Student Thesis

Information Quality Management in Information Extraction: A Survey

submitted by

Nikolas Jansen

born August 6, 1982 in Bocholt

Technische Universität Dresden

Faculty of Computer Science
Institute of Systems Architecture
Chair of Computer Networks

Responsible professor: Prof. Dr. rer. nat. habil. Dr. h. c. A. Schill

Supervisor: Dipl.-Medien-Inf. Klemens Muthmann
External supervisor: MSc. Wojciech Barczynski (SAP Research)

Submitted January 14, 2011

Technische Universität Dresden and SAP Research CEC Dresden

TECHNISCHE UNIVERSITÄT DRESDEN

Fakultät Informatik Institut für Systemarchitektur, Lehrstuhl Rechnernetze

Task for the Belegarbeit

Name, Surname: Jansen, Nikolas

Subject of Studies: Medieninformatik ***Matr. Nr.:*** 2 9 2 8 4 0 2

Topic: ***„Information Quality Management in Information Extraction: A Survey"***

Description:

The benefits of analytics over unstructured text is widely recognized both in research and industry, in many areas, such as marketing (extracting user's opinions) and public relations (how customers position a product). However to realize such analytics, which provides sound results, we need to address quality issues in text processing – Information Extraction. Information Extraction achieves a level of accuracy between 90-98% only in identifying simple entities (e.g., person) and just 50-60% for complex entities. Within the UIM Team at SAP Research we investigate methods that help to address this problem. A key element of the foreseen solution is a method that would allow an IE developer to describe quality aspects of Information Extraction methods. This information could than be used as a base for assessment of analytic results.

The task of this work is to survey the research of information quality management for information extraction. The central focus will lie on classifying the research activities along the extraction process. Further will this work evaluate the discussed approaches according to standard information quality assessment methodologies.

Main emphasis:

- fundamentals of information extraction and information quality
- summary and comparison of methodologies for information quality management in information extraction
- evaluation of information quality management approaches
- outlook and conclusions

Supervisor: Dipl.-Medien-Inf. Klemens Muthmann

External supervisor MSc. Wojciech Barczynski

Responsible professor: Prof. Dr. rer. nat. habil. Dr. h. c. Alexander Schill

Institute: Systemarchitektur

Start: 15.07.10

End: 14.01.11

Signature of responsible professor

Confirmation

I confirm that I independently prepared the thesis and that I used only the references and auxiliary means indicated in the thesis.

Ich erkläre, dass ich die vorliegende Arbeit eigenständig und ausschließlich unter Verwendung der im Quellenverzeichnis aufgeführten Literatur- und sonstigen Informationsquellen angefertigt habe.

Dresden, January 14, 2011

Abstract

Information is a valuable asset for organizations and individuals. Huge amounts of information are buried in unstructured text. Information extraction research provides appropriate mechanisms to retrieve this knowledge. However the quality of extracted information is not sufficient for further automated analytics. Recent research activities have tackled those quality issues. In this work we give a comprehensive review of information quality research applied on information extraction. First, we provide a structured overview on information quality research in general. We then describe the concept of information extraction to introduce the specifics this area will bring to information quality management. The main section of this work presents and categorizes recent research work tackling quality issues in information extraction systems. For the survey, we structure the investigated approaches along the components involved in the process from unstructured text to exploitable knowledge. We conclude this work with an outlook of what is left to be done to manage information quality for information extraction results.

Contents

1 Introduction

This chapter illustrates the area of interest, introduces the thesis within its research context, motivates the topic, and describes the structure of the survey.

1.1 The Overall Picture

Information is one of the most valuable assets in private and public organizations. Organizations search for and evaluate information in order to make important decisions and plan future business approaches. Although the organizational decision making process is a complex, and messy process, there is no doubt that wise decisions need complete and correct information as a foundation. As Redman [Red92] stated "If information technologies are the engines of the Information Age, then data and information are the fuels."

In the 90's a process called Business Intelligence was introduced in order to satisfy enterprise executives' request for efficient and effective analysis of the vast amount of enterprise data. This allowed them to better understand the situation of their business and improve their decision process. Over the last two decades, the techniques and tools to perform Business Intelligence have evolved. In fact, Business Intelligence has changed the approach to business management from both a technological and organizational point of view. The foundation for this analytical process is processed structured data used in data warehouses. However, there is another important type of data that is not incorporated into analytics through standard Business Intelligence processes, textual unstructured data. This type of data includes like emails, contracts, management reports, customer relationships, memos and even law suits. According to [Gro08], about 80% of information within a company is currently unstructured. In it's raw form the data has limited value since analytics will not go further than keyword search. Hence a huge amount of information is and keeps being buried in unstructured textual data.

Consequently, over the past two decades significant effort has been focused on the problem of extracting structured information from such data. The extracted

information is then exploited in search, browsing, querying and mining. However even the most advanced and up-to-date Information Extraction (IE) systems are not perfect. Due to the noisy nature of the extraction process the output is not necessarily of highest quality. In numbers IE achieves an accuracy of 90-98% in identifying simple entities and just 50-60% accuracy for relationships between entities [Fel06]. According to [PM06] accuracy values are close to 90% for individual fields and only 70-80% for the extraction of multi-field stuctured entities like author names, title and journal names from citations. In general it is safe to say that it is almost impossible to quarantee a degree of accuracy in a real world setup [Sar08].

Information extracted from unstructured textual data is not limited to organizational usage. As unstructured data on the World Wide Web (WWW) exploded in recent years, individuals benefit from that fact also. This information exists even for everyday activities like consulting reviews before buying a printer or booking a hotel. Other applications of IE are Community Information Management and Semantic Search [DRV06].

As Business Intelligence and other forms of analytics over unstructured data become more and more popular, the issue of Information Extraction Quality needs to be addressed. Therefore, information quality management is one of the biggest challenges in IE research.

1.2 Contribution

In this survey we will investigate the most promising approaches to enhance information quality in IE results.

First of all, we will take a look at recent Information Quality (IQ) research to find a basic understanding of how we can identify, classify, measure and manage quality of information. After having identified possible quality problems that can arise with information, we will define and classify quality dimensions to handle the aforementioned problems in a structured manner. IQ assessment methodologies are categorized as either into subjective or objective assessment. Finally, we will take a look at IQ in information integration systems.

Next, we dedicate a Chapter to provide background knowledge about IE. We will explain the process of entity and relationship extraction and elaborate on the various steps in an IE pipeline like lexical and syntactical analysis.

With the ground work laid in the preceding chapters, we can start the main contribution of this work, a survey on result quality management approaches for IE

systems. We will take a close look at these approaches and classify them into three different areas to tackle quality issues. The first area is document retrieval. The second area deals with quality concerns during the information extraction process. The last area is to consider the output quality of strategies of joining extraction programs to complex IE systems.

We do not claim this review to be exhaustive or representative of all the research in this field, but we do believe it gives a good introduction to two distinct research areas and a feel for the breadth and depth of work combining effort in those.

1.3 Outline

The remainder of this work is organized as follows.

- Chapter 2 presents a review on information quality research. We structure and categorize approaches presented over the last two decades.
- In Chapter 3 we will introduce IE. All necessary preliminaries to understand the process of information extraction will be provided.
- Chapter 4 surveys recent research activities dealing with information quality management in IE systems. We will structure presented works and provide a classification.
- Chapter 5 sums up results of this work and provides an outlook.

2 Information Quality

This chapter provides an introduction to Information Quality (IQ) research. First we present the extensive attention IQ has received over the last three decades. In Section 2.2 we list and categorize quality problems that can arise when dealing with data and information. Further we identify, define and classify IQ dimensions that are linked to one or more IQ problems. Section 2.4 presents IQ assessment methodologies. To conclude this Chapter, we take a look at IQ concerns in information integration systems.

2.1 Introduction

IQ has become a critical concern of private and public organizations. The growth of information repositories and the direct access to information from various sources has increased the awareness of the quality of information in organizations. Low information quality is one of the reasons why business initiatives fail and huge losses are generated. The repercussions of flawed or incomplete information are not just costly and pervasive, they can even be disastrous.

Fischer and Kingma investigate in [FK01] the explosion of the space shuttle *Challenger* and the shooting down of an Iranian Airbus by the USS *Vincennes* with regard to the poor quality of data. Flawed information was used in the decision making process that led to these disastrous events. Although other factors like psychology, sociology, communication and culture cannot be omitted, Fischer and Kingma state that it is difficult to believe that proper decisions could have been made with all the examples of poor data quality that they uncovered.

Eppler and Helfert take it a step further and provide in [EH04] an analysis of the cost that poor IQ causes. Besides the cost of having low quality information itself, the process of detecting and repairing it generate huge losses. Redman [Red92] examines this issue more concretely and states that the cost of poor IQ is at least two percent of revenue, which does not include the immeasurable loss of reputation and customer satisfaction.

Considering the examples and loss estimations, it is safe to say that low IQ is one of the most pressing problems for decision makers. This fact reveals the need for methodologies to manage IQ. Whether it means just measuring or also enhancing the quality of information, many projects have proposed methods to reach these goals.

Before we move on to review research activities in this field we need a common understanding of terminology. The term *information* is so intertwined with *data* that some researchers, such as in Wang et al. [WRY01], viewed them as synonyms and use data and information interchangeably. From an information professional's perspective, the distinction between data and information can be as follows: data or data elements, are specific entries in a database; information is the combination of different pieces of data to produce new quantities that provide insight into the processes producing data [GH96]. Madnick et al. [MWLZ09] mention the tendency to use the term data quality when referring to technical issues and to use the term information quality for non-technical issues. In this work, we use the term information quality (IQ) for the full range of issues.

In quality literature, the concept of *fitness for use* is widely adopted [WS96]. This definition emphasizes the importance of taking the viewpoint of a consumer. However, information consumers are not very capable of finding errors in information and altering the way they use the information [KGD97]. Information systems are therefore required to incorporate mechanisms to ensure high quality information before presenting to the consumer. This fact leads to an IQ concept which is defined from the data perspective [KSW02] and therefore more appropriate for the context of this work.

Information Quality (IQ): Information that meets specifications and requirements

Various studies confirmed that IQ is a multi-dimensional concept [WS96] [WW96].

IQ Dimension: IQ attribute that represents a single aspect or construct of IQ

With the definition of IQ and the concept of IQ dimension we have the necessary tools to investigate quality approaches in IE systems. Section 2.2 lists and categorizes IQ problems so that Section 2.3 can define interesting IQ dimensions derived from the aforementioned problems.

2.2 Information Quality Problems

Several problems with information can affect IQ and therefore the fitness for use by the information consumer or the information does not meet certain requirements. Hence these problems affect IQ. A number of works identify and describe information quality problems. This matter is essentially studied in two research communities: database and management. The database community concentrates just on the technical point of view while the management community also incorporates problems from the consumer point of view. Ge et al. [GHOH+07] classify IQ problems considered from the data and the user perspective. Two main works representing each perspective are reviewed. So additionally from the database perspective the work from Oliveira et al. [ORH05] studies problems with data values or instances and represents a comprehensive list of IQ problems. Huang et al. [HLW99] identify a complete list of IQ problems from the consumer perspective. Ge et al. [GHOH+07] also divide each perspective into context-independent and context-dependent which will finally lead to a two-by-two conceptual classification of IQ problems illustrated in Figure 2.1.

	Data Perspective	**User Perspective**
Context-independent	Problems in the database	Problems in processing information
Context-dependent	Violation of business specifications	Problems with indented use by user

Figure 2.1: Description of classification

Figure 2.2 categorizes a subset of IQ problems into the conceptual model.

2.3 Information Quality Dimensions

Once we provide a classification of possible problems we discuss IQ dimensions. These are defined as a set of attributes representing a single aspect of IQ. Most studies describe IQ as a multi-dimensional concept, but the literature names different sets of dimensions.

According to Wang and Strong [WRY01] there are three approaches used in the literature to define IQ dimensions, *intuitive*, *theoretical* and *empirical*.

	Data Perspective	User Perspective
Context-independent	Missing data Duplicate data Incorrect value Outdated data	Inaccessible Hardly retrievable Difficult to aggregate
Context-dependent	Violation of domain constraints Violation of organization's business rules	Doubtful credibility Irrelevant Incomplete

Figure 2.2: Information quality problems [GHOH+07]

Intuitive approach: Most studies fall into this category in which IQ dimensions derive from the researcher's experience and demands of particular cases.

Theoretical approach: IQ dimensions are identified based on data deficiencies that can occur during the data manufacturing process. No information consumer needs are considered. Wand and Wang [WW96] use an ontological approach in which IQ dimensions are derived by examining the inconsistencies between a real world system and an information system.

These two approaches focus on data as a product in terms of development characteristics instead of use characteristics.

Empirical approach: This approach captures the IQ attributes that are important to the information consumer.

Wang and Strong [WRY01] define an empirical approach and list 15 dimensions. Bovee et al. [BSM03] introduce four simple main attributes to IQ: accessibility, interpretability, relevance and integrity with underlying criteria. 16 information quality dimensions are defined by Kahn et al. [KSW02]. Huang et al. [HLW99] combine the system and user perspectives and develop four IQ dimension categories: intrinsic, contextual, representational and accessibility.

For the purpose of evaluating quality efforts in Information Extraction we investigate four IQ dimensions proposed in various work, *accuracy*, *completeness*, *timeliness* and *uniqueness*.

Accuracy The dimensions *accuracy* is a qualitative assessment of freedom from error, with a high assessment corresponding to a small error [Cyk96]. Bovee et al. [BSM03] meant by *accuracy* being true or error free with respect to some known, designated, or measured value. Manning et al. define *accuracy* as the percentage of correct pieces of information in regard to the total amount of pieces. In general *accuracy* is the *free-of-error* measure.

Completeness is the degree to which values are present in the attributes that require them [Cyk96]. It refers to having all required parts of an entity's information present. If an attribute is mandatory, a non-null value is expected. Once a null value appears in a required attribute, the information is considered incomplete. Huang et al. [HLW99] discussed incomplete data resulting from operational problems, like data entry errors, and design failure, such as not including desired attributes in a database. Kahn et al. [KSW02] add another aspect to *completeness* and state it is the extent to which information is not missing and is of sufficient breadth and depth for the task at hand.

Timeliness represents the degree to which specified data values are up to date [Cyk96]. Chen et al. [CZW98] present a time-oriented view on IQ. Response and turnaround time for a query or a degree of freshness of the result are mentioned. For later use, all those measures are referred to as *timeliness*.

Uniqueness is the state of being the only one of its kind or being without an equal or equivalent [Cyk96].

Dependency of IQ Dimension

A number of works on IQ point out that certain IQ dimensions correlate to each other. Redman [Red97] mentions that *accuracy* is influenced by *timeliness*. Olsen [Ols02] and Cappiello [CFP03] imply a relationship between *accuracy* and *completeness*. Amicis [ABB06] analyzes a correlation between syntactic *accuracy* and *timeliness* and also between *timeliness* and *completeness*.

Correlations of IQ dimensions can be either negative or positive. A positive relationship would be *timeliness* and *accuracy*. Improvement of quality in one dimension affects the other dimension positively. *Accuracy* and *completeness* are negatively correlated. We will investigate this trade-off closer when evaluating IE systems in Section 4.2.1.

2.4 Information Quality Assessment

Pipino et al. [PLW02] categorize IQ assessment into objective and subjective assessment. Objective IQ assessment reveals the IQ problems in data sets while subjective assessment reflects the needs and experiences of information consumers.

Objective IQ assessment is to measure the extent to which information conforms to quality specifications and references. Overall objective IQ assessment can be considered as the procedure of comparing current data value with optimal data value. This assessment method fits well when evaluating IE systems.

Subjective IQ assessment measures to what extend the information has fitness for use by consumers. Consumers assess the quality of information according to their demands and expectations. Hence subjective IQ assessment focuses on the discrepancy between current quality of information and a user's expectation.

For objective IQ assessment, software is used to automatically measure the data and the derived information by a set of quality rules. Subjective IQ assessment always uses survey to measure the contextual information by data consumers.

IQ assessment for IE systems is going to be objective. Software measures discrepancies of IE results to so-called gold-standard values.

2.5 IQ in Information Integration Systems

IQ in information integration systems is still an active research area after more than two decades of intensive study. The integration process on its own affects the quality of information. Concepts from the area of databases and data integration systems like entity resolution, record linkage, provenance and uncertainty are interesting from the IQ point of view. Provenance information like knowledge about source and process of information can help the consumer to assess quality of information integration results. A probabilistic view on information opens up a new perspective to IQ. *Accuracy* for example could have a probabilistic value representing uncertainty about the freedom from error. Provenance and uncertainty management are investigated in more detail later in this work.

2.6 Summary

In this Chapter we presented an introduction to Information Quality research. Definitions of IQ from the data and consumer perspective are presented. Section 2.2

categorizes IQ problems into a two-by-two conceptual model. In Section 2.3 we define four IQ dimensions that are appropriate for later use on IQ in Information Extraction. These dimensions are *accuracy*, *completeness*, *timeliness* and *uniqueness* are considered. A distinction between objective and subjective IQ assessments is presented in Section 2.4. The last Section introduces concepts from data integration systems to the field of IQ. Concepts like provenance and uncertainty management are important in efforts of improving IQ in Information Extraction.

3 Information Extraction

This chapter introduces the research field of *Information Extraction*. After a short introduction in Section 3.1 including a brief history and application examples, Section 3.2 provides an overview of the IE pipeline and presents the individual components. Section 3.3 wraps this Chapter up with a short look at related work.

3.1 Introduction

Information Extraction (IE) refers to the creation of structured representation of selected information drawn from unstructured content. This goal is achieved by automated extraction of structured information such as entities and semantic relationships between entities from sources like natural language text. IE provides mechanisms to capture knowledge from web pages, research papers, management reports, community forums, email, etc then represent it in a fixed format and unambiguous way. The information locked in natural language is transformed into a structured, normalized database form. This embodiment opens up new possibilities to organize, analyze and query the vast amount of information that is buried in unstructured sources. Before IE was introduced, the search for information was limited to either keyword-based document searches or manually acquired knowledge by domain experts. Both approaches have clear shortcomings. They are either not capable of finding semantics like relationships of entities, or they are way too cost intensive. Besides improving search, IE is useful in a diverse set of applications which are covered in Section 3.1.3.

3.1.1 History of Information Extraction

IE has received intense attention in multiple research communities. In the 80s and 90s the *Artificial Intelligence* community dedicated much work to get an accurate representation of the content of an entire text [Zec97]. Overlapping work from *Artificial Intelligence* and linguistic research communities has bred the field of *Natural Language Processing*. This refers to computer systems that analyze, attempt to

understand, or produce one or more human languages [All03]. As a next step for IE the term *Named Entity* was introduced at the *Sixth Message Understanding Conference* (*MUC*) [GS96]. Although a related term *fact extraction* was in use as far back as the 60s, the field of IE grew out of the *MUC* competitions with the introduction of *Named Entity Recognition*. At that time, the *MUC* was focusing on tasks where structured information about company and defense related activities were extracted from unstructured text (e.g. newspaper articles). In defining the task, people noticed that it is essential to recognize information units like names, including person, organization and location names, and numeric expressions including time, date, money and percent expressions. Identifying references to these entities in text was recognized as one of the important sub-tasks of IE and was called *Named Entity Recognition*. In the early 90s Rau [Rau91] approached the technology of *Named Entity Recognition* for the first time by describing a system for *Extracting Company Names from Text*.

Within the last decade the *Automatic Content Extraction* (*ACE*) [DMP+04] program has contributed largely to IE. The objective of the *ACE* program exceeds the simple extraction of *Named Entity* by considering also relationships between the entities and so called events in which entities participate. Additional advances of the *ACE* to the preceding *MUC* is the identification of the entity itself (ACE) and not just the word representing the *Named Entity* (MCU) from unstructured text. Between 2000 and 2004 following tasks have been defined as part of the *ACE* program [DMP+04]:

Recognition of entities, not just names (2000-2001) According to the *ACE* Entity Detection and Tracking (EDT) task, all occurrences of an entity, whether a name, a description, or a pronoun need to be found. Based on a practical coreference resolution on entity level, all mention of an entity should be collected into equivalence classes.

Recognition of relations (2002-2003) In the Relation Detection and Characterization (RDC) task relations that occur between pairs of entities can be characterized as one of the following five general types. *Role* (role a person plays,...), *Part* (part-whole, part-of,...), *At* (location), *Near* (relative location), *Social* (relative, associate,...).

Event extraction (2004) In the Event Detection and Characterization (EDC) task annotators identify and characterize five types of events in which entities par-

ticipate. The five targeted types of events include Interaction, Movement, Transfer, Creation and Destruction.

With the explosion of the amount of unstructured data in the *World Wide Web* and the need of querying this information in a much richer form a more versatile community of researchers focused their activities on IE including groups from machine learning, information retrieval, database, web, and document analysis.

3.1.2 Information Retrieval and Information Extraction

Information Retrieval (IR) is the management of text on the document level as opposed to IE which is text management at the content level. Hence, IR identifies relevant documents from a large collections that most closely conform to the restrictions of a query, while IE extracts relevant information from documents and produces structured data ready for post processing. IR is a much more mature science than IE and has been around as long as databases of documents have existed. These two fields have different aims but are also complementary and can be used in combination to provide powerful tools for text processing. In Section 4.1 we will introduce document retrieval strategies as one component in the unstructured data management system that has a share in the overall result quality. Also evaluation metrics for IE, Section 4.2.1, have their origin in IR, Section 4.1.2.

3.1.3 Application

The structured representation of information from text is applied in many different areas. Three example applications are mentioned here:

Semantic Search The aim of semantic search is to improve traditional search results (based on information retrieval technology) by providing high quality answers. IE enables the user to send structured search queries involving entities and their relationship to a semantic search engine. The availability of structured information about unstructured text in a machine understandable form offers a new way of answering those queries.

Personal/Community Information Management [DRC+06] [Jon07] Information management systems organize data in a structured inter-linked form. Whether it is personal data like documents, emails and contacts or data that is produced and consumed by communities, information management systems provide user

services such as browsing, keyword search and structured querying. The automatic extraction of structured entities and the relationships between entities significantly enriches the aforementioned user services.

Business Intelligence over unstructured Data Business Intelligence on structured enterprise data became popular in the 90's and aims to support wise business control and planning. But the days of just analyzing structured data and information from operational silos are over since organizations need to aggregate information across different departments and business function. Information integration from all possible sources plays a crucial role in modern business intelligence. A large amount of cooperate information is buried in unstructured documents, like contracts, consultancy reports, product documentation, slides, tables, consumer surveys, emails and memos as well as intranet content and even instant messaging. Also internal wikis, technical blogs and various community portals contain valuable knowledge. According to a Butler Report on *Document Records Management* [Gro08], about 80% of the information inside companies is currently unstructured. This information needs to be extracted and analyzed to incorporate it into the operative and strategical descision making process.

3.2 Information Extraction System

IE systems are component-based software programs that are able to identify particular types of entities and relationships between entities in natural language text for storage and retrieval in a structured database. Figure 3.1 illustrates the main components of an extraction system according to [AS06]. This section introduces the individual components and their functions.

3.2.1 Additional Input

Structured and semi-structured text

In order to improve extraction accuracy existing databases of structured information like entities and relationships can be incorporated in the extraction process. Some extraction operators make use of dictionaries in a structured and normalized form as input. This type of input is nevertheless just to assist the extraction of information from other types of input. Semi-structured or labeled text also provides valuable

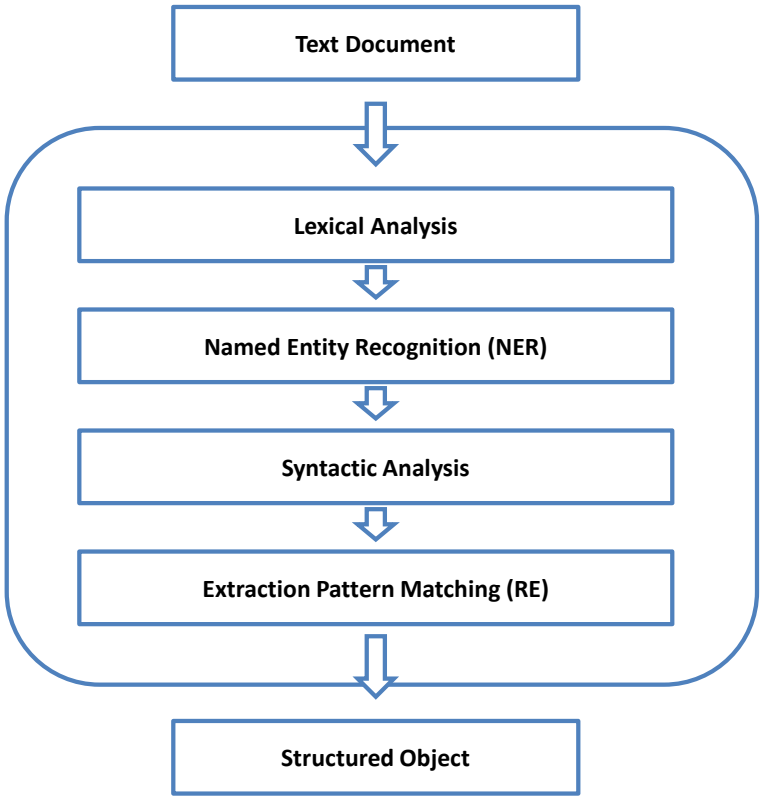

Figure 3.1: Information Extraction Pipeline [AS06]

information to improve the extraction result. Contextual information about entities is contained in labeled text and therefore even more exploitable for extraction systems. IE systems exploit this information to learn extraction patterns via machine learning techniques. Labeling unstructured text involves substantial effort but contextual information about an entity is very valuable.

Unstructured text

When dealing with formatted text, it is important to identify local regions of content like paragraphs, lists or table structures. Web pages are unstructured formatted text documents and can be pre-processed by so-called wrappers that bring the text in regular form.

3.2.2 Lexical Analysis

Extracting information from natural language text requires lexical analysis as a fundamental preprocessing step. Linguistic information over text is inevitable for later stages in the IE process. Tokenizer and *Part-of-Speech* tagger are components of lexical analysis.

Tokenizer

The tokenizer analyzes sentences and splits the entire natural language text into very simple tokens. Boundaries of sentences in the document are identified. Tokens are either punctuation marks or words also including numbers. Uppercase and lowercase text can be distinguished.

Part-of-Speech Tagger

The *Part-of-Speech* tagger annotates each token with a grammatical category. Grammatical categories include noun, verb, adjective, adverb, article, conjunction and pronoun [Sar08]. An example sentence with *Part-of-Speech* tags from the *Penn Treebank* tag set [MMS93] could look like:

> From/*IN* the/*DT* beginning/*NN* it/*PRP* took/*VBD* a/*DT* man/*NN* with/*IN* extraordinary/*JJ* qualities/*NNS* to/*TO* succeed/*VB* in/*IN* Mexico/*NNP*

where:

- *NNS* is a *Noun,plural*
- *PRP* is an *Adverb, comparative*
- *JJ* is an *Adjective*
- *VP* is a *Verb, base form*

Decomposed text like that is then processed by a *Named Entity Tagger*.

3.2.3 Named Entity Recognition

The tokenized text documents are fed to a *Named Entity Tagger*. As mentioned in Section 3.1.1 the term *Named Entity* was coined by the *MUC* and refers to noun phrases (type of token) within a text document like Person, Location, and Organization. Numeric values like phone numbers and dates are also *Named Entities*.

These entities are potential values of attributes in relations which are covered in the next section. During the *ACE* competition more than 100 specific entity types are proposed to identify mentions of entities in text.

Two main paradigms are known to annotate unstructured text by identifying mentions of named entities, rule-based and statistical methods.

Rule-based Methods

Rule-based IE systems are appropriate for many real-life extraction tasks. The principle is a collection of rules of the general form: "Contextual Pattern ⇒ Action" that are deployed on the tokenized text. Policies are defined that control the firing of the rules. Contextual patterns are basically regular expressions defined over properties of tokens in the document and information about the context in which the token appears. Properties of tokens can be [Sar08]:

- The string representing the token
- The *Part-of-Speech* tag of the token
- The Orthography type of the token

IE systems usually rely on extraction rules specifically made for a particular document collection. It is difficult is to reuse or adopt these rules to new document collections. Rule-based extraction systems are either automatically trainable where rules are learned from annotated corpora or interaction with the user or they can be knowledge-engineered in which rules are hand crafted by domain experts [App99]. A knowledge-engineered system requires significant amount of manual labor to create the rules but are usually much more precise. Therefore hand crafted rules are used in closed domain situations with high result quality demands. Automatically trained systems produce much more robust rules and are well suited for IE in open domains with unstructured text that contains a lot of noise. These systems can capture more complex patterns that are hard to encode with hand-crafted rules. Construction of training data is labor intensive and can be problematic in terms of quantity of appropriate documents. Section 4.1.4 introduces optimizations to the training process called *Active Learning*.

Statistical Methods

Entity extraction with statistical methods labels various parts of the tokenized unstructured text. Each token is assigned an entity label identifying entities as con-

secutive tokens with the same entity label. Another approach is to use text chunks instead of using token as labeling unit. The later method works well in well-formed natural language sentences while the first method is appropriate for extraction addresses. Basically, statistical entity extraction makes a decision for each token or group of token based on a weighted sum of predicate firings. Examples of well-known models for classification of tokens are Support Vector Machines, Hidden Markov Models [FM99] [SCR03] and Conditional Random Fields [LMP01]. The last method is considered to outperform all earlier proposed methods for sequence labeling [Sar08].

3.2.4 Syntactic Analysis

Syntactic Analysis or parsing is the process of determining the grammatical structure of text composed of tokens. The structure of sentences is identified and words are grouped into noun phrases, prepositional phrases and verb phrases. Examples for noun and verb phrases appear below [App99]:

Noun phrase: Seven solemn Sicilian sailors

Verb phrase: had been solemnly sailing

Parsing text will lead to a parse tree that groups words into syntactic phrases. This is an important step towards relationship extraction. A parsing tree contains significantly more information than just *Part-of-Speech* tags to understand relationships between the entities in a sentence. A comprehensive survey on algorithms for document structure analysis is provided by Mao et al. [MRK03]. After syntactic analysis, the text is ready to be handed over to the relationship extraction component.

3.2.5 Extraction Pattern Matching

A relationship extraction component processes the document after the named entity tagger has annotated the text properly and syntactical analysis has been performed to build the parsing tree. Relationships are pre-defined constructs associating two or more entities with each other. Examples are "is-headquarter-of" between an organization and a location, "is-CEO-of" between a person and an organization, "has-phonenumber" between a person and phone number and "is-price-of" between a product name and a currency amount. Extraction patterns to identify those

relationships are either manually constructed [YG98], automatically learned [CS99] or composed by combining the two methods.

3.2.6 Output

The foremost expected output of IE systems is a set of relational tables representing the instances of the extracted relationships. These tables are then used to run analytical queries against them or data mining algorithms to exploit the knowledge in the extracted information. The original unstructured text will not be considered in further processing.

The second type of output is the original unstructured text itself. The document exists then in annotated form where all mentions of the structured information is identified and attached to the document.

The difference is basically either to find all mentions in a specific text or to find instances in a document collection to populate a specific relational table.

3.3 Related Work

IE has been under intense scrutiny for the last two decades. Many introductions to IE have been published. Early works that gave a comprehensive overview of the field of IE how we consider it today are [CL96] [Gri97] both around the time the *MUC* competition ended. IE tutorials have been held at conferences like ACM SIGKDD (Knowledge Discovery and Data Mining) [CM03] [AS06], ACM SIGMOD (Management of Data) [DRV06] and the ICML (International Conference on Machine Learning) [Fel06]. Journals like the AMC Queue have featured articles [McC05]. The most recent and extensive survey of the field of IE is [Sar08]. Specialized surveys like [CKGS06] are also available.

3.4 Summary

In this Chapter we provide an introduction to Information Extraction. The origins of this field of research are presented with the *Message Understanding Conference* and the *Automatic Content Extraction*. The extraction system pipeline with components like *Lexical Analysis*, *Named Entity Recognition*, *Syntactic Analysis* and *Extraction Pattern Matching*. Rule-based and statistical methods for IE are presented. Also the distinction between automatically trainable and knowledge-engineered systems is stated.

4 Information Quality in Information Extraction

IE Systems are far from perfect and might produce spurious information or miss information that they should extract. Real-world applications like a financial analyst tracking business for a specific sector in news articles or a medical research group monitoring the outbreak of diseases from medical records, rely on the information extracted by data management systems for unstructured data. IE is just one of three steps from unstructured text to valuable knowledge. Below, the three basic steps of exploiting information in unstructured sources are mentioned:

1. Retrieving relevant documents from input text collection
2. Extracting structured object like entities and relations between the entities
3. Performing analytics such as precise queries or data mining tasks over the extracted relations

In each of these steps information quality is a constant concern and can be measured and improved. Therefore we organize this survey on information quality in IE along the processing pipeline from unstructured text to exploiting the information buried in the text by answering queries and performing analytics. First we investigate document retrieval and evaluate different strategies to find relevant documents and their effect on the output quality. The next section tackles quality improvement approaches for IE systems. Topics like declarative information extraction, uncertainty management and adaptive information extraction are covered. The last section deals with join and query optimizations to improve the analytical processes on extracted data.

4.1 Document Retrieval

IE is the process of identifying instances of structure like entities and relationships between entities in unstructured sources. Natural language text documents such as

emails, contracts, reports and even web pages can be listed as unstructured sources. The goal is to exploit these text documents to the fullest by extracting information and populate relational database tables for further analytical processing. Databases of text documents can be extremely large considering the examples of a big company's archive of legal documents, customer e-mail or the ultimate repository for text base documents, the *World Wide Web*. A simple approach to extract information is to retrieve and process every single document in the input database. The problem with this strategy is that the components that transform natural language text into structured, normalized database form are computationally very expensive. Hence these systems are in the need of a more efficient execution strategy. A refinement idea first mentioned in 1992 at the *Message Understanding Conference* [LT92] is to discriminate between relevant and irrelevant documents regarding the extraction task within the text database.

In this section we will review the impact of document retrieval strategies on the efficiency and ultimately on the result quality of IE systems. First we will look at the early ideas of text and relevance filtering at the *MUC* followed by advanced document retrieval strategies. We introduce a quality aware optimizer and methods to perform IE over evolving text collections.

4.1.1 Text Filtering

The *MUC* realized that approximately 50% of the documents in the corpora that were used for the information extraction evaluation task were irrelevant [LT92] for the evaluation itself. Due to this discovery the *MUC* decided to create a new sub task. The task was to categorize the corpora into relevant and irrelevant documents to achieve higher efficiency and better performance for the evaluation task of IE systems. The idea of discriminating the input documents to enhance the evaluation task evolved to a component in IE systems to make the extraction system itself more effective. At the *third* and *fourth MUC* the effect of relevance filtering on the effectiveness of the extraction system was recognized but unfortunately could not be quantified. The authors of [LT92] mention that concentrating just on relevant documents signigicantly reduces the chance of extraction spurious data at later stages of the analysis. Chinchor et al. [CLT93] propose to adopt metrics to evaluate the effectiveness of extraction system from the field of *Information Retrieval*. We will take a look at IR in the next section to find out which concepts we can transfer to IE.

4.1.2 Information Retrieval

The focus in *Information Retrieval* research lays on text classification systems which make binary decisions for text document as either relevant or non-relevant with respect to a user's information need. Manning et al. name three things that are necessary to measure information retrieval effectiveness [MRS08].

1. A document collection
2. A test suite of information needs expressed as queries
3. A set of binary decisions of either relevant or non-relevant for each query-document pair.

The information retrieval system returns a set of documents that were classified as relevant with regard to the query representing the information need. Capturing the user information need is not a trivial task. Jain et al [JIG08] introduces another factor to the need of the user based on efficiency and output quality needs. With the pre-assessed binary decision for each query document pair which Manning introduces as the *ground truth* or *gold standard* judgment of relevance the effectiveness of the information retrieval system can be measured. Two main measures for effectiveness are introduced, *precision* and *recall*. These are defined in [MRS08] as follows.
Precision (P) is the fraction of retrieved documents that are relevant

$$Precision = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

Recall (R) is the fraction of relevant document that are retrieved

$$Recall = \frac{\#(\text{relevant items retrieved})}{\#(\text{relevant items})} = R(\text{retrieved}|\text{relevant})$$

To summarize the decision an information retrieval system can make we take a look at a contingency table in Figure 4.1:
The measures *precision* and *recall* can then be expressed as:

$$Precision = \frac{tp}{(tp + fp)}$$

$$Recall = \frac{tp}{(tp + fn)}$$

	Relevant	**Nonrelevant**
Retrieved	true positive (tp)	false positive (fp)
Not retrieved	false positive (fn)	true negatives (tn)

Figure 4.1: Contingency table for information retrieval systems [MRS08]

In Section 2.3 we introduced the information quality measure *accuracy*. *Accuracy* is defined as the percentage of correct pieces of information in regard to the total amount of pieces. From the contingency table above the measure *accuracy* can be expressed as:

$$Accuracy = \frac{(tp + tn)}{(tp + fp + fn + tn)}$$

According to [MRS08] *accuracy* is nevertheless no satisfying measure for effectiveness since a IR system labeling all documents as non-relevant would have a high accuracy but would be completely unsatisfying to the system user. Therefore the measures of *precision* and *recall* are more appropriate to evaluate IR systems. These two numbers express the amount of true positives while mentioning the percentage of all relevant documents have been found and also the amount of false positives. An IR system with a high degree of effectiveness should deliver a good amount of recall while tolerating only a low number of false positives. Taking a closer look at these two measures it becomes clear that they function in a trade off to one another. Perfect recall is achieved by returning all documents but leads to a very low degree of precision due to the vast amount of false positives. Tuning a system in order to get very high precision will lead to the fact that the percentage of relevant documents that are actually retrieved (*recall*) will be lower.

In [CLT93] the two measures of *precision* and *recall* are adopted to measure effectiveness of an extraction system. *Recall* is described as the degree of *completeness* of the extracted records and *precision* is described as the degree of *accuracy* of the results. We cover the evaluation of IE systems in more detail in Section 4.2.1.

4.1.3 Document Retrieval Strategies

As stated in the beginning of this section,IE systems need techniques to replace the computationally expensive natural language processes such as named entity tagging, syntactic parsing and rule matching with relatively cheap filtering components. This is especially important when dealing with large databases or the web as input for the extraction system. In the context of this section we can note that the extraction task is composed of two main parts. First, retrieving relevant documents from a text database and second processing the retrieved documents by IE systems. The basic text filtering components from the *MUC* have evolved over the last two decades. In this section we will list and categorize strategies to retrieve the relevant documents.

The goal of a document retrieval component is to efficiently filter the correct subset of documents from the input text database that are likely to contain the structured information of interest. For example taking the *World Wide Web* as the input database which exceeds millions of documents, it falls into place that relatively few documents will contribute to the extraction task at hand.

In general, document retrieval strategies can be categorized as crawl-based and query-based.

Crawl-based Strategies

Ipeirotis et al. [IAJG06] mention *Scan* as the basic strategy that processes every document in the text collection exhaustively. This strategy does not provide any text filtering therefore produces perfect *recall*. A major disadvantage of these scan based strategies is inefficiency when the text collection comprises a large number of non-relevant documents. Examples of real world information extraction systems that use *Scan* as the document retrieval strategy are [Gri97] and [YG98]. Although *Scan* does not perform any relevance filtering it is mentioned here as a reference for later performance and result quality evaluation.

A natural improvement of *Scan* and following the idea of text filtering in [LT92] is *focused Scan*. Ipeirotis et al. [IAJG07] coined this term for execution strategies for text-centric tasks like IE that classify documents in the collection as useful. The *MUC* text filtering approach belongs to this category. In [LT92] system level text filtering components can be further categorized into pre-, intra-, post-parse filtering according to the point in time where the judgment takes place. Later systems such as *DIDEROT* [CWG+93], *Proteus* [GHY02] and *Ripper* [Coh95] can be seen as pre-parse text filtering systems using *filtered Scan*. Sarawagi [Sar08] categorizes

this type of strategy as focused crawling. For hyperlinked text collections like the *World Wide Web* the area of focused crawling brings a subset of relevant documents into the input database. The extraction system in [Bri99] called *DIPRE* uses this strategy based on the URL structure. A survey on focused crawling algorithm can be found in [Nov04].

Crawl-based strategies are usually slow but they produce a result with a high degree of *completeness*. If the user is after an exhaustive quality oriented answer and is willing to wait a relatively long time, crawl-based strategies fit very well.

Query-based Strategies

The second category is query-based document retrieval strategies. Sarawagi [Sar08] sums this category up as strategies that exploit pre-existing indices on the unstructured source to fetch only the documents of interest. Two types of queries against an index are mentioned, standard IR-style keyword queries and pattern queries for entity-level filtering.

Agichtein and Gravano introduce an unsupervised query based method for retrieving relevant documents in [AG03]. The System *QXtract* automatically derives queries from a sampling procedure. A small set of documents is processed by an IE system producing result tuples and an identifier for the document that are useful for this extraction task. *QXtract* takes the useful documents as good examples and generates queries that are likely to find documents relevant to the extraction task. Ipeirotis et al. [IAJG07] [IAJG06] call this Automatic Query Generation strategies.

These type of strategies are less time-consuming and therefore very efficient. The trade-off is the *completeness*. Efficiency related strategies meant to avoid processing useless documents may compromise the output *completeness*. Sometimes this quick and dirty approach is exactly what the user is looking for.

4.1.4 Active Learning

Rule-based IE systems that use machine learning algorithms to generate the extraction rules need a considerable amount of manual labor to produce a set of annotated training documents. These documents are selected from a large text collection of unlabeled documents. In order to optimize this manual annotation process, IE researchers have investigated the field of *Active Learning* [SW01], [TCM99], [CDPW02]. The principle of *Active Learning* is to select a document retrieval strategy to find the next document to be presented to the user annotating training docu-

ments. The goal is to find the document that contains the richest set of entities and relations in terms of training data for the machine learning algorithms. Document are also retrieved that contain information that have not yet been considered in the rule generating process.

The author of [FK03] investigates several selection strategies for *Active Learning* and evaluate them regarding the *recall/precision* trade-off. Below is a list of these strategies and an evaluation of the result in [FK03].

Compare This strategy calculates a degree of difference to previously processed documents and selects the least similar one. With this strategy, the extraction system that is being trained produces high *recall* but very low *precision*.

ExtractCompare In this algorithm, the difference between already annotated information and the next document is the decision factor. With the worst *precision* and an average *recall*, this strategy is not favorable.

Melita This strategy selects documents for annotation that do not match patterns that were already learned from previous extractions. It leads to high *recall* but has one of the worst degrees of *precision*.

NameFreq The documents with the most unusual personal names are selected for annotation in this approach. For extracting people's names, *NameFreq* has the highest *precision* but by far the worst *recall*.

Ensemble is a combined strategy. It uses half the documents selected by *Melita* and half the documents selected by *NameFreq*. The goal is to combine the *precision* of *NameFreq* and the *recall* of *Melita*.

Finn and Kushmerick [FK03] show that the selection strategy when deciding which document to use next for training annotations is important for the quality of the trained extraction system. Either *precision* or *recall* oriented extraction rules are learned depending on the strategy used for *Active Learning*.

4.1.5 Incorporating Quality Metrics

One fundamental problem of traditional document retrieval systems is the assumption that documents are of the same quality. Algorithms and techniques were developed for library systems where the source documents have been carefully chosen. The pre requirements for modern IE systems using traditional document retrieval

strategies are very different. Source documents are of varying quality. Incorporating the quality of the source document into the retrieval process is necessary when using large heterogeneous text document collection like the web as input for IE systems. In [ZG00] information quality is incorporated into information retrieval in web search engines. Six quality metrics are investigated:

- *currency*: reflects the time stamp of last modification for the document
- *availability*: broken links to number of total links
- *information-to-noise ratio*: ratio of length of token after lexical analysis to size of the document
- *authority*: based on score by Yahoo Internet Life (YIL) reviews
- *popularity*: number of links pointing to site
- *cohesiveness*: how closely the major topics are related in the page

These metrics are used to determine the quality of web pages to use as metadata in later searches.

Although [ZG00] concentrates on information retrieval in web search the techniques to incorporated quality metrics can be easily adopted by document retrieval strategies in IE. *Currency*, *information-to-noise ratio* and *cohesiveness* are well suited to be considered during document retrieval next to general relevancy decisions. We will nevertheless keep focusing just on the four IQ dimensions that we defined in Section 2.3 for the remainder of this work.

4.1.6 Quality Aware Optimizer

The choice of documents processed by the IE systems affect the quality of the extracted objects. Processing non-relevant documents is not just inefficient and expensive, but it may also produce incorrect extraction results. So far, efficiency has been the dominant consideration for the choice of document retrieval strategies. Jain and Ipeirotis [JI09] investigate the effect of document retrieval strategy on the result quality and use that information for a quality-aware optimizer for IE systems. They propose an analytical model and a randomized maximum likelihood approach to estimate system parameters for the execution of IE systems. Based on this, they introduced an end-to-end quality-aware optimizer to choose the best execution strategy for the user's efficiency and quality constrains concerning *accuracy*

and *completeness*. Execution strategy incorporates extraction system parameters and document retrieval strategies. The authors of [JI09] utilize *Receiver Operating Characteristic* curves to express the output quality of different execution strategies. Section 4.2.1 introduces *Receiver Operating Characteristic* as a method to describe the behavior of IE systems for a single document.

4.1.7 Information Extraction over Evolving Data

Many real-world text corpora evolve over time. IE over dynamic data means performing extraction periodically therefore the same processes happen repeatedly. Each time documents are added, deleted or modified, extraction systems are required to capture the information contained in the new set of documents. Performing IE on each snapshot of the set in isolation from scratch is tedious. Chen et al. propose in [CDYR08] a system called *Cyclex* that efficiently executes repeated IE over evolving data by recycling previous results. For time-sensitive applications like stock analysis or auctions, applying the extraction processes in a short interval is very important. *Timeliness* is an important characteristic that the results of extraction systems for these applications need. This introduces a new dimension of result quality next to *accuracy* and *completeness* from Section 2.3. The key idea in the approach in [CDYR08] is to exploit the fact that consecutive snapshots of a document set contain overlapping data so previous extraction results can be recycled.

4.2 Extraction System

In this section IE systems are investigated for approaches to manage result information quality. To address quality issues we first need to note methods that have been proposed over the years to evaluate the components of extraction systems. Ideas and concepts to improve result quality for Named Entity Tagging and Pattern Matching are summarized.

4.2.1 Information Extraction Evaluation

Evaluation has a long history in IE. The *MUC* can be considered the starting point, where most of the evaluation methodology was developed. Hirschman [Hir98] gives an overview of the eleven years and seven installments of this competition which attracted talented researches from many different areas to try their best to enhance and optimize IE systems. To evaluate the capability of extraction systems, the

MUC provided a standard annotated corpora and defined evaluation measures. We mentioned in Section 4.1.1 the adoption of measures for evaluation from the area of IR. At *MUC* the candidate extraction system had to fill templates with information extracted from documents in the corpora. As a reference value for each template human generated answer keys were provided. The answer keys are what we referred to as the *gold-standard* in Section 2.4. The *MUC* evaluated extraction systems in two aspects, recognizing the correct type (TYPE) and identifying the exact text (TEXT). An extension of this idea was mentioned by Freitag in [Fre98] who proposes three different criteria for matching the reference key and the extracted template fill, namely *Exact*, *Contains*, *Overlap*. TEXT just has a binary value for matching. The evaluations system determine for both aspects the values of correctly filled slots in the template (COR), actual fills attempted (ACT) and the total amount of possible correct fills (POS) for each system based on the same *MUC* corpora. Optimization ideas for evaluation systems using relevance filtering of documents in the *MUC* corpora are mentioned in Section 4.1.1. The following equations are presented based on the three mentioned measures [GS96] [CLT93]:

$$Recall/Completeness = \frac{\text{COR} + 1/2\ \text{PART}}{ACT}$$

$$Precision/Accuracy = \frac{\text{COR} + 1/2\ \text{PART}}{POS}$$

Since extraction systems are evaluating the two aspects *TYPE* and *TEXT*, 1/2 PART represents partial correct fills where just one of the two aspects is a correct fill. For example, in the case that the entity type was recognizes correctly but the span of the entity was erroneous.

The *MUC* proposed an additional single value score called *F-Measure* for extraction systems. It is the harmonic mean of *precision* and *recall*. Balanced systems are given preference based on the nature of the harmonic mean.

In addition to the evaluation measures *precision*, *recall* and *F-Measure* the *MUC* made another important contribution to the field of IE. The *MUC* corpora is a large collection of annotated documents and is available for training and testing along with the evaluation software [Dou98].

In the equations above we used *recall*/*completeness* and *precision*/*accuracy* as synonyms. In IQ literature, the dimensions *accuracy* and *completeness* are mentioned frequently. We define the two dimensions in Section 2.3. IE literature uses

different terminology for the two dimensions namely *precision* and *recall*. We use in the context of this work *recall*/*completeness* and *precision*/*accuracy* interchangeably.

Recent systems use the notion of *precision* and *recall* to evaluate the quality of IE systems. The natural trade-off between *precision* and *recall*, see Section 2.3, lets extraction systems be trained (machine learning) or set up (hand crafted) for one or the other. Precision-oriented systems are useful to extract critical data for example from medical records. Extracted information should have a high fraction of correct tuples in this case. Recall-oriented systems on the other hand are tuned to extract as much tuples as possible to not miss any information in the input documents. This might be valuable for an analyst interested in tracking all company acquisitions mentioned in news articles.

Automatic Content Extraction (ACE) Program

ACE [DMP+04] evaluation methods are far more complex. Each entity type has a parameterized weight. The evaluation score is comprised of the weights of extracted entities. Errors such as missed entity or incorrect entity type have customizable cost values. These costs are subtracted from 100% to build the final score called Entity Detection and Recognition Value. The customizable costs make the evaluation method more complex and flexible. To stay consistant with the IQ dimensions defined in Section 2.3, we will not persue the *ACE* evaluation concept in this work.

Receiver Operating Characteristic

The above mentioned trade-off between *precision* and *recall* can be expressed in so called *precision recall curves*. In [PF01] Provost and Fawcett realize that during the task of evaluating classifier, the measure of *precision* depends heavily on the distribution of good and bad documents in the test set. Therefore Jain and Ipeirotis [JI09] state that using *precision recall curves* is not a statistically robust manner to characterize IE systems. They propose to utilize *Receiver Operating Characteristic* to express the output quality of an extraction system. By plotting the true positive rate *tp* (*recall*) as the ordinate and the false positive rate *fp* as the abscissa a *Receiver Operating Characteristic* curve graphically summarizes the trade-off between *precision* and *recall* orientation of IE systems in a statistically robust manner.

4.2.2 Information Extraction Operators

In rule-base IE, structured objects like named-entities and relationship are extracted base on a coordinated set of rules called operator or annotator. Extracted structures are referred to as annotation in the unstructured text. Basic operators have one specific type of entity as an extraction goal. Rules to describe what to extract can be defined in regular expressions or dictionaries. As mentioned in Section 3.2.3 the rules are either learned from a set of pre-annotated documents by some machine learning techniques or hand-crafted by domain experts describing structure to be extracted. The first step in the IE process is to identify *Named Entities* which are later used as attributes for the more complex structures like relationships between previously extracted entities. For example, Redmond needs to be identified as a city and Microsoft as a company before the fact that Redmond is the headquarter of Microsoft can be extracted. Even entity level extraction systems are cascaded where the output of the first operator is used as the input for the next operator. For example an address block that is extracted out of previously annotated name, street and city structures in a text. Over the last two decades since the task of entity extraction was first introduced, many optimizations for basic operators have been proposed.

Bootstrapped learning Bootstrap-learning systems start with a set of seed instances of a given relation, which are used to identify extraction patterns for the relation. These patterns are then used to extract further instances [RJ99] [LG03] [AG00].

Unsupervised Information Extraction One major research path that received a lot of attention recently is semi-supervised or unsupervised methods of IE. The text collections nowadays are very large and extremely heterogeneous. Even with machine learning entity tagging, extensive manual labor is inevitable. Pattern generation for relationship extraction is tedious for open domains. Etzioni et al. [ECD+05] develop an unsupervised, domain-independent system to extract information from the *World Wide Web* called *KnowItAll*. Knowledge is extracted from text without using hand-tagged training examples. In order to produce high *recall* in the result despite the fact that no information about extracted objects is known prior to the extraction process *KnowItAll* has three components [ECD+05]:

- **Pattern Learning** (PL): learns domain-specific pattern (extraction rules and

validation pattern)

- **Subclass Extraction** (SE): automatically identifies subclasses of entities
- **List Extraction** (LE): locates lists of class instances

Domain-independent patterns are used to bootstrap the system to recursively find new entities. Fast deployment for many entity and relation types without the prerequisite of annotated training data is possible.

4.2.3 Declarative Information Extraction

Systems performing IE over large sets of text documents are complex. Providing efficient support to develop, debug and optimize these systems is subject to recent IE research. Decomposing the extraction task in smaller subtasks and stitching extraction "blackboxes" together is the usual practice to develop large IE systems. IE programs created in such a way can become complex and difficult to understand very fast. Recent research activities like UIMA [Apa08], GATE [CMBT02] and xlog [SDNR07] propose *compositional* frameworks and declarative languages to develop extraction programs. IE programs written in such languages are easier to develop, debug and maintain. Other works are dealing with optimizing such extraction programs via query optimization, Section 4.3.1, to execute them effectively over evolving data, Section 4.1.7, to make them best-effort, Section 4.3.5 and to add provenance, Section 4.3.4.

Algebraic Information Extraction

The objective for this algebraic approach to rule-based IE is to be scalable for large data sets and large numbers of results. Traditional grammar-based IE systems using series of cascading rules to extract entities have significant drawbacks when it comes to scalability. To avoid those bottlenecks, Reiss et al. [RRK+08] introduce an algebraic approach. Inspired by relational databases, the authors develop a system where annotator rules are treated as queries in a formal algebraic framework. They present an SQL-like language which enables them to make use of standard query optimizations techniques, see Section 4.3.1.

4.2.4 Uncertainty Management

Even with the latest IE tools using methods like *Conditional Random Fields* (CRF) [LMP01] [SP03] or *Hidden Markov Models* (HMM) [FM99] [SCR03], it is impossible

to guarantee perfect extraction *accuracy* in real-life deployments. For extraction tasks in open domains with extremely heterogeneous sources we stated in Section 3.2.3 that statistical extraction system are most appropriate. To overcome low *accuracy* extraction like in [PM06] of 70-80% for multi-field extraction and therefore 20-30% erroneous extracted structures, recent systems attach a confidence score to each entity expressing the probability that the extracted object is correct. In addition to extracted, objects an associated confidence value is produced. IE systems using CRF and HMM also produce ranked lists of extraction with different probability of correctness for each token or group of tokens. The goal of such confidence values are at the end to have extraction systems that capture these scores in an imprecise data model to provide an indicator of correctness for answer of queries over the extracted data. Uncertainty management is a promising approach to improve the overall result quality of exploiting information from unstructured text. To represent the uncertainty in IE probabilities of extraction results need to be handled by an imprecise data management system. Probabilistic databases offer the necessary features. Generic models of imprecision and techniques to process queries and quantify the resulting probability to the user permitting risk assessment and decision making under uncertainty.
In [GS06], Gupta and Sarawagi present a method to populate imprecise databases from statistical models of IE. The authors combine the probabilistic extraction models of *Conditional Random Fields* with well-known row-level and column-level uncertainty in imprecise databases. They show that capturing extraction uncertainty in a probabilistic database produces results with higher *accuracy* thus less errors than the usual practice of storing the topmost extraction in a conventional database.
Culotta and McCallum [CM04] provide also a model for assessing the confidence of extracted information using *Conditional Random Fields*. Their work focuses on assigning accurate confidence values to individual occurrences of an extracted field based on textual features. Based on the probabilistic nature, statistical IE system are easy to extend to output a list of the k highest probability extractions instead of a single most likely extraction. Uncertainty management in rule-based IE is much more difficult. Michelakis et al. [MKHV09] propose a probabilistic framework for rule-base information extraction. Uncertainty is modeled through varying precisions associated with each rule in an annotator.

4.2.5 Ontology-based Information Extraction

Ontology development and population is a crucial task in semantic web applications. Unfortunately cost-intensive manual labor is necessary to perform these tasks. Therefore the automation of relating text to ontologies opens up new possibilities in optimization of semantic web application. *Ontology-based Information Extraction* enables automatic ontology population by identifying relevant key terms and relating them to concepts in the ontology. Output of *Ontology-based Information Extraction* is the original document where entities in the text are annotated with links to their semantic description. Formal ontologies [Gua98] are used rather than just a dictionary like in traditional IE systems to annotated entities. SemTag [DEG+03], Magpie [DD04], and KIM [KPT+04] are examples of systems that semantically annotate web pages and link entities to ontological classes. Other than in traditional IE, results in *Ontology-based Information Extraction* can be more than either correct or incorrect. Due to the ontological classification the distinction is a bit more fuzzy. Even the partial correctness proposed by the *MUC* evaluation metrics in 4.2.1 is still insufficient to express different degrees of correctness when using ontological classification. Maynard et al. [MM06] proposes new evaluation metrics for *Ontology-based Information Extraction*. They use a variant of Hahn's Learning Accuracy measure called *Balanced Distance Metric* [May05] and integrate this with a standard *precision* and *recall* metric to measure similarity between Key (*gold standard*) and Response (output of the system). A graded correctness score is produced that mirrors semantic similarity. Although the *Balanced Distance Metric* score gives an intuitively plausible value the authors state that in some occasions the score does not conform well with human judgment.

4.2.6 Adaptive Information Extraction

It is desired to build IE systems that are scalable in multiple dimensions. The first dimension is to rapidly process large document collections. Attempts like declarative and algebraic IE make extraction systems scalable for huge amounts of input data. The second dimension is the heterogeneity of document types and formats. Adaptive information extraction combines efforts from machine learning and unsupervised IE to build extraction systems that are capable to capture information in a wide variety of document types and formats. Turmo [TAC06] introduces adaptive IE to extract business intelligence information from web pages. Kushmerick and Thomas [KT03] build information agents on the basis of adaptive IE [FMK05].

4.2.7 Enterprise Information Extraction

Chiticariu et al [CLRR10] investigate resent research developments and open challenges for Enterprise Information Extraction. For applications like semantic search, data as a service, business intelligence or data-driven mashups extraction systems have to meet special requirements to handle these enterprise-level data-intensive tasks gracefully. The authors identify three main requirements, scalability, accuracy and usability. Several of the optimization technologies for IE systems presented in this work are listed to meet these requirements [CLRR10].

4.3 Queries over Extracted Data

Exploiting information buried in unstructured text is more than just extracting named entities and relationships between those entities. This information is to be stored in a structured database for further processing. The last important step of gaining knowledge is to generate queries against the extracted database and thereby producing answers for the user's questions. This Section presents recent approaches to optimize query generation and enhance output quality by applying various join strategies. We also review the promising approach for producing higher quality results by incorporating the provenance of both answers and non-answers. We summarize efforts to use user interaction and user feedback to make IE better.

4.3.1 Query Optimization

Jain et al. propose in [JDG07] and [JDG08] a querying approach to exploit information that is embedded in unstructured text. Structured relations are extracted from text databases by an IE system and stored in a way that SQL queries can be issued over the relations. The two main challenges of efficiency and output quality are addressed. Depending on the efficiency and result quality requirements the query optimizer selects the appropriate query execution plan. Figure 4.2 illustrates the stages considered in the execution plan for an example query Q1 according to [JDG07].

```
Q1:  SELECT Company, CEO FROM CompanyInfo WHERE Location = 'Redmond'
```

We covered the possible document retrieval strategies in Section 4.1.3 and the extraction process is according to Section 4.2. Decisions along the extraction plan are based on the *Goodness* metric. Jain et al. introduce *Goodness* of a query execution

as a function of result quality per execution time unit. *Result quality* is defined as the mean of *precision* and *recall*. With these measures candidate execution strategies are evaluated and chosen according to the user-specific quality requirements. Wang et al. [WFGH10] address the problem of querying probabilistic data, see Section 4.2.4 with traditional query processing. They integrate an IE system based on *Conditional Random Fields* with a probabilistic database. The core algorithm for query processing is the Viterbi inference algorithm.

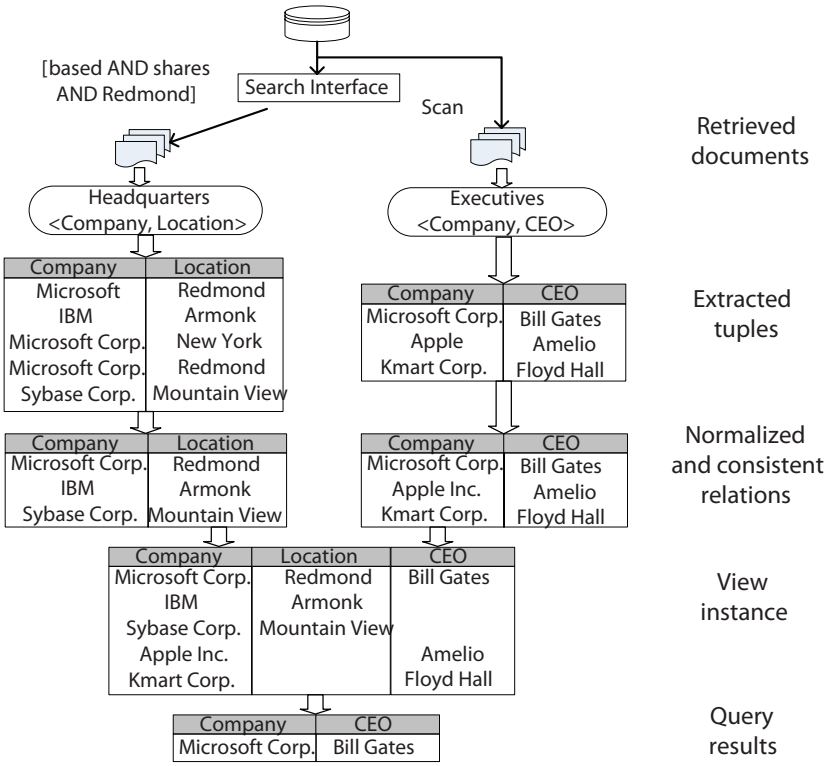

Figure 4.2: Stages in the execution of query Q1 from [JDG07]

4.3.2 Join optimization

Real-world applications of IE often require that the results from multiple IE systems be consolidated to form the output of interest. As opposed to single IE systems scenarios producing answers to complex SQL like queries, multiple IE systems have to be employed. Due to the noisy nature of IE, different join execution plans can produce results of different quality. Therefore, the quality of the consolidated output

is of crucial importance when facing the decision for a join strategy. Systems and architectures like Avatar [JKR+06], DBLife [DSC+07] and UIMA [Apa08] consolidate output from multiple IE programs to produce the desired result data. So far, these systems focus on efficiency as the consideration when choosing a join strategy. Jain et al. [JIGD08] present an in-depth analysis to understand, estimate and incorporate output quality into the processing of joins for IE.

4.3.3 Natural Language Queries

Natural Language Query answering can be performed on structured knowledge-basis or on free-text-basis. IE plays an important role in the later. *Ontology-based Information Extraction*, see Section 4.2.5, can be used to relate entities in text with concepts in an ontology. The difference in Natural Language Queries is the fact that a system has to convert natural-language questions into queries that can be processed by an IE system [KEW01].

4.3.4 Provenance

Provenance information describes the origin of data. It is the concept of attaching meta data about who created it, where and when specific data was introduced by the database community. Data management system needed to provide additional information about answers to queries since source data has become more and more unsupervised, decentralized and mostly arise as the result of many transformations and therefore considered less trustworthy. Data mining and query processing systems on extracted data from unstructured sources have the same type of problem. Extracted information is error prone and not guaranteed to be complete. Providing hints about the provenance or also called lineage can address the problem of uncertainty of extraction results. The objective of provenance in IE is to express the level of uncertainty. Lineage information is able to express in which stage of the extraction process results have their origin. For example, an extracted contact information with a name and a phone number can be investigated for provenance information. Questions like

- "Which annotator extracted the Name"
- "What regular expression recognized the phone number"
- "Which pattern matched to produce the contact relationship"

would lead to lineage information. Research on provenance in the database community is extensive, works like [cT04] and [CCT09] give a good introduction to this matter. Bohannon et al. [BMY+09] introduce an Extraction Management System that focuses as one of three main benefits on auditing which is basically providing lineage information about an extracted datum. Maintaining this lineage is essential for credit assignment along the pipeline, identifying erroneous operators and data, and timely reruns of operators to ensure high quality extraction. Huang et al [HCDN08] propose a conceptual framework to provide lineage information for non-answers. The introduced system provided techniques to detect and understand errors in the extraction process. To be more precise the system provides information why a tuple is not an answer to a query. Two types of information are provided. First, considering the example of extracting persons belonging to the program committee of a conference and the extraction system believes that person X does not belong to the committee. The systems in [HCDN08] can provide lineage information why person X was not recognized whilst in fact being on the program committee. Second, in case of a false positive extraction information about the fact that this tuple is not to be extracted are useful for the user. Huang et al. describe the framework to be able to answer questions like:

> "Could this non-answer become an answer, and if so, how? That is, what modifications to the existing database (for example, tuple insertions or updates) would be required for the non-answer to change to an answer?"

4.3.5 Best-Effort Information Extraction

The traditional approach to IE focuses on producing precise extraction results. Shen et al. propose in [SDM+08] an IE system that concentrates on best-effort IE to overcome limitations of traditional IE like long debug loops, inadequacy for time-sensitive tasks and inefficiency. The *interactive Flexible Extraction System* iFlex enables a developer to write an initial approximate extraction program using declaratives languages like introduced in Section 4.2.3. The approximate program processor of iFlex produces an approximate result which the developer can alter by refining the extraction program. The *next-effort assistant* of iFlex suggests refinement spots in the program to maximize the benefits of the developer's effort by obtaining increasingly more precise results. Best-effort IE is well suited for scenarios where *timeliness* and efficiency is more important than high *accuracy* of the result.

4.3.6 User Feedback

The imprecise nature of IE makes user feedback valuable for management systems over unstructured text. Extraction errors, if at all reported by users, have to be fixed by the developer of the extraction system. This procedure is slow and ineffective. Chai et al. [CVDN09] use declarative IE systems to efficiently incorporate user feedback into information extraction. They propose a system to extend declarative IE systems like DBLife [DSC+07] enabling them to use feedback from a multitude of users in a Web 2.0 fashion.

4.4 Existing Systems

In this section we present existing IE systems. This list is not meant to be exhaustive but rather gives a grasp of systems that implement the concepts presented in this work.

Snowball

Snowball [AG00] is a system for extraction relations from large collections of plain-text documents that requires minimal training for each new scenario. The goal of Snowball is not to capture all information from every document as completely as possible. It aims to extract tuples from all documents and to combine the result into one table. Snowball can be considered *precision*-oriented.

Avatar

AVATAR [JKR+06] is a rule-based information extraction system that aims at high *precision* extraction results from text-documents. The extraction system, developed at IBM Almaden Research Center, uses probabilistic database techniques to provides high *accuracy* while increasing *completeness*. Rule-based annotators are being mapped to queries against the probabilistic database.

KnowItAll

KnowItAll [ECD+04] is an unsupervised, domain-independent IE systems that extracts information from the *World Wide Web*. It specializes in extracting large collections of facts (e.g., names of scientists or politicians). The three main components *Pattern Learning*, *Subclass Extraction* and *List Extraction* to achieve a high

degree of *completeness* despite the fact that no knowledge about the structures is know prior to the process are explained in Section 4.2.2.

DBLife

DBLife [DSC+07] is a *Community Information Management* System. The system extracts and integrates information from raw Web pages and presents an unified view of entities and relationships in the database community. DeRose et al. propose a top-down, compositional and incremental approach framework called *Cimple* to build an extraction system that is easier to develop, understand, debug and optimize.

Cimple

Cimple [DSC+07] aims to build an IE systems as part of a larger community building platform, with concentrating on the development of a declarative information extraction language and optimization techniques and handling evolving data.

5 Conclusions

This chapter sums up results of this work and provides an outlook.

5.1 Summary

We started this work with an introduction to the research field of Information Quality in Chapter 2. Research activities of the last two decades on IQ dimensions are mentioned. We selected four IQ dimension and provided definitions that are consistent with the literature. Three dimensions are used throughout this survey to investigate IQ efforts in Information Extraction.

- Accuracy / Precision
- Completeness / Recall
- Timeliness

The fourth IQ dimension defined in Chapter 2 *uniqueness* is not considered in the survey. Nevertheless this dimension comes into play in the area of entity resolution. Text is inherently ambiguous and therefore techniques to disambiguate extracted data are much needed. *Uniqueness* can contribute in evaluating such techniques. Entity resolution is however not covered in this work.

Chapter 3 introduces Information Extraction as a pipeline of components that work together to extract structured objects like entities and relationships between entities from natural language text. The extracted information is stored in relational tables for further processing. Following components are explained in their functionality:

- Lexical Analysis
- Named Entity Recognition
- Syntactic Analysis

- Extraction Pattern Matching

Chapter 2 and Chapter 3 provide basic knowledge that is required to understand the following Chapter on Information Quality efforts in Information Extraction.

The survey in Chapter 4 is structured along the pipeline of components that are involved in the process of transforming unstructured text to valuable and exploitable knowledge. We look at following parts of this process separately.

- Document Retrieval
- Extraction System
- Query Optimization

Figure 5.1 summarize approaches in the first stage of managing unstructured data, namely document retrieval. Crawl- and query-based strategies follow different goals to improve IQ. Crawl-based strategies are *recall*-oriented and try to select the subset of documents that will lead to the most complete extraction result. Query-based strategies focus more on efficiency and *timeliness* to provide fast results. *Active Learning* provides different strategies to select the next document that is presented to a domain expert for annotation. Strategies are either *recall* or *precision* oriented. Combinations of strategies can lead to improvement in both dimensions. Quality aware optimizer improves also both, *accuracy* and *completeness* by using *Receiver Operating Characteristic* to provide a statistically robust way to express the trade-off between the two dimensions. Execution strategies are than selected that provide high quality result concerning *accuracy* and *completeness*. Cyclex enables IE to work efficiently on evolving data and opens up new possibilities for time-sensitive extraction tasks.

Efforts in extraction systems to improve IQ are illustrated in Figure 5.2. Unsupervised IE aims to provide a high degree of *recall* while making the extraction process very efficient and easy to maintain. Efficiency in developing, understanding and debugging extraction programs is also the goal of declarative IE. Uncertainty management incorporates probabilities of correctness of possible extraction results which achieves more accurate results. Ontology-based IE relates text to concepts in ontologies. Rich knowledge databases are used to find mentions of entities and relationships in text which improves *completeness* of the result. Adaptive IE is a combination of machine learning and declarative IE to provide efficient extraction

Document Retrieval Strategies	Accuracy/ Precision	Completeness/ Recall	Timeliness	Efficiency
Crawl-based Strategies		✓		
Query-based Strategies			✓	✓
Active Learning	✓	✓		✓
Quality Aware Optimizer	✓	✓		
Evolving Data (Cyclex)			✓	✓

Figure 5.1: Document Retrieval

systems that reach a high degree of *completeness* on very heterogeneous text collections.

Figure 5.3 shows in which dimensions the approaches in query optimization improve IQ. Query and join optimization lead to extraction result with higher *accuracy*. Providing provenance information enables optimization and correction of the extraction process and therefore improves *accuracy* as well. Best-effort IE aims at efficient extraction. Response and turnaround time, which we defined as *timeliness* are improved. User feedback provides valuable insights from the user to debug and optimize the extraction program in a timely manner. Optimization lead to a higher degree of *accuracy*.

5.2 Outlook

Information Quality is still a pressing problem in unstructured information management. In this work we focus on IQ assessment and improvement from the data perspective. More research is need to capture dimension that represent the user perspective of IQ. Dimensions like *relevancy* or *interpretability* are interesting to investigate in IE. IE evolves to extract also events, opinions and sentiments. User feedback is therefore of significant value. Provenance becomes even more important if we want to leverage user feedback to improve the quality of extraction over time.

Extraction System				
	Accuracy/ Precision	Completeness/ Recall	Timeliness	Efficiency
Unsupervised IE		✓		✓
Declarative IE				✓
Uncertainty Mgmt	✓			
Ontology-based IE		✓		✓
Adaptive IE		✓		✓

Figure 5.2: Extraction System

Query Optimization				
	Accuracy/ Precision	Completeness/ Recall	Timeliness	Efficiency
Query Optimization	✓			✓
Join Optimization	✓			✓
Provenance	✓			
Best-Effort IE			✓	
User Feedback	✓		✓	

Figure 5.3: Query Optimization

Bibliography

[ABB06] Fabrizio De Amicis, Daniele Barone, and Carlo Batini. An analytical framework to analyze dependencies among data quality dimensions. In *ICIQ*, pages 369–383, 2006.

[AG00] Eugene Agichtein and Luis Gravano. Snowball: extracting relations from large plain-text collections. In *DL '00: Proceedings of the fifth ACM conference on Digital libraries*, pages 85–94, New York, NY, USA, 2000. ACM.

[AG03] Eugene Agichtein and Luis Gravano. Querying text databases for efficient information extraction. *Data Engineering, International Conference on*, 0:113, 2003.

[All03] James F. Allen. Natural language processing. In *Encyclopedia of Computer Science*, pages 1218–1222. John Wiley and Sons Ltd., Chichester, UK, 2003.

[Apa08] Apache. Uima, June 2008. Documentation available at `http://incubator.apache.org/uima/`.

[App99] Douglas E. Appelt. Introduction to information extraction. *AI Commun.*, 12:161–172, August 1999.

[AS06] E. Agichtein and S. Sarawagi. Scalable information extraction and integration (tutorial), 2006.

[BMY+09] Philip Bohannon, Srujana Merugu, Cong Yu, Vipul Agarwal, Pedro DeRose, Arun Iyer, Ankur Jain, Vinay Kakade, Mridul Muralidharan, Raghu Ramakrishnan, and Warren Shen. Purple sox extraction management system. *SIGMOD Rec.*, 37:21–27, March 2009.

[Bri99] Sergey Brin. Extracting patterns and relations from the world wide web. In *WebDB '98: Selected papers from the International Workshop*

on The World Wide Web and Databases, pages 172–183, London, UK, 1999. Springer-Verlag.

[BSM03] M. Bovee, R. P. Srivastava, and B. Mak. A conceptual framework and belief-function approach to assessing overall information quality. *International Journal of Intelligent Systems*, 18:51 – 74, 2003.

[CCT09] James Cheney, Laura Chiticariu, and Wang-Chiew Tan. Provenance in databases: Why, how, and where. *Found. Trends databases*, 1:379–474, April 2009.

[CDPW02] Fabio Ciravegna, Alexiei Dingli, Daniela Petrelli, and Yorick Wilks. Timely and non-intrusive active document annotation via adaptive information extraction. 2002.

[CDYR08] Fei Chen, AnHai Doan, Jun Yang, and Raghu Ramakrishnan. Efficient information extraction over evolving text data. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 943–952, Washington, DC, USA, 2008. IEEE Computer Society.

[CFP03] Cinzia Cappiello, Chiara Francalanci, and Barbara Pernici. Time-related factors of data quality in multichannel information systems. *J. Manage. Inf. Syst.*, 20:71–92, December 2003.

[CKGS06] Chia-Hui Chang, Mohammed Kayed, Moheb Ramzy Girgis, and Khaled F. Shaalan. A survey of web information extraction systems. *IEEE Trans. on Knowl. and Data Eng.*, 18(10):1411–1428, 2006.

[CL96] Jim Cowie and Wendy Lehnert. Information extraction. *Commun. ACM*, 39(1):80–91, 1996.

[CLRR10] Laura Chiticariu, Yunyao Li, Sriram Raghavan, and Frederick R. Reiss. Enterprise information extraction: recent developments and open challenges. In *SIGMOD '10: Proceedings of the 2010 international conference on Management of data*, pages 1257–1258, New York, NY, USA, 2010. ACM.

[CLT93] Nancy Chinchor, David D. Leww, and Lynette Hirschman T. Evaluating message understanding systems: an analysis of the third message understanding conference (muc-3. *Computational Linguistics*, 19:409–449, 1993.

[CM03] William W. Cohen and Andrew McCallum. Information extractionfrom the world wide web. ACM SIGKDD 2003, 2003.

[CM04] Aron Culotta and Andrew McCallum. Confidence estimation for information extraction. In *Proceedings of HLT-NAACL 2004: Short Papers on XX*, HLT-NAACL '04, pages 109–112, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

[CMBT02] Hamish Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. Gate: A framework and graphical development environment for robust nlp tools and applications. 2002.

[Coh95] William W. Cohen. Fast effective rule induction. In *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123. Morgan Kaufmann, 1995.

[CS99] Michael Collins and Yoram Singer. Unsupervised models for named entity classification. 1999.

[cT04] Wang chiew Tan. Research problems in data provenance. *IEEE Data Engineering Bulletin*, 27:45–52, 2004.

[CVDN09] Xiaoyong Chai, Ba-Quy Vuong, AnHai Doan, and Jeffrey F. Naughton. Efficiently incorporating user feedback into information extraction and integration programs. In *Proceedings of the 35th SIGMOD international conference on Management of data*, SIGMOD '09, pages 87–100, New York, NY, USA, 2009. ACM.

[CWG+93] Jim Cowie, Takahiro Wakao, Louise Guthrie, Wang Jin, James Pustejovsky, and Scott Waterman. Diderot information extraction system. 1993.

[Cyk96] Stern Cykana, Paul. Dod guidelines on data quality management. In *Proceedings of the Conference on Information Quality, Cambridge, MA, 1996, pp. 154-171*, pages 154–171, Cambridge, MA, 1996.

[CZW98] Ying Chen, Qiang Zhu, and Nengbin Wang. Query processing with quality control in the world wide web. *World Wide Web*, 1:241–255, April 1998.

[DD04] John Domingue and Martin Dzbor. Magpie: supporting browsing and navigation on the semantic web. In *Proceedings of the 9th international conference on Intelligent user interfaces*, IUI '04, pages 191–197, New York, NY, USA, 2004. ACM.

[DEG+03] Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. In *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pages 178–186, New York, NY, USA, 2003. ACM.

[DMP+04] G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ace) program–tasks, data, and evaluation. *Proceedings of LREC 2004*, pages 837–840, 2004.

[Dou98] A Douthat. The message understanding conference scoring software user's manual. In *Proceedings of the 7th Message Understanding Conference*, 1998.

[DRC+06] Anhai Doan, Raghu Ramakrishnan, Fei Chen, Pedro Derose, Yoonkyong Lee, Robert Mccann, and Mayssam Sayyadian. Community information management. 2006.

[DRV06] AnHai Doan, Raghu Ramakrishnan, and Shivakumar Vaithyanathan. Managing information extraction: state of the art and research directions. In *SIGMOD '06: Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 799–800, New York, NY, USA, 2006. ACM.

[DSC+07] Pedro DeRose, Warren Shen, Fei Chen, AnHai Doan, and Raghu Ramakrishnan. Building structured web community portals: a top-down, compositional, and incremental approach. In *VLDB '07: Proceedings of the 33rd international conference on Very large data bases*, pages 399–410. VLDB Endowment, 2007.

[ECD+04] Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and

Alexander Yates. Web-scale information extraction in knowitall: (preliminary results). In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 100–110, New York, NY, USA, 2004. ACM.

[ECD+05] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: an experimental study. *Artif. Intell.*, 165(1):91–134, June 2005.

[EH04] Martin J. Eppler and Markus Helfert. A classification and analysis of data quality cost. *Proceedings of the Ninth International Conference on Information Quality*, pages 311– 325, 2004. note note.

[Fel06] R. Feldman. Tutorial: Information extractin, theory and practise, 2006.

[FK01] C.W. Fischer and B.R. Kingma. Criticality of data quality as exemplified in two disasters. *Information and Management*, 39(2):109–116, December 2001. note note.

[FK03] A. Finn and N. Kushmerick. Active learning selection strategies for information extraction. In *Proc. Workshop Adaptive Text Extraction and Mining*, 2003. European Conf. Machine Learning.

[FM99] Dayne Freitag and Andrew Kachites Mccallum. Information extraction with hmms and shrinkage. In *In Proceedings of the AAAI-99 Workshop on Machine Learning for Information Extraction*, pages 31–36, 1999.

[FMK05] A. Finn, B. McLernon, and N. Kushmerick. Adaptive information extraction research at UCD. In *Proc. Dagstuhl Seminar on Machine Learning for the Semantic Web*, 2005.

[Fre98] Dayne Freitag. *Machine Learning for Information Extraction in Informal Domains.* PhD thesis, Carnegie Mellon University, 1998.

[GH96] L.A. Galway and C.H. Hanks. *Data quality problems in army logistics: classification, examples, and solutions.* RAND, first edition, 1996.

[GHOH+07] Mouzhi Ge, Markus Helfert, Robert O Hare, M Lynne Markus, and Barbara Klein. A review of information quality research - develop a research agenda. 2007.

[GHY02] Ralph Grishman, Silja Huttunen, and Roman Yangarber. Information extraction for enhanced access to disease outbreak reports. *Journal of Biomedical Informatics*, 35(4):236 – 246, 2002. Sublanguage - Zellig Harris Memorial.

[Gri97] Ralph Grishman. Information extraction: Techniques and challenges. 1299:10–27, 1997.

[Gro08] Butler Group. Document and records management. Technical report, Butler Group, 2008.

[GS96] Ralph Grishman and Beth Sundheim. Message understanding conference-6: a brief history. In *Proceedings of the 16th conference on Computational linguistics*, pages 466–471, Morristown, NJ, USA, 1996. Association for Computational Linguistics.

[GS06] Rahul Gupta and Sunita Sarawagi. Creating probabilistic databases from information extraction models. In *VLDB '06: Proceedings of the 32nd international conference on Very large data bases*, pages 965–976. VLDB Endowment, 2006.

[Gua98] Nicola Guarino. Formal ontology and information systems, 1998.

[HCDN08] Jiansheng Huang, Ting Chen, AnHai Doan, and Jeffrey F. Naughton. On the provenance of non-answers to queries over extracted data. *Proc. VLDB Endow.*, 1(1):736–747, 2008.

[Hir98] L Hirschman. The evolution of evaluation: Lessons from the message understanding conferences. *Computer Speech & Language*, 12(4):281 – 305, 1998.

[HLW99] Kuan-Tse Huang, Yang W. Lee, and Richard Y. Wang. *Quality information and knowledge*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.

[IAJG06] Panagiotis G. Ipeirotis, Eugene Agichtein, Pranay Jain, and Luis Gravano. To search or to crawl?: towards a query optimizer for text-centric tasks. pages 265–276, 2006.

[IAJG07] Panagiotis G. Ipeirotis, Eugene Agichtein, Pranay Jain, and Luis Gravano. Towards a query optimizer for text-centric tasks. *ACM Trans. Database Syst.*, 32, November 2007.

[JDG07] Alpa Jain, AnHai Doan, and Luis Gravano. Sql queries over unstructured text databases. In *ICDE*, pages 1255–1257, 2007.

[JDG08] Alpa Jain, AnHai Doan, and Luis Gravano. Optimizing sql queries over text databases. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 636–645, Washington, DC, USA, 2008. IEEE Computer Society.

[JI09] Alpa Jain and Panagiotis G. Ipeirotis. A quality-aware optimizer for information extraction. *ACM Trans. Database Syst.*, 34(1):1–48, 2009.

[JIG08] Alpa Jain, Panagiotis Ipeirotis, and Luis Gravano. Building query optimizers for information extraction: the sqout project. *SIGMOD Rec.*, 37(4):28–34, 2008.

[JIGD08] Alpa Jain, Panagiotis G. Ipeirotis, Luis Gravano, and Anhai Doan. Understanding, estimating, and incorporating output quality into join algorithms for information extraction. 2008.

[JKR+06] T. S. Jayram, Rajasekar Krishnamurthy, Sriram Raghavan, Shivakumar Vaithyanathan, and Huaiyu Zhu. Avatar information extraction system. 2006.

[Jon07] William Jones. Personal information management. *Annual Rev. Info. Sci & Technol.*, 41:453–504, December 2007.

[KEW01] Cody Kwok, Oren Etzioni, and Daniel S. Weld. Scaling question answering to the web. *ACM Trans. Inf. Syst.*, 19:242–262, July 2001.

[KGD97] Barbara D. Klein, Dale L. Goodhue, and Gordon B. Davis. Can humans detect errors in data? impact of base rates, incentives, and goals. *MIS Q.*, 21(2):169–194, 1997.

[KPT+04] A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff. Semantic annotation, indexing, and retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2(1):49–79, December 2004.

[KSW02] Beverly K. Kahn, Diane M. Strong, and Richard Y. Wang. Information quality benchmarks: product and service performance. *Commun. ACM*, 45(4):184–192, 2002.

[KT03] N. Kushmerick and B. Thomas. Adaptive information extraction: Core technologies for information agents. *Lecture Notes in Computer Science*, 2586, 2003. Intelligent information agents: The AgentLink perspective; M. Klusch, S. Bergamaschi, P. Edwards and P. Petta, editors.

[LG03] R. Y. Winston Lin and R. Grishman. Bootstrapped learning of semantic classes from positive and negative examples. In *In Proceedings of the ICML Workshop on The Continuum from Labeled to Unlabeled Data*, August 2003.

[LMP01] John Lafferty, Andrew Mccallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[LT92] David D. Lewis and Richard M. Tong. Text filtering in muc-3 and muc-4. In *Proceedings of the 4th conference on Message understanding*, MUC4 '92, pages 51–66, Morristown, NJ, USA, 1992. Association for Computational Linguistics.

[May05] Diana Maynard. Benchmarking ontology-based annotation tools for the semantic web. In *In UK e-Science Programme All Hands Meeting (AHM2005) Workshop Text Mining, e-Research and Grid-enabled Language Technology*, 2005.

[McC05] Andrew McCallum. Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9):48–57, 2005.

[MKHV09] Eirinaios Michelakis, Rajasekar Krishnamurthy, Peter J. Haas, and Shivakumar Vaithyanathan. Uncertainty management in rule-based information extraction systems. In *SIGMOD '09: Proceedings of the 35th SIGMOD international conference on Management of data*, pages 101–114, New York, NY, USA, 2009. ACM.

[MM06] Extraction Diana Maynard and Diana Maynard. Metrics for evaluation of ontology-based information. In *In WWW 2006 Workshop on Evaluation of Ontologies for the Web*, 2006.

[MMS93] Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini.

Building a large annotated corpus of english: the penn treebank. *Comput. Linguist.*, 19:313–330, June 1993.

[MRK03] Song Mao, Azriel Rosenfeld, and Tapas Kanungo. Document structure analysis algorithms: a literature survey. volume 5010, pages 197–207. SPIE, 2003.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[MWLZ09] Stuart E. Madnick, Richard Y. Wang, Yang W. Lee, and Hongwei Zhu. Overview and framework for data and information quality research. *J. Data and Information Quality*, 1:2:1–2:22, June 2009.

[Nov04] Blaz Novak. A survey of focused web crawling algorithms. 2004.

[Ols02] Jack Olson. *Data Quality: The Accuracy Dimension*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[ORH05] Paulo Oliveira, Fátima Rodrigues, and Pedro Henriques. A formal definition of data quality problems. In *IQ*, 2005.

[PF01] Foster Provost and Tom Fawcett. Robust classification for imprecise environments. *Mach. Learn.*, 42:203–231, March 2001.

[PLW02] Leo L. Pipino, Yang W. Lee, and Richard Y. Wang. Data quality assessment. *Commun. ACM*, 45(4):211–218, 2002.

[PM06] Fuchun Peng and Andrew McCallum. Information extraction from research papers using conditional random fields. *Information Processing and Management*, 42(4):963 – 979, 2006.

[Rau91] L.F. Rau. Extracting company names from text. In *Artificial Intelligence Applications, 1991. Proceedings*, pages 29 – 32, 1991.

[Red92] Thomas C. Redman. *Data quality: management and technology*. Bantam Books, Inc., New York, NY, USA, 1992.

[Red97] Thomas C. Redman. *Data Quality for the Information Age*. Artech House, Inc., Norwood, MA, USA, 1st edition, 1997.

[RJ99] Ellen Riloff and Rosie Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*, AAAI '99/IAAI '99, pages 474–479, Menlo Park, CA, USA, 1999. American Association for Artificial Intelligence.

[RRK+08] Frederick Reiss, Sriram Raghavan, Rajasekar Krishnamurthy, Huaiyu Zhu, and Shivakumar Vaithyanathan. An algebraic approach to rule-based information extraction. *Data Engineering, International Conference on*, 0:933–942, 2008.

[Sar08] Sunita Sarawagi. Information extraction. *Found. Trends databases*, 1(3):261–377, 2008.

[SCR03] Marios Skounakis, Mark Craven, and Soumya Ray. Hierarchical hidden markov models for information extraction. In *Proceedings of the 18th international joint conference on Artificial intelligence*, pages 427–433, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[SDM+08] Warren Shen, Pedro DeRose, Robert McCann, AnHai Doan, and Raghu Ramakrishnan. Toward best-effort information extraction. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1031–1042, New York, NY, USA, 2008. ACM.

[SDNR07] Warren Shen, AnHai Doan, Jeffrey F. Naughton, and Raghu Ramakrishnan. Declarative information extraction using datalog with embedded extraction predicates. In *VLDB '07: Proceedings of the 33rd international conference on Very large data bases*, pages 1033–1044. VLDB Endowment, 2007.

[SP03] Fei Sha and Fernando Pereira. Shallow parsing with conditional random fields. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 134–141, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[SW01] Tobias Scheffer and Stefan Wrobel. Active learning of partially hidden markov models. In *In Proceedings of the ECML/PKDD Workshop on Instance Selection*, 2001.

[TAC06] Jordi Turmo, Alicia Ageno, and Neus Català. Adaptive information extraction. *ACM Comput. Surv.*, 38(2):4, 2006.

[TCM99] Cynthia A. Thompson, Mary E. Califf, and Raymond J. Mooney. Active learning for natural language parsing and information extraction. In *Proc. 16th International Conf. on Machine Learning*, pages 406–414. Morgan Kaufmann, San Francisco, CA, 1999.

[WFGH10] Daisy Zhe Wang, Michael J. Franklin, Minos Garofalakis, and Joseph M. Hellerstein. Querying probabilistic information extraction, 2010.

[WRY01] Lee Yang W Wang Richard Y., Ziad Mostapha. *Data Quality*. Springer, first edition, 2001.

[WS96] Richard Y. Wang and Diane M. Strong. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4):5–33, March 1996.

[WW96] Yair Wand and Richard Y. Wang. Anchoring data quality dimensions in ontological foundations. *Commun. ACM*, 39(11):86–95, 1996.

[YG98] Roman Yangarber and Ralph Grishman. Nyu: Description of the proteus/pet system as used for muc-7. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998.

[Zec97] Klaus Zechner. A literature survey on information extraction and text summarization, 1997.

[ZG00] Xiaolan Zhu and Susan Gauch. Incorporating quality metrics in centralized/distributed information retrieval on the world wide web. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 288–295, New York, NY, USA, 2000. ACM.

List of Figures