in more than one way. To illustrate this, consider the following sentence "Simpson saw the terrorist with the telescope" which is structurally ambiguous since it can be interpreted as providing information about which terrorist Simpson saw (the one with a telescope) or about what instrument Simpson used to see the terrorist. The problem we are dealing here with, is the so called PP-Attachment (prepositional phrase attachment). Ambiguities regarding sentence boundaries, clause boundaries and structure, part-of-speech labels and word meanings all complicate sentence analysis. Many sources of ambiguity become simplified when the domain is restricted. For instance, the

word “joint” appearing in news articles focusing on business tie-ups is mostly used as an adjective and denotes some sort of joint business activity, whereas in the medical articles one might expect that the word “joint” would be used more frequently as a noun.

The task of *deep text processing* (DTP) is the process of computing all possible interpretations and grammatical relations in natural language text. Because of the high complexity of such full linguistic analysis [Cole et al., 1996] and due to fact that correctly determining such information is not always necessary and may be a waste of time, there is an increased tendency towards applying only partial analysis, so called *shallow text processing* (STP) which is generally considerably less time-consuming and could also be seen as trade-off between simple pattern matching and fully-fledged linguistic analysis.

STP could be briefly characterized as a process of computing text analysis which are less complete than the output of DTP systems. It is usually restricted to identifying non-recursive structures or structures with limited amount of structural recursion, which can be identified with a high degree of certainty. Language regularities which cause problems are not handled and instead of computing all possible readings a STP-engine computes only underspecified structures. Let us consider as an example the recognition of compounds which are massively used in business texts (e.g., in the German “Wirtschaftswoche” corpus consisting of business news, circa 7,2% of the words are compounds). The syntactic structure of a compound may be complex and ambiguous. For example, the structure of the German compound "Biergartenfest" (*beer garden party*) could be [beer [garden party]] (*garden party with beer*) or [[beer garden] party] (*party in the beer-pub*). Furthermore, a compound may have more than just one valid syntactic segmentation (e.g., the German compound “Weinsorten” could be decomposed into “Weins + orten” (*wine places*) or “Wein + sorten” (*wine brands*). Since semantically correct segmentation of compounds as well as computation of their internal structure requires a great deal of knowledge, a STP engine would usually compute a single syntactically valid segmentation and determine the head while leaving internal bracketing underspecified [Neumann and Piskorski, 2002]. On the other side, computing information about all possible segmentations and internal bracketings might be even unnecessary for successfully performing most of the real-world IE tasks.

The term shallow text analysis usually refers to identifying named entities and some phrasal constituents (e.g., base noun phrases, verb clusters) without indicating their internal structure and function in the sentence, but does not have to be restricted to recognizing only this kind of information. Currently developed STP systems are usually based on finite-state technology. The tendency towards applying finite-state technology can be

briefly motivated in two ways. Firstly, finite-state devices are time and space efficient due to their closure properties and the existence of efficient optimization algorithms. Secondly, the local natural language phenomena can be easily and intuitively expressed as finite-state devices. Moreover, the linguistic description of a given phenomena can be usually broken into a number of autonomous finite-state sub-descriptions. In this way, a finite-state cascade allows for a strong decomposition of the linguistic analysis into many subtasks, ordered by increasing complexity. The strongly incremental character of this approach entails simplicity of the grammars w.r.t. both size and facility of modification. Obviously, there exist much more powerful formalisms like context-free or unification based grammars [Cole et al., 1996] which allow to describe phenomena beyond the descriptive capacity of finite-state devices, but since industry prefers more pragmatic solutions finite-state based approaches are recently in the center of attention. It is not only due to the higher time complexity, but also due to somewhat more recursive character of these formalisms, which causes debugging and modifying higher-level grammar a more elusive task (e.g., changes to a rule in a context-free grammar may in principle influence the application of any rule in the grammar). Furthermore, finite-state approximation grammars [Mohri and Nederhof, 2001] have been shown to provide a surprisingly effective engine for partial parsing system design. STP-engines based on the finite-state cascades proved to be almost as accurate as those based on more complex formalisms.

One of the major bottlenecks of DTP systems is the lack of robustness, i.e., they either return large number of concurrent analyses or none at all. STP tackles this problem via underspecification and ranking (weighted grammars), which limits the number of analyses and guarantees higher probability of producing at least one analysis [Aït-Mokhtar et al., 2002].

2.4 IE-oriented NLP Platforms

Since the beginning of 90's, the development of shallow and robust parsing systems has emerged [Chanod, 2000]. On the one pole various efficient monolingual shallow text processors have been introduced. [Piskorski and Neumann, 2000] presents SPPC, an efficient, robust and high-performance STP-engine which make exhaustive use of finite-state technology. It consists of several linguistic components ranging from tokenization to subclause and sentence structure recognition. SPPC provides IE-oriented tools, such as frequency-based term extraction and automatic generation of lexico-syntactic patterns based on a bag of seed words [Finkelstein-Landau and Morin, 1999]. For instance, patterns like: [ORG] "and" [ORG] "merge" [NP] "and" [NP], which covers partially the joint-

venture foundation event from the sentence in figure 2-1 can be automatically acquired. The current version of SPPC has been in particular fine-tuned for analyzing German business documents up to several megabytes with an average of processing 30 000 words per second, which together with its high linguistic coverage reflects the state-of-the-art performance and functionality of STP systems, and demonstrates their real-world application maturity.

An Achilles heel of early STP systems was the fact that they were dedicated to processing texts in a single language (mainly English and couple of other major languages) and tailored to a specific domain, and had somewhat black-box character. Therefore, the emphasis on multilinguality and reusability has grown, which is crucial for increasing the suitability of end-user IE applications in the process of customization. [Cunningham, 2002] presented GATE, a framework for development of language processing components and systems. It provides a set of prefabricated ready-made and reusable software blocks for basic operations such as tokenization, morphological analysis, or finite-state transduction on annotated documents. They can be coupled in order to form a system instance in a straightforward manner. GATE allows for integrating of new and external processing resources at run-time. Further, it provides a visual system development environment and resources management tools. Several multilingual text extraction tools were built on top of it, including mainly named-entity recognition systems [Maynard et al., 2002]. Ellogon, a text-engineering platform presented in [Petasis et al., 2002] resembles to large extent the GATE framework. Its key added values as the authors claim are full Unicode support, an extensive multilingual GUI, reduced hardware requirements (implemented in C++, whereas GATE is mainly implemented in JAVA), and supporting a wide range of operating systems. Mainly the user-friendly graphical environment for development and integration of linguistic components makes this platform particularly attractive w.r.t. constructing IE systems.

An important aspect in the context of development of IE systems is the specification language for grammar writing. The possibly most widely spread system GATE (over 2500 users worldwide) uses JAPE (Java Annotation Pattern Engine). A JAPE grammar contains pattern/action rules, where the left-hand side (LHS) of a rule is a regular expression over atomic feature-value constraints, while the right-hand side (RHS) of a rule is a so called annotation manipulation statement for output production, which calls native code (C++, Java), making rule writing difficult for non-programmers. An attempt to find a compromise between processing efficiency and expressiveness was done in SProUT [Becker et al., 2002], another platform for development of STP engines. The grammar formalism of SProUT can be

seen as an amalgamation of finite-state and unification-based techniques, where a grammar is a set of pattern/action rules, where the LHS of a rule is a regular expression over typed feature structures with functional operators and coreferences, representing the recognition pattern, and the RHS of a rule is a seqeuence of typed feature structures, specifying the output structure. Coreferences provide a stronger expressiveness since they create dynamic value assignments and serve as means of information transport into the output descriptions. The typed feature structures are used as a uniform I/O data structure which ensures a smooth communication between components, and supports cascaded architectures.

2.5 Evaluation of IE Systems

The input and output of an IE system can be defined precisely, which facilitates the evaluation of different systems and approaches. For the evaluation of IE systems the *precision* and *recall* measures were adopted from the IR research community. These metrics may be viewed as estimating systems effectiveness from the user's perspective, since they measure the extent to which the system produces all the appropriate output (recall) and only the appropriate output (precision). We define these measures formally. Let N_{key} be the total number of slots expected to be filled according to a reference comprising of an annotated corpus representing ground truth, and let $N_{correct}$ be the number of correctly filled slots in the system response. Further, let $N_{incorrect}$ denote the number of incorrectly filled slots, where a slot is said to be filled incorrectly, either if it does not allign with a slot in the reference (spurious slot) or if it is scored as incorrect (i.e., has invalid value). Then, precision and recall are defined as follows:

$$precision = \frac{N_{correct}}{N_{correct} + N_{incorrect}} \qquad recall = \frac{N_{correct}}{N_{key}}$$

Intuitively, it is impossible for an IE system to achieve 100% recall except on the trivial tasks, since textual documents offer differing amounts of relevant information to be extracted and the proper answers ocassionally do not come from a closed set of predetermined solutions. Sometimes the *F-measure* is used as a weighted harmonic mean of precision and recall, where β value is used to adjust their relative weighting (β gives equal weighting to recall and precision, and lower values of β give increasing weight to precision). Formally we define the F-measure as follows:

$$F = \frac{(\beta^2 + 1) \times precision \times recall}{\beta^2 \times precision + recall}$$

On top of these measures a further, less known measure, the so called *slot error rate*, *SER* [Makhoul et al., 1999] is defined as follows:

$$SER = \frac{N_{incorrect} + N_{missed}}{N_{key}}$$

,where N_{missed} denotes the number of slots in the reference that do not allign with any slots in the system response. It is simply the ratio between the total number of slot errors and the total number of slots in the reference. For particular need, certain error types may be weighted in order to deem them more or less important than others.

2.6 IE vs. IR

An IR system finds relevant texts and presents them to the user, whereas typical IE system analyzes texts and presents only specific user-relevant information extracted from them. IE systems are obviously more difficult and knowledge intensive to build and they are in particular more computationally intensive than IR systems. Generally, IE systems achieve higher precision than IR systems. However, IE and IR techniques can be seen as complementary and can potentially be combined in various ways. For instance, IR could be embedded within IE for pre-processing a huge document collection to a manageable subset to which IE techniques could be applied [Gaizauskas and Robertson, 1997]. On the other side, IE can be used as a subcomponent of an IR system to identify terms for intelligent document indexing (e.g., conceptual indicies). Such combinations clearly represent significant improvement in retrieval of accurate and prompt information (cf. figure 2-5). For instance, [Mihalcea and Moldovan, 2001] introduced an approach for document indexing using named entities, which proved to reduce the number of retrieved documents by a factor of 2, while still retrieving relevant documents.

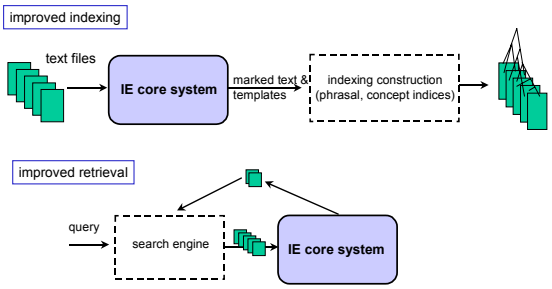

Figure 2-5. The advanced IE technologies improve intelligent indexing and retrieval

The main gain obtained by enriching the texts with named-entity tags, is that this enables the formation of queries that include answer types in addition to the keywords specified in the input question.

2.7 Message Understanding Conferences

The rapid development of the field of IE has been essentially influenced by the Message Understanding Conferences (MUC). These conferences were conducted under the auspices of several United States government agencies with the intention to coordinate multiple research groups and government agencies seeking to improve IE and IR technologies [Grishman and Sundheim, 1996]. The proceedings of MUC provide an important reference to the current state-of-the-art results and techniques used in the field of information extraction. The MUC conferences defined several generic types of IE tasks. These were intended to be prototypes of IE tasks that arise in real-world applications and they illustrate the main functional capabilities of current IE systems. The tasks defined in these conferences are of central importance to the field, since they constitute the most rigorously defined set of information specifications, information representation formats, and a set of corpora that are widely available to the research community. Hence, they provide a framework within which current approaches and systems may be evaluated.

The MUC-1 (1987) and MUC-2 (1989) focused on automated analysis of military messages containing textual information about naval sightings and engagements, where the template to be extracted had 10 slots. Since MUC-3 (1991) the task shifted to information extraction from newswire articles (e.g., concerning terrorist events, international joint-venture foundations, management succession, microelectronics, and space vehicle and missile launches) and templates became somewhat more complex. In MUC-5 the joint-venture task required 11 templates with a total of 47 slots for the output. Further, in MUC-5, the nested template structure and multilingual IE

were introduced. In MUC-6 (1995), the IE task was subdivided into subtasks in order to identify task-independent component technologies of IE task which could be immediately useful. The generic IE tasks for MUC-7 (1998) provide progressively higher-level information about texts and they were defined as follows:

(NE) Named Entity Recognition Task requires the identification and classification of named entities such as organizations, persons, locations, temporal expressions (e.g., date, time), and quantities (e.g., monetary values), which has been singled out for this task.

(TE) Template Element Task requires the filling of small-scale templates for specified classes of entities in the texts, such as organizations, persons, certain artifacts with slots such as name, name variants, title, nationality, description as supplied in the text, and subtype. Generally, attributes of entities are used as slot fillings (see template for organization and person in figure 2-6). In other words, TE associates descriptive information with the entities.

(TR) Template Relation Task requires filling a two-slot template representing a binary relation with pointers to template elements standing in the relation, which were previously identified in the TE task. This might be, for instance, an employee relation between a person and a company (cf. figure 2-6), a 'product-of' relation (cf. figure 2-7), or a subsidiary relationship between two companies. The possibilities in real-world applications are endless.

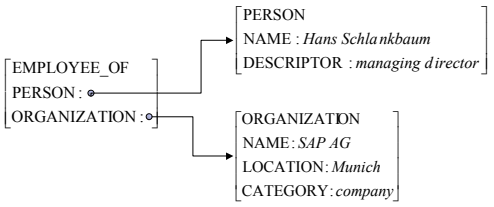

Figure 2-6. An instantiated template representing the "employee-of" relation

(CO) Co-reference Resolution requires the identification of expressions in the text that refer to the same object, set or activity. These include variant forms of name expressions, definite noun phrases and their antecedents, and pronouns and their antecedents (e.g., in the sentence in figure 2-1 the possessive pronoun "their" has to be associated with "Siemens AG" and "The General Electric Company". The CO task can be seen as a bridge

between NE task and TE task. It differs from other MUC tasks since it constitutes a component technology.

(ST) Scenario Template Task requires filling a template structure with extracted information involving several relations or events of interest, for instance, identification of partners, products, profits and capitalization of joint ventures (see figure 2-6). Scenario templates tie together TE entities and TR relations into event descriptions. ST task is intended to be the MUC approximation to a real-world IE problem.

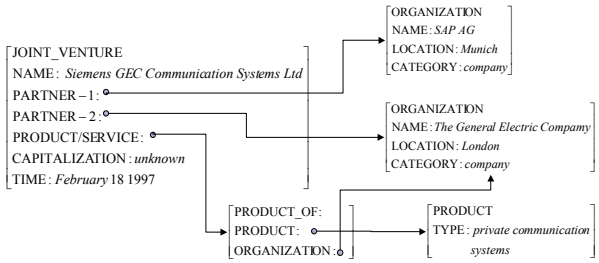

Figure 2-7. An instantiated scenario template for joint-venture

The table in figure 2-8 gives an overview of the tasks evaluated in MUC-3 through MUC-7.

Con\Task	NE	CO	RE	TR	ST
MUC-3					YES
MUC-4					YES
MUC-5					YES
MUC-6	YES	YES	YES		YES
MUC-7	YES	YES	YES	YES	YES

Figure 2-8. Tasks evaluated in MUC-3 through MUC-7

In a MUC evaluation, participants were initially given a detailed description of the scenario along with the annotated training data (*training corpus*) in order to adapt their systems to the new scenario (1 to 6 months). After this, each participant received a new set of documents (*test corpus*), applied their systems to extract information from these documents, and returned the extracted templates to the conference organizer. These results were then compared to the manually filled set of templates (*answer key*). State-of-the-art results for IE tasks for English reported in MUC-7 are presented in figure 2-9. Some experiments were conducted in order to compare the best system results and results obtained by human annotators.

For instance, in the NE task for MUC-7 two human annotators achieved an F-measure of 96.9% and 98.6%. Hence, NE recognition can now be said to function at human performance level. However, for all other tasks IE systems are less accurate than human annotators. The human score for TE and ST task can be around 93% and 80% respectively, where the latter result illustrates the complexity involved. [Appelt and Israel, 1999] argued that 60% can be considered as an upper bound on the proportion of relevant information which is expressed in an explicit and straightforward way, and can be extracted without involving sophisticated linguistic analysis.

MEASURE\TASK	NE	CO	RE	TR	ST
RECALL	92	56	86	67	42
PRECISION	95	69	87	86	65

Figure 2-9. Maximum results reported in MUC-7

In MUC-7, for the first time, the evaluation of some tasks (e.g., NE task) was run using training and test corpus from comparable domains for all languages considered (Chinese, Spanish and Japanese). The results were on an average slightly worse than the results for English (e.g., F-measure of ca. 91% and 87% for Chinese and Japanese respectively) [Chinchor, 1998]. [Sang, 2002] reported on the evaluation of systems participating in a shared language-independent NE task conducted within the Conference on Natural Language Learning (CoNLL-2002). Twelve different systems have been applied to Spanish and Dutch corpora, where the best systems achieved an F-measure of 81.4% and 77.1% respectively.

3. IE SYSTEMS IN THE BUSINESS DOMAIN

3.1 Early IE Systems

The earliest IE systems were deployed as commercial product already in the late eighties. One of the first attempts to apply IE in the financial field using templates was the ATRANS system [Lytinen and Gershman, 1993], based on simple language processing techniques and script-frames approach for extracting information from telex messages regarding money transfers between banks. JASPER [Andersen et al., 1992] is a IE system that extracts information from reports on corporate earnings from small sentences fragments using robust NLP methods. SCISOR [Jacobs et al., 1990] is an integrated system incorporating IE for extraction of facts related to the company and financial information (corporate mergers and acquisitions).

These early IE systems had a major shortcoming, namely they were not easily adaptable to new scenarios. On the other side, they demonstrated that relatively simple NLP techniques are sufficient for solving IE tasks narrow in scope and utility.

3.2 LOLITA

The LOLITA System [Costantino et al., 1997], developed at the University of Durham, was the first general purpose IE system with fine-grained classification of predefined templates relevant to the financial domain. Further, it provides a user-friendly interface for defining new templates. LOLITA is based on deep natural language understanding and uses semantic networks. Different applications were built around its core. Among others, LOLITA was used for extracting information from financial news articles which represent an extremely wide domain, including different kind of news (e.g., financial, economical, political, etc.). The templates have been defined according to the "financial activities" approach and can be used by the financial operators to support their decision making process and to analyze the effect of news on price behavior. A financial activity is one potentially able to influence the decisions of the players in the market (e.g., brokers, investors, analysts etc.).

The system uses three main groups of templates for financial activities: company related activities – related to the life of the company, company restructuring activities - related to changes in the productive structure of companies and general macroeconomics activities, including general macro-economics news that can affect the prices of the shares quoted in the stock exchange. The table in figure 2-10 gives some examples of templates extracted by the LOLITA system.

Company related activities	ownership, shares, takeovers, mergers, new issue of shares, privatization, market movements, dividend announcement, sales forecasts, investigations, profits/sales results, legal action
Company restructuring activities	new product, joint venture, staff change, new factory
General macroeconomics activities	interest rate movements, currency movements, inflation, unemployment, trade deficit

Figure 2-10. Templates extracted by the LOLITA IE system

In the “takeover template” task, as defined in MUC-6, the system achieved precision of 63% and recall of 43%. However, since the system is

based on DTP techniques, the performance in terms of speed can be, in particular situations, penalized in comparison to STP-based systems. The output of LOLITA was fed to the financial expert system [Constantino, 1999] to process an incoming stream of news from on-line news providers, companies and other structured numerical market data to produce investment suggestions.

3.3 MITA

IE technology has been recently successfully used in the insurance domain. MITA (Metalife's Intelligent Text Analyser) was developed in order to improve the insurance underwriting process [Glasgow et al., 1998]. The Metalife's life insurance applications contain free-form textual fields (an average of 2.3 textual fields per application) such as: physician reason field - describing a reason a proposed insured last visited a personal physician, family history field – describing insured's family medical history and major treatments and exams field which describes any major medical event within the last five years.

In order to identify any concepts from such textual fields that might have underwriting significance, the system applies STP techniques and returns a categorization of these concepts for risk assessment by subsequent domain-specific analyzers. For instance, MITA extracts a 3-slot template from the family history field. The concept slot in the output structure describes a particular type of information that can be found in a specific field, the value slot is the actual word associated with particular instance of the concept and the class slot denotes the semantic class that the value denotes (cf. figure 2-11).

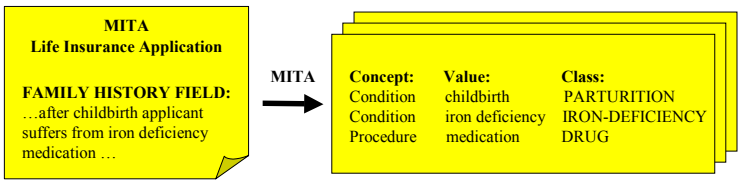

Figure 2-11. An example of output generated by MITA

The MITA system has been tested in a production environment and 89% of the information in textual field was successfully analyzed. Further, a blind testing was undertaken to determine whether the output of MITA is sufficient to make underwriting decisions equivalent to those produced by an underwriter with access to the full text. Evaluation of the results showed that

only up to 7% of the extractions resulted in different underwriting conclusions.

3.4 History Assistant

[Jackson et al., 1998] presents History Assistant - an information extraction and retrieval system for the juridical domain. It extracts rulings from electronically imported court opinions and retrieves relevant prior cases and cases affected from a citator database, and links them to the current case. The role of a citator database enriched with such linking information is to track historical relations among cases. On-line citators are of great interest to the legal profession because they provide a way of testing whether a case is still good law.

History Assistant is based on DTP. In particular, it uses context-free grammars for computing all possible parses of the sentence. The problem of identifying the prior case is a non-trivial task since citations for prior case are usually not explicitly visible. History Assistant applies IE for producing structured information blocks, which are used for automatically generating SQL queries to search prior and affected cases in the citator database. Since information obtained by the IE module might be incomplete, additional domain specific knowledge (e.g., court hierarchy) is used in cases when extracted information does not contain enough data to form a good query. The automatically generated SQL query returns a list of cases which are then scored using additional criteria. The system achieved a recall of 93.3% in the prior case retrieval task (i.e., in 631 out of the 673 cases the system found the prior case as a result of an automatically generated query).

3.5 Trends

The most recent approaches to IE concentrated on constructing general purpose, highly modular, robust, efficient and domain adaptive IE systems.

FASTUS [Hobbs et al., 1997] is a very fast and robust general purpose IE system for English and Japanese, built in the Artificial Intelligence Center of SRI International. It is conceptually very simple, since it works essentially as a set of cascaded nondeterministic finite-state transducers. The main design goal of this system was to take full advantage of finite-state technology and represent each stage of processing as finite-state device. The composite output structures of successive stages of processing provide the input for the next stage. FASTUS was one of the best scoring systems in the MUC competitions and was utilized by a commercial client for discovering an ontology underlying complex Congressional bills, for ensuring the consistency of laws with the regulations that implement them. Further, parts

of FASTUS were converted into a commercial product for purposes of named entity recognition.

[Humphreys et al., 1998] describe LaSIE-II, a highly flexible and modular IE system, which was an attempt to find a pragmatic middle way in the shallow vs. deep analysis debate which characterized the last several MUCs. The result is an eclectic mixture of techniques ranging from finite-state recognition of domain-specific lexical patterns to using restricted context-free grammars for partial parsing. Its highly modularized architecture (9 submodules) enabled one to take deeper insight into strengths and weaknesses of the particular subcomponents of the system and their interaction. Furthermore, this system provides graphical tools for selecting the control flow through different module combinations. Lasie-II was the only system which took part in all of the MUC-7 tasks and achieved fairly good results (e.g., in ST task recall of 47 % and 42% precision).

Similarly to LaSIE-II, the two top design requirements of the IE2 system [Aone et al., 1999], developed at SRA International Inc. were modularity and flexibility. SGML was used to spell out system interface requirements between the sub-modules of the system, which allow an easy replacement of any sub-module in the workflow. Hence, each module can be developed, tested and improved independently of the other. Further, IE2 provides annotation tool for creating training corpora and visual diagnostic tools for evaluating and debugging the IE submodules which obviously speeds up the development process significantly. The IE2 system achieved the highest score in TE task (recall: 86%, precision 87%), TR task (recall: 67%, precision: 86%) and ST task (recall: 42%, precision: 65%) in the MUC-7 competition.

REES (Relation and Event Extraction System) – a close cousin of IE2, presented in [Aone and Santacruz, 2000], was the first attempt to constructing large-scale event and relation extraction system based on STP methods. It can extract more than 100 types of relations and events related to the area of business, finance and politics, which represent much wider coverage than is typical of IE systems. For 26 types of events related to finance it achieved an *F*-measure of 70%.

[Maynard et al., 2002] presented MUSE, a cross-genre entity recognition system, which borrows some ideas of LaSIE-II, but is mainly based on the finite-state transduction grammar formalism provided in the GATE framework. It was designed to robustly process multiple types of text, with minimal adaptation requirements, through the use of a set of resource switches, which operate according to certain linguistic or other features of the text (e.g., text format triggers different grammar rules). The evaluation of the system demonstrated that the robustness of NE extraction in the context

of multiple genres is comparable to robustness achieved by single-genre systems.

Pinocchio, is another environment for developing and running IE applications, which is based on a finite-state approximation of full parsing is presented in [Ciravegna et al., 2000].

3.6 Commercial Systems

The rapid progress in the field of IE fueled an ever-growing interest in the development of commercial IE software for the use in an industrial, governmental and educational context.

Teragram, (http://www.teragram.com) presented Concepts Extractor, a customizable system which automatically detects and extracts concepts from text and documents (based on simple linguistic analysis), such as people's names, company names, publicly traded companies, people's titles and positions, and geographical locations. Additionally, it can also output information associated with the extracted concepts. For example, the extracted publicly traded companies are associated with their ticker symbol and the stock market where they are listed.

The Insight Discoverer Extractor developed by Temis (http://www.temis-group.com), based in France, uses finite-state based shallow processing techniques (morpho-syntactic and semantic tagging and named entity recognition) to extract concepts and relations between concepts. In particular, the extraction patterns are regular expressions over lemmata (cannonical word forms), and syntactic or semantic labels. Furthermore, the patterns allow for defining roles to the extracted entities. For instance, the pattern: [PERSON: IN] "replaced" [PERSON:OUT], would help to fill the template for the second sentence in the example of template merging given in figure 2-4. This product has been deployed in various contexts, including customer email analysis, automatic internet watch for relevant information, and news-wires summarization.

One of the best known commercial ventures that has been spun out of the research work in the field of IE at university, is Cymfony, based in Buffalo, USA (http://www.cymfony.com). Cymfony's InfoXtract is a system which utilizes a leveraged combination of state-of-the-art statistical and grammar-based approaches for extracting entities, relationships between entities, and events of interest. In particular, this tool is capable of identification of numerous key relationships involving industries, companies, people and brands. In order to achieve high speed and robustness, the technique of cascaded shallow grammars has been exploited, where each level of the cascade contributes to increasingly deeper levels of IE. InfoXtract engine, provides the core technology for other advanced text processing tools

developed by Cymfony such as text summarization and question answering which will be addressed in the next chapter.

The commercial systems outlined here reveal an enormous application potential of IE technology, and encourage utilization of more advanced NLP techniques in the future. Other US-based companies with similar technology in development include GTE Labs (http://www.gte.com), AT&T Labs (http://www.research.att.com), and MITRE (http://www.mitre.org).

4. BEYOND INFORMATION EXTRACTION

The last decade has witnessed great advances and interest in the area of information extraction based on STP. In the very recent period, new trends in information processing from texts based on lightweight linguistic analysis closely related to IE have emerged.

4.1 Textual Question Answering

Textual Question Answering (Q/A) aims at identifying the answer of a question in large collections of on-line documents, where the questions are formulated in natural language and the answers are presented in form of highlighted piece of text containing the desired information. The current Q/A approaches integrate existing IE and IR technologies [Gaizauskas and Humphreys, 2000]. An IR system treats a question as a query and returns a set of top ranked documents. Knowledge extracted by an IE system from documents may be modeled as a set of entities extracted from text and relations between them and further used for concept-oriented indexing, which facilitates localization of the answer to the stated question. [Srihari and Li, 1999] presented Textract - a Q/A system, based on relatively simple IE techniques using NLP. This system extracts open-ended domain independent general event templates expressing the information like WHO did WHAT (WHOM) WHEN and WHERE (in predicate-argument structures). Such information may refer to argument structures centering around the verb notions and associated information of location and time. The results are stored in a database and used as a basis for question answering, summarization and intelligent browsing. Figure 2-12 shows a simplified general event template corresponding to the joint-venture foundation event template presented in figure 2-1.

$$\begin{bmatrix} \text{PREDICATE}: \textit{merge} \\ \text{ARGUMENT1}: \textit{Siemens AG} \\ \text{ARGUMENT2}: \textit{The General Electric} \\ \text{TIME}: \textit{February } 18, 1997 \\ \text{LOCATION}: \textit{Munich} \end{bmatrix}$$

Figure 2-12. General event template corresponding to joint-venture foundation event

Textract, and other similar systems based on lightweight NLP techniques [Attardi and Burrini, 2000], [Harabagiu et al., 2000] achieved surprising good results in the competition of answering fact-based questions in TREC (Text Retrieval Conference) [Voorhess, 1999].

4.2 Text Classification

The task of *Text Classification* (TC) is assigning one or more pre-defined categories from a closed set of such categories to each document in a collection. Traditional approaches in the area of TC use word-based techniques for fulfilling this task. Intuitively, a word like "joint" would either occur in medical or financial texts, but the typical phrases for these two domains, which contain this word, would have probably a slightly different structure. This observation led to a growing exploration of text categorization methods based on NLP which goes beyond simple stemming. [Riloff and Lorenzen, 1998] presented AutoSlog-TS, an unsupervised system that generates domain specific extraction patterns, which was used for automatic construction of high-precision text categorization system. Autoslog-TS retrieves extraction patterns (with a single slot) representing local linguistic expressions that are slightly more sophisticated than keywords. For instance, for extracting information from sentence in figure 2-1 Autoslog-TS would use following patterns: "[SUBJECT] have merged" and "have merged [DIRECT OBJECT]", where such patterns are not simply extracting adjacent words since extracting information depends on identifying local syntactic constructs (verb and its arguments). AutoSlog-TS takes as input only a collection of pre-classified texts associated with a given domain and combines simple STP and statistical methods for automatic generation of a bag of extraction patterns for TC. This new approach to integrating STP techniques in text classification proved to outperform classification using word-based approaches.

Further, similar unsupervised approaches [Yangarber et al., 2000], using light linguistic analysis were presented for acquisition of lexico-syntactic patterns (syntactic normalization: transformation of clauses into common predicate-argument structure), and extracting scenario-specific terms and

relations between them [Finkelstein-Landau and Morin, 1999], which shows an enormous potential of shallow processing techniques in the field of text mining.

4.3 Text Mining

Text mining (TM) combines the disciplines of data mining, information extraction, information retrieval, text categorization, probabilistic modeling, linear algebra, machine learning, and computational linguistics to discover valid, implicit, previously unknown, and comprehensible knowledge from unstructured textual data [Gotthard et al., 1997]. Obviously, there is an overlap between TM and IE, but in text mining the knowledge to be extracted is not necessarily known in advance. [Rajman, 1997] presents two examples of information that can be automatically extracted from text collections using simple STP methods: probabilistic associations of keywords and prototypical document instances. Association extraction from the keyword sets (e.g., named entities and nominal phrases) allows to satisfy information needs expressed by queries like "find all associations between a set of companies including Siemens and Microsoft and any person". Prototypical document instances may be used as representative of classes of repetitive document structures in the collection of texts and constitute good candidates for a partial synthesis of the information content hidden in a textual base. They can be computed by extracting frequent term sets, which in turn can be extracted from a training corpus by utilizing STP methods. All document parts (e.g., sentences, paragraphs) which instantiate any of the top-scoring frequent term sets flow into the prototypical document. Prototypical documents can be used in the process of generating text summaries discussed in the next section.

Text mining contributes to the discovery of information for business and also to the future of information services by mining large collections of text [Abramowicz and Zurada, 2001]. It will become a central technology to many businesses branches, since companies and enterprises "don't know what they don't know" [Tkach, 1999].

4.4 Text Summarization

The goal of *Text Summarization* (TS) is a compression of a textual document or a text collection into a short text, usually limited to a few hundreds of words, which compactly represents information content of the document or collection. The standard summarization techniques are based on sentence extraction, where clues like sentence location, word frequency or linguistic clues are used for estimating sentence significance. [Sekine and

Nobata, 2002] proposed an approach which integrates IE techniques into summarization. The used five independent scores for estimating the sentence significance, including sentence position, sentence length, tf/idf-based measure (average of the tf/idf scores of all words in the sentence), similarity measure between headline and the sentence, and a measure based on the weights of lexico-syntactic IE patterns which match the sentence, with the assumption that patterns appearing more often in a domain are more important. These scores were then combined by interpolation in order to calculate the total score of a sentence. The integration of the IE-pattern-based measure paid off since the performance of this summarization system was better than that of all other systems participating in the Single Document Summarization Task in Document Understanding Conference (DUC) in 2001.

[Harabagiu and Lăcătuşu, 2002] have presented GISTEXTER, which shows that high-quality multi-document summarization can be achieved by integrating IE techniques. An IE system is used in order to extract the information base needed for creating a summary. Lexico-semantic patterns (e.g., [CASUALTY-EXPRESSION] to [NUMBER] from [DISASTER-WORD]) are matched against the document collection in order to identify topic relevant information. Additionally, a mapping from template slots to the text fragments containing the information that fills the slots is generated. Further, all the entities from the text collection that corefer with the information filling any slot are stored in coreference chains since they claim that in order to be comprehensible, summaries should include sentences or text fragments that contain the antecedents of all anaphoric expressions from relevant text fragments. A multi-document summary is generated incrementally by inspecting the most representative templates, in order to select sentences containing the text fragments mapped from these templates. The importance of the templates is measured as a sum of all frequency counts of all it slots. An additional factor integrated in the importance measure is a preference for templates that have larger number of mapped text fragments traversed by coreference chains. The set of highly ranked sentences for creating the summary is extended by sentences which contain antecedents of the anaphoric expressions appearing in the highly ranked sentences. Since the final summary length is a parameter of GISTEXTER, various additional procedures are deployed for enhancing/compacting the summary. GISTEXTER was one of the best scoring systems in the DUC 2002 evaluation.

5. SUMMARY

Prompt, sound and timely information is an essential factor in competition in business today. We have learned that IE technology can provide dramatic improvements in conversion of the overflow of raw textual information into valuable and structured data which is easier to search in. In particular, IE technology has been successfully used in financial, insurance and legal domain in various real-world applications dealing with processing vast amount of textual data. Interestingly, very good results can be achieved by applying relatively simple and efficient shallow text processing techniques that do not require much linguistic sophistication.

The recent trend towards applying partial linguistic analysis in IE systems does not mean that such partial parsing is adequate for solving all problems dealing with processing huge collections of textual data. For instance, for some extraction tasks it is worthwhile to spend hours rather than minutes of CPU time if this produces better results. Hence, further important research direction will be the integration of shallow and deep text processing such that a DTP might be called for those structures recognized as being of great importance. Further, deep processing could be applied to text fragments which might have some relevance, but could not have been successfully processed by the STP engine. An initial work in this area has been presented in [Crysmann et al., 2002] and [Feiyu and Krieger, 2003].

In this chapter we demonstrated also that STP – core IE technology has been successfully applied in other fields of information technology which are closely related to IE. The diagram in figure 2-13 reflects an enormous application potential of STP in various fields of information technology. STP can be considered as an automated generalized indexing procedure. The degree and amount of structured data a STP component is able to extract plays crucial role for subsequent high-level processing of extracted data. In this way, STP offers distinct possibilities for boosting productivity in workflow management, e-commerce and data warehousing [Abramowicz et al., 2002]. Potentially, solving a wide range of business tasks can be substantially improved by using IE. Therefore, an increased commercial exploitation of IE technology could be observed.

The question of developing text processing technology base that applies to many problems is still being major challenge of current research. In particular, future research in this area will focus on multilinguality, cross-document event tracking, automated learning methods to acquire background knowledge, portability, greater ease of use and stronger integration of semantics.

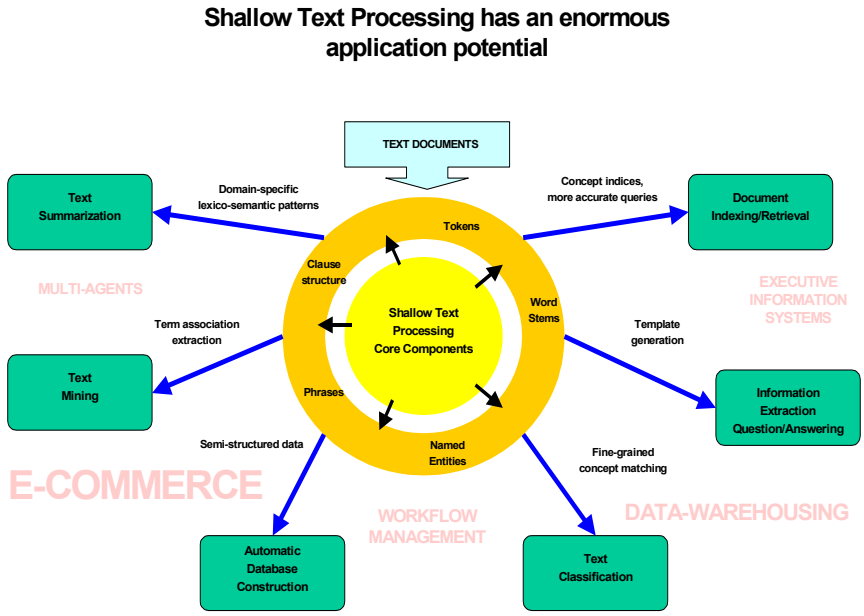

Figure 2-13. Application potential of shallow text processing

REFERENCES

[Abramowicz at. al. 2002] W. Abramowicz, P. Kalczyński, K. Węcel. *Filtering the Web to Feed Data Warehouses*. Springer, ISBN 1852335793, London, 2002.

[Aït-Mokhtar et al., 2002] S. Aït-Mokhtar, J. Chanod, C. Roux. *Robustness beyond shallowness: incremental deep parsing*. In the Journal of Natural Language Engineering, Volume 8 (2/3), pages 121-144, Cambridge University Press, United Kingdom, 2002.

[Andersen et al., 1992] P.M. Andersen, P.J. Hayes, A.K. Heuttner, L.M. Schmandt, I.B. Nirenburg, S.P. Weinstein. *Automatic extraction of facts from press releases to generate news stories*. In Proceedings of ANLP 1992, Trento, Italy, pages 170-177, 1992.

[Aone et al., 1999] C. Aone, L. Halverson, T. Hampton, M. Ramos-Santacruz, T. Hampton. *SRA: Description of the IE2 System used for MUC-7*. Morgan Kaufmann, 1999.

[Aone and Ramos-Santacruz, 2000] C. Aone, M. Ramos-Santacruz. *RESS: A Large-Scale Relation and Event Extraction System*. In the Proceedings of ANLP 2000, Seattle, USA, 2000.

[Appelt and Israel, 1999] D. Appelt and D. Israel. *An Introduction to Information Extraction Technology*. A Tutorial prepared for IJCAI Conference, 1999.

[Becker et al., 2002] M. Becker, W. Drożdżyński, H.U. Krieger, J. Piskorski, U. Schäfer, F. Xu. *SProUT - Shallow Processing with Typed Feature Structures and Unification*. In Proceedings of ICON 2002, Mumbai, India, December, 2002.

[Chanod, 2000] Jean-Pierre Chanod. *Robust Parsing and Beyond*. In: Robustness in Language Technology, G. Van Noord and JC Junqua eds., Kluwer, 2000.

[Chinchor, 1998] – N. A. Chinchor. *Overview of MUC7 /MET-2*. In Proceedings of the Seventh Message Understanding Conference (MUC7), 1998.

[Ciravegna et al., 2000] F. Ciravegna, A. Lavelli, and G. Satta. *Bringing information extraction out of the labs: The Pinocchio environment*. In Proceedings of the 14th European Conference on Artificial Intelligence, Berlin, Germany, 2000.

[Cole et al., 1996] R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, V. Zue. *Survey of the State of the Art in Human Language Technology*. Cambridge University Press ISBN 0-521-59277-1, 1996.

[Costantino et al., 1997] M. Costantino, R.G. Morgan, R. J. Collingham, R. Garigliano. *Natural Language Processing and Information Extraction: Qualitative Analysis of Financial News Articles*. In Proceedings of the Conference on Computational Intelligence for Financial Engineering (CIFEr '97), New York City, March 23-25, 1997.

[Costantino, 1999] M. Costantino. *IE-Expert: Integrating Natural Language Processing and Expert System Techniques For Real-Time Equity Derivatives Trading*. The Journal of Computational Intelligence in Finance, Vol.7, No.2, pp.34-52, March 1999.

[Crysmann et al., 2002] B. Crysmann, A. Frank, B. Kiefer, H. Krieger, S. Müller, G. Neumann, J. Piskorski, U. Schäfer, M. Siegel, H. Uszkoreit, and F. Xu. *An Integrated Architecture for Shallow and Deep Processing*. In Proceedings of ACL-2002, University of Pennsylvania, Philadelphia, July 2002.

[Cunningham, 2002] H. Cunningham. *GATE, a General Architecture for Text Engineering*. In Computing and the Humanities, Vol. 36, pp. 223-254, 2002.

[Finkelstein-Landau and Morin, 1999] M. Finkelstein-Landau, E. Morin. *Extracting Semantic Relationships between Terms: Supervised vs. Unsupervised Methods*. In proceedings of International Workshop on Ontological Engineering on the Global Information Infrastructure, Dagstuhl Castle, Germany, pages 71-80, 1999.

[Gaizauskas and Robertson, 1997] R. Gaizauskas R, A. Robertson. *Coupling Information Retrieval and Information Extraction: A New Text Technology for Gathering Information from the Web*. In Proceedings of RIAO'97, Canada, pages 356-370, 1997.

[Gaizauskas and Humphreys, 2000] R. Gaizauskas and K. Humphreys. *A Combined IR/NLP Approach to Question Answering Against Large Text Collections*. In Proceedings of RIAO 2000, Paris, 2000, pages. 1288-1304.

[Glasgow et al., 1998] B. Glasgow, A. Mandell, D. Binney, L. Ghemri, D. Fisher. *MITA : An Information-Extraction Approach to the Analysis of Free-Form Text in Life Insurance Applications*. AI magazine, 19(1) :59--71. 1998.

[Grishman and Sundheim, 1996] R. Grishman and B. Sundheim. *Message Understanding Conference -- 6: A Brief History*. Proceedings of the 16th International Conference on Computational Linguistics (COLING), pages 466--471, Kopenhagen, Denmark, 1996.

[Harabagiu et al., 2000] S. Harabagiu, M. Pasca, S. Maiorano. *Experiments with open-domain textual question answering*. In Proceedings of the COLING-2000. Association for Computational Linguistics, Morgan Kaufmann, 2000.

[Harabagiu and Lăcătuşu, 2002] S. Harabagiu and F. Lăcătuşu. *Generating Single and Multi-Documat Summaries with GISTextrer*. Workshop on Text Summarization in conjunction with the ACL 2002, Philadelphia, USA, 2002.

[Hobbs et al., 1997] J. Hobbs, D. Appelt, J. Bear, D. Israel, M. Kameyama, M. Stickel, M. Tyson. *FASTUS - A cascaded Finite-State Transducer for Extracting Information from Natural Language Text*. Chapter 13 in [Roche and Schabes, 97], 1997.

[Humphreys et al., 1998] K. Humphreys, R. Gaizauskas, S. Azzam, C. Huyck, B. Mitchell, H. Cunningham, Y. Wilks. University of Sheffield: *Description of the LaSIE-II System as used for MUC-7*. In the MUC-7 Proceedings, 1998.

[Jackson et al., 1998] P. Jackson, K. Al-Kofahi, C. Kreilick and B. Grom. *Information extraction from case law and retrieval of prior cases by partial parsing and query generation*. In Proceedings of the ACM 7th International Conference on Information and Knowledge Management, pages 60-67, Washington United States, 1998.

[Jacobs et al., 1990] P. Jacobs and L. Rau. *SCISOR: extracting information from online news*. Communications of the ACM, 33, 11, pages 88-97, 1990.

[Kehler, 1998] A Kehler. *Learning Embedded Discourse Mechanisms for Information Extraction*. In Proceedings of the AAAI Spring Symposium on Applying Machine Learning to Discourse Processing, Stanford, CA, March 1998.

[Lytinen and Gershman, 1986] S. Llytinen and A. Gershman. *ATRANS: Automatic Processing of Money Transfer Messages*. In Proceedings of the 5th National Conference of the American Association for Artificial Intelligence, IEEE Computer Society Press, 1993.

[Makhoul et al., 1999] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. *Performance measures for information extraction*. In Proceedings of DARPA Broadcast News Workshop, Herndon, VA, Feb. 1999.

[Manning and Schütze, 1999] C. Manning, H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachuusetts, London, England, 1999.

[Maynard et al., 2002] D. Maynard, V. Tablan, H. Cunningham, C. Ursu, H. Saggion, K. Bontcheva, and Y. Wilks. *Architectural elements of language engineering robustness*. In the Journal of Natural Language Engineering, Volume 8 (2/3), pages 257-274, Cambridge University Press, United Kingdom, 2002.

[Mihalcea and Moldovan, 2001] R. Mihalcea and D. Moldovan. *Document Indexing Using Named Entities*. In Studies in Informatics and Control Journal, Vol. 10, Number 1, March 2001.

[Mohri and Nederhof, 2001] M. Mohri and M. Nederhof. *Regular Approximation of Context-Free Grammars through Transformations*. In J.C. Junqua and Gertjan van Noord, editors, *Robustness in Language and Speech Technology*, pages 153-163. Kluwer Academic Publishers, The Netherlands, 2001.

[Neumann and Piskorski, 2002] G. Neumann and J. Piskorski. *A Shallow Text Processing Core Engine*. In the Journal of Computational Intelligence, August, 2002, vol. 18, no. 3, pp. 451-476(26) Blackwell Publishers Inc, Boston, USA and Oxford, UK, 2002.

[Petasis et al., 2002] G. Petasis, V. Karkaletsis, G. Paliouras, I. Androutsopoulus, and C. Spyropoulus. *Ellogon: A New Text Engineering Platform*. In the Proceedings of LREC 2002, Las Palmas, Gran Canaria, Spain, 2002.

[Piskorski and Skut, 2000], J. Piskorski, W. Skut. *Intelligent Information Extraction*. In the Proceedings of Business Information Systems 2000, Poznan, Poland, 2000.

[Piskorski and Neumann, 2000] J. Piskorski, G. Neumann. *An Intelligent Text Extraction and Navigation System*. In the Proceedings of RIAO 2000, Paris, France, 2000.

[Rajman, 1997] M. Rajman. *Text Mining, knowledge extraction from unstructured textual data*. In Proceeding of EUROSTAT Conference, Frankfurt, Germany, 1997.

[Riloff and Lorenzen, 1998] E. Riloff, J. Lorenzen. *Extraction-based text categorization: Generating domain-specific role relationships automatically*. In Strzalkowski, T., ed., Natural Language Information Retrieval, Kluwer Academic Publishers, 1998.

[SAIC, 1998] SAIC, editor. *Seventh Message Understanding Conference (MUC-7).* http://www.muc.saic.com, 1998.

[Sang, 2002] Tjong Kim Sang Erik F. *Introduction to the CoNLL-2002 shared task: language-independent named entity recognition*. In Proceedings of CoNLL-2002, Roth Dan (editor), Taipei, p. 155-158, 2002.

[Sekine and Nobata, 2002] S. Sekine, CH. Nobata. *Sentence Extraction with Information Extraction technique*. Workshop on Text Summarization in conjunction with the ACL 2002, Philadelphia, USA, 2002.

[Srihari and Li, 1999] R. Srihari and W. Li. *Information extraction supported question answering*. In Proceedings of the Eighth Text Retrieval Conference (TREC-8)., 1999.

[Tkach, 1999] D. Tkach. *The pillars of knowledge management*. In Knowledge Management 2(3), page 47, 1999.

[Voorhess and Tice, 1999] E. Voorhess and D. Tice. *The TREC-8 Question Answering Track Evaluation*. National Institute of Standards and Technology, Gaithersbugh, 1999.

[Yangarber et al., 2000] R. Yangarber, R. Grishman, P. Tapanainen and S. Huttunen. *Unsupervised Discovery of Scenario-Level Patterns for Information Extraction*. In Proceedings of ANLP-NAACL 2000, Seattle, USA, 2000.

[Xu and Krieger, 2003] F. Xu and H.-U. Krieger, Extraction of Domain Specific Events and Relations via a Combination of Shallow and Deep NLP. Research Report, DFKI, Saarbrücken, Germany, to appear in 2003.