A DATA-INTENSIVE APPROACH TO NAMED ENTITY RECOGNITION USING DOMAIN AND LANGUAGE INDEPENDENT METHODS

A Dissertation Submitted
to the Graduate School
University of Arkansas at Little Rock

in partial fulfillment of requirements
for the degree of

DOCTOR OF PHILOSOPHY
in Integrated Computing

in the Department of Information Science
of the Donaghey College of Engineering and Technology

December 2010

Olukayode Isaac Osesina

M. S., Karlstad University, Sweden, 2006
M. A., Anglia Ruskin University, England, 2006
B. S., Budapest University of Technology and Economics, Hungary, 2003

This dissertation, “A Data-Intensive Approach to Named Entity Recognition Using Domain and Language Independent Methods,” by Olukayode Isaac Osesina, is approved by:

Dissertation Advisor:

John R. Talburt
Professor of Information Science

Dissertation Committee:

Elizabeth Pierce
Professor of Information Science

Mariofanna G. Milanova
Professor of Computer Science

Mihail E. Tudoreanu
Associate Professor of Information Science

Ningning Wu
Associate Professor of Information Science

Graduate Dean:

Patrick J. Pellicane
Professor of Construction Management

A DATA-INTENSIVE APPROACH TO NAMED ENTITY RECOGNITION USING DOMAIN AND LANGUAGE INDEPENDENT METHODS

by Olukayode Isaac Osesina, December 2010

ABSTRACT

In this dissertation, I proposed a novel approach to Named Entity Recognition (NER) in which the contextual and intrinsic indicators are used for locating named entities and their semantic meanings in unstructured textual information (UTI). Named entity is the process of locating a word or a phrase that references a particular entity within a text. The data-intensive approach introduced in this dissertation departs from the traditional Natural Language Processing used in NER tasks in that it does not apply linguistic rules or knowledge in the entity recognition process. It leverages the wide availability of huge amounts of data as well as high-performance computing to provide a NER solution that is independent of the UTI language and subject domain. The use of complex rules, data models and statistical algorithms are substituted with the scanning of annotated large volumes of data at high speeds to find exact or similar known instances of the solution.

The proposed NER approach does not require external linguistic resources (e.g. language dictionary) or subject domain resources (e.g. glossary, gazette) in order to acquire the language or subject domain knowledge. Instead, it derives all such knowledge from example documents in which entities of interest have been annotated. The text specific characteristics (language and subject) is decoupled from the analytical processes of locating entities within UTI;

consequently, the developed system is agile and can be applied uniformly to UTI in different languages and domains for which annotated example documents exist. A key feature of this NER approach is its ability to determine the semantic meaning or role of an entity within UTI. For example, differentiating between dates of independence and the civil war found in the same document.

The motivation behind the proposed NER approach, the different techniques used for locating entities and disambiguating among them, limitations of the system as well as discussion of experiments results are presented in this dissertation.

Dedication

I dedicate this dissertation to my parents Olufemi J. and Oluwatoyin D. Osesina for their immeasurable sacrifices to ensure my happiness and success.

Acknowledgement

I am very grateful for the invaluable guidance and support of my mentor and dissertation advisor Dr. John R. Talburt without which this dissertation would have been possible. Brainstorming ideas that that led to the successful completion of this dissertation with him were some of the highlights of my time in Little Rock. I particularly appreciate his wisdom in identifying my strengths and weaknesses very early and created the environment that enabled me to thrive during my research.

I am grateful for having the opportunity to work closely with Dr. Mihail “Edi” Tudoreanu. Our social media and situational awareness projects not only produced enviable results, they opened my eyes to the enormous possibilities of information visualization. Writing codes with Edi significantly improved my programming skills which was important to the successful completion of this dissertation. Working with him during the summers at Tec^Edge was very enjoyable.

I thank Dr. Elizabeth Pierce, Dr. Mariofanna Milanova and Dr. Ningning Wu all my dissertation committee members for their various suggestions and advise that improved the quality of my dissertation.

I am very grateful for the encouragement and friendship of Adeyemi Fowe and Olufunso Kumolu during my study in Little Rock. I fondly remember the days when Fowe turned up his “activities” a few notches, he makes nothing seem impossible.

I appreciate Mark and Lisa Leggett family for their love and support. Knowing them made living in Little Rock homey. I really enjoyed watching and cheering for their beautiful children (Isaac, Carmen and Joshua) at soccer matches.

My profound appreciation goes to my family who saw me through thick and thin. My parents Olufemi and Oluwatoyin Osesina whose enduring love and support were instrumental to my successful education career cannot be thanked enough. Their unselfish financial, moral and prayerful support made my education far away from home possible. I really admire you. To the best and most supporting siblings anyone can ask for Olujoke, Temitayo, Adenike, Olukunle and Ayobami Osesina who ensured that the huge geographical distances in our locations did not reduce the closeness of our relationships, I deeply appreciate you all.

Last but not least, I thank God for his infinite grace that helped me to successful complete my dissertation.

Table of Contents

Table of Figures

List of Tables

Chapter 1

Introduction

Named Entity Recognition (NER) is one of the major tasks in Natural Language Process (NLP). It is the process of locating a word or a phrase that references a particular entity within a text. NER is one of the first steps in computers and natural language interaction and it is used in solving problems such as information retrieval, machine translation and question answering.

NER has been researched for several decades and the popular approaches include linguistic grammar-based approach (Klein, et al., 1963) (Harris, 1962) where language grammar rules are manually programmed, statistical models (Jelinek, 1990) (Church, 1989) (Bahl, et al., 1976) where the rules are automatically extracted from annotated training data and approaches involving the use of linguistic resources such as WordNet (Fellbaum, 1998) or thesauri such as Roget's (Fellbaum, 1998). Combinations of the different approaches have also been explored.

In this dissertation, a data-intensive NER approach that uses the contextual and intrinsic indicators from annotated example documents to identify named entities and their semantic role(s) within unstructured text without understanding the meaning of the text is presented. This approach also identifies the semantic role of the identified entity within the text. It differs from the previous approaches in that it is independent of the language and domain of the text and does not require manual programming of grammar rules, external linguistic resource(s) or statistical models for locating named entities within unstructured text. This novel approach takes advantage of the increasing capacity and availability of high-performance computers to scan through large volumes of data to find exact or similar instances in which the solution is known.

Although significant progress has been made in NER over the past years; for example, the MUC-6 achieved a near human level performance of about 96% F-measure on newswire documents (Grishman, 1997) and CoNLL-2009 shared tasks achieved average F-measure of over 80% (Hajic, et al., 2009) on a selected number of multilingual documents. The unavailability of publicly shared semantically annotated corpus and the inability of most of the existing systems to perform semantic NER, makes a direct comparison between the NER method introduced in this dissertation and others difficult or impossible.

This data-driven approach to NER follows the trend of organizations such as Google, Yahoo!, Bing and AOL with access to massive amounts of data and the high-performance computing resources that provide solutions to problems that previously required complex algorithmic and statistical models by taking advantage of fast computational speeds to extract information on prior knowledge contained in a vast corpora of relevant information (Anderson, 2008). For example, the use of dictionaries to suggest correct search terms to users have been replaced by using examples of same/similar previous user searches (instead of searching dictionaries for close spelling matches). Google’s translation of languages by finding instances of past translations of the same words and phrases without actually “knowing” any of the language rules (Anderson, 2008) is another example data-driven solution approach.

Another evidence of the trend to this data-intensive solution approach is research into the use of “big data” such as map-reduce (Yang, et al., 2007) that presents effective methods of utilizing huge volumes of data by reducing redundancies and making programs threading relatively easy. Essentially, the easy accessibility to massive amounts of data as well as the increasing availability of the resources and know-how to process it is breeding new data solution philosophies. This can perhaps be surmised by Peter Norvig, Google’s research director’s statement that “All models are wrong, and increasingly you can succeed without them” (Anderson, 2008) a twist on the often quoted statement by George Box that “All models are wrong, but some are useful” (Box, et al., 1987 p. 424).

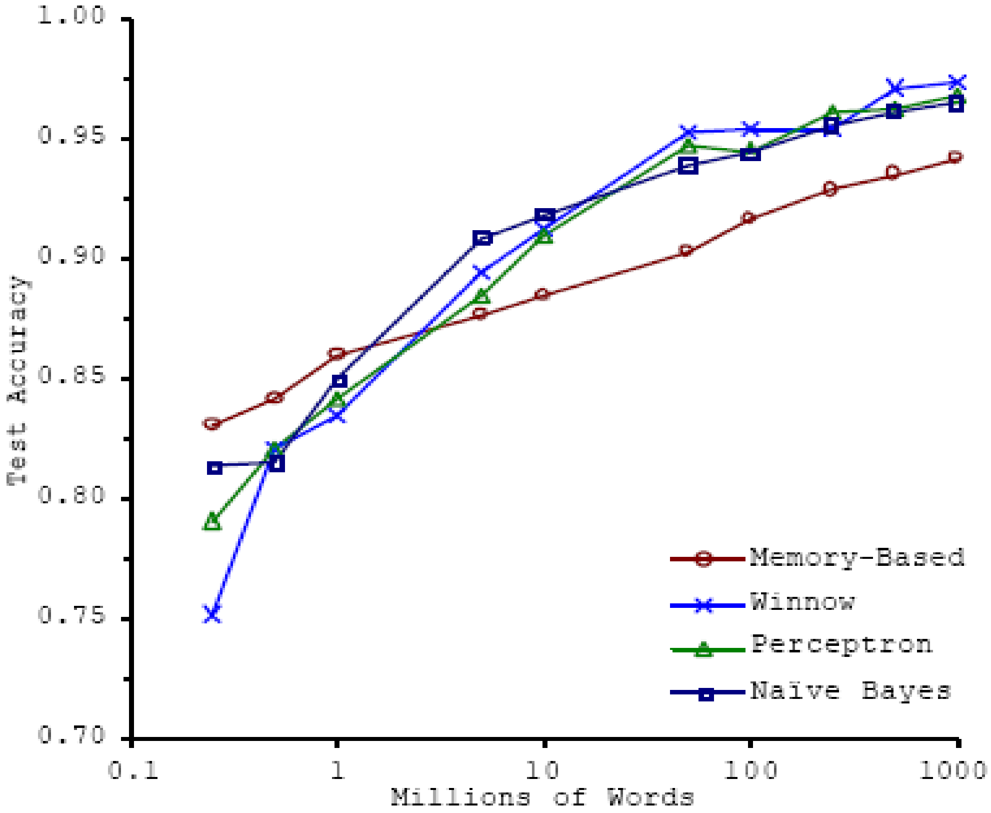


Figure 1.1: Learning curves for confusion set disambiguation

It shows the test accuracy of different machine learning for natural language disambiguation in relation to the amount of words used (Banko, et al., 2001)

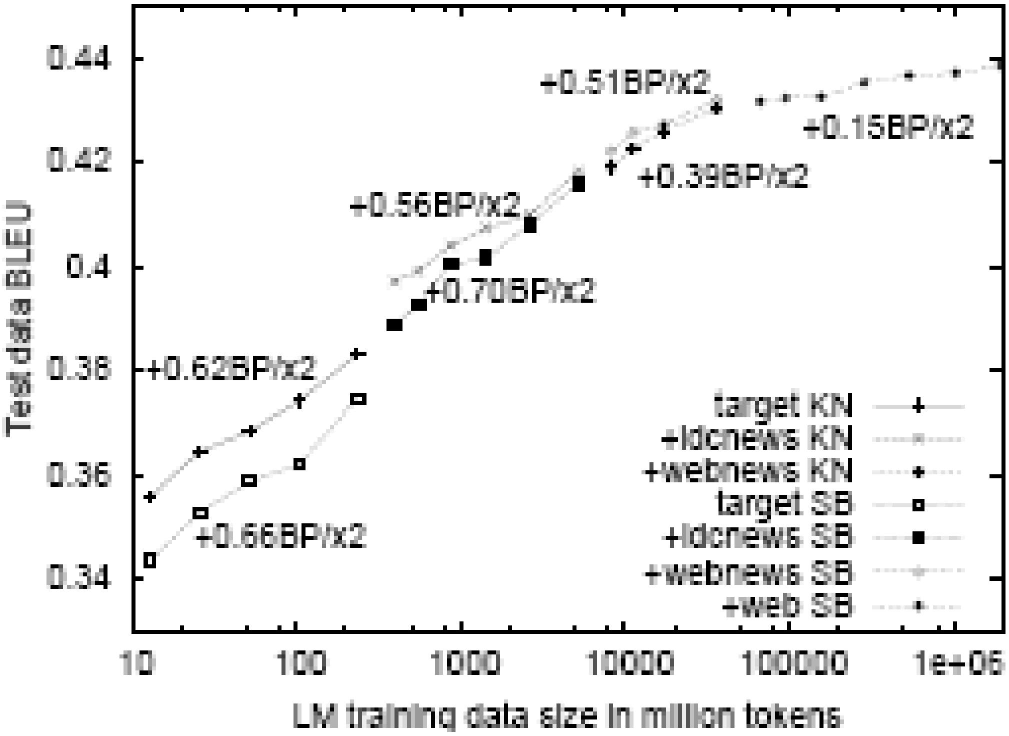


Figure 1.2: BLEU scores for varying amounts of data using Kneser-Ney (KN) and Stupid Backoff (SB) smoothing algorithms.

The numbers above the lines indicate the relative increase in language model size e.g. x1.8/x2 means that the number of n-grams grows by a factor of 1.8 each time the amount of training data is doubled. (Brants, et al., 2007)

1.1. Named Entity Recognition

The term “named entity” was developed during the Sixth Message Understanding Conference as a way of identifying the names of different entities e.g. people, organizations, geographical locations within a text (Grishman, et al., 1996). NER is the process of locating a word or a phrase that references a particular entity in unstructured text. For example in the text “XYZ Inc announced on Jan. 1, 2010 that it has acquired controlling interest in ABC Corp,” entity recognition tasks would identify *XYZ Inc* and *ABC Corp* as organizations while identifying *Jan. 1, 2010* as a date. Semantic named entity recognition is an important aspect of intelligent information extraction and management research (Jiang, et al., 2006); it determines the semantic role of the entity within the context of the document. For example, it would identify *XYZ Inc* as the acquiring organization; *ABC Corp* as the acquired organization and *Jan. 1, 2010* as the acquisition date from the above example. This is especially important when there is the need to distinguish between multiple occurrences of named entities in a text. These roles and relationships are important in deeply understanding the real world meaning of the document with regards to the relationship(s) between/among the entities.

In order to formally represent and share information about identified entities and their semantic meanings among different systems, a specification of common vocabulary, definitions, relations and functions in a universe of discourse called ontology is introduced. Ontology is an explicit specification of a conceptualization (Gruber, 1993). In representing the information obtained from the NER process in a relational table or XML format for instance, the ontology objects could be used as column headings or XML-tags respectively. The NER approach introduced in this disertation allows for user defined ontology, i.e. users are able to define the types of entities and relationships of interest within a corpus.

1.2. NER in Organizational Context

The significant increase in the number of sources and volume of high-value, unstructured information available to organizations in recent years e.g. social media (Levas, et al., 2005) and the need to automatically or semi-automatically derive business intelligence and information contained within them for decision support from the knowledge is promoting new research interests into NER.

Data within an organization can be broadly classified into structured and unstructured data. Structured data is data that comes repetitively in the same format and layout (Inmon, et al., 2008) e.g. bank transactional data. Unstructured data can be defined as data without a conceptual or data type definition (Weglarz, 2004) and exist as bitmap data (e.g. image, audio, and video) and text (character) data. Although unstructured data exist in several forms as described above, the term shall be used to refer only to unstructured textual information (UTI) in this paper unless otherwise stated.

In addition to these data categories existing in different formats, they also often exist in different universes within an organization and are thus accessed, processed and utilized in different environments (Inmon, et al., 2008 pp. 15-31). The analytical processes routinely used by organizations for business intelligence and decision support (e.g. calculation of revenue, profit, market share, key performance indicators) traditionally occur within the structured environment because most of the current computer technologies available for performing data analytics, automated decision-making etc are designed to only work on structured or “flat” data representation (Friedman, et al., 1999). Those available for processing large volumes of unstructured data are complex and inefficient (Raghuveer, et al., 2007) (Ferrucci, et al., 2004) (Bitton, 2006) (McCallum, 2005) (Inmon, et al., 2008 pp. 1-14). The unstructured environment is mostly used for human communications, on-demand search, ad-hoc intelligence extraction etc.

Given that unstructured data constitutes up to 80 percent the data accessible to an organization (Merrill Lynch & Co., 1998) (Goth, 2007), NER can be an effective tool for moving a substantial amount of previously unutilized/under-utilized data into the structured data universe where existing data analytics tools and routine processes can be applied to it. NER can thus lead to a high information quality within an organization in the sense that there is a higher level of organizational data asset utilization; this can produces tangible benefits to the organization and at the same time increase the value of the data asset (Fisher, et al., 2008 pp. 148-150) (Talburt, 2009).

1.3. NER Approaches

Different NER approaches use different parsing methods to identify entities within UTI. The two main parsing methods revolve around grammar-based rules and statistical techniques. The main emphasis of these techniques has been on speed and coverage of parsing while trading depth of analysis to robustness (Bangalore, 1997).

1.3.1. Grammar-based NER Approach

Grammar-based NER approach use manually programmed grammar rules to locate named entities within UTI (Abney, 1996) (Hobbs, et al., 1993) (Klein, et al., 1963). It uses finite state cascade consisting of a sequence of grammar levels and sets of regular expression patterns to parses the text. It often performs shallow parsing, i.e. the outputs are usually in the form of parts of speech (e.g. verb groups, noun groups) and do not convey information about the elements in each group. Although each level of cascade only produces locally optimal output, they are usually deterministic, fast and reliable; they have been successfully used as a preprocessor in many information extraction systems (Hobbs, et al., 1993).

Because some of the grammar rules and patterns used in those methods are language and/or domain dependent, the system needs to be modified in order to be used in a new language or domain environment. This can be relatively expensive considering the level of expertise needed in writing and programming the rules. Furthermore, because the grammar rules are crafted with the assumption that the text to be targeted is grammatically correct and the systems do not use statistical information for disambiguation, the performance degrades when applied to text that contain incorrect grammar, slangs or unusual abbreviations (Bangalore, 1997 p. 6).

1.3.2. Machine Learning NER Approach

Machine learning NER approach (also referred to as statistical model NER approach) automatically extracts language and grammatical rules from manually annotated training corpora (Jelinek, 1990) (Church, 1989) (Bahl, et al., 1976) using machine learning algorithms. This automatic rules extraction enables it to extract rules from large volumes of training corpus and robustly handle grammatically incorrect or ill formed sentences unlike the grammar-based rules approach. It uses patterns and relationships in the annotated training corpora to build statistical models that are used to probabilistically identify and classify nouns into groups such as person, organization, location etc. (also known as deep parsing). Deep parsing unlike shallow gives priority to locating globally optimal outputs over locally optimal ones.

The linguistic rules generated using these approaches are not easy for humans to read thereby making it difficult to modify. For this reason, the system usually needs to be retrained when using new or modified annotated corpora. Also because the rules are coupled with the annotated corpus domain, the system needs to be retrained using corpus from the target domain document family.

1.3.3. Hybrid NER Approach

Hybrid NER methods combine the grammar-based rules with the statistical model NER approaches (Black, et al., 1993) (Mikheev, et al., 1999) (Srihari, et al., 2000). Although this approach combines the strengths of the grammar-based rules and statistical model NER methods, it also inherits their weaknesses e.g. manually programmed grammar rules still have to be modified for different domain and/or languages.

1.4. NER Approach Using Contextual and Intrinsic Indicators

A language and domain independent data-intensive NER approach that also identifies the semantic roles of named entities within unstructured text is presented in this dissertation. This NER approach uses contextual and intrinsic indicators to create entity identification heuristics by inferring the language and domain knowledge information from the characters (and not words) of annotated example corpus (also termed knowledgebase). The sequence of characters surrounding the annotated entities in the example corpus provide it with the language and domain information used in identifying and disambiguating between candidates[1], while the values of the annotated entities serve as a glossary for the different entity type values and are used to determine the likelihood that candidates (values) belongs to particular entity classes.

This approach is motivated by the interest in creating a NER system that does not rely on NLP and not tightly coupled with the document domain or ontology, thereby eliminating or reducing the need for system re-engineering when either environment changes. The envisioned NER system can be deployed seamlessly across different languages and domains in a manner that facilitates a ‘bring your own data and entities’ approach to NER. A key factor considered during this research is the ability of the system to recognize entities using a wide array of linguistic and domain information without the need for manually programming grammar-rules and avoiding the humanly understandable rules

[1] A candidate is a word or phrase identified as a possibly correct entity within a text until it is validated/chosen as correct

derived by complex statistical models and algorithms. The system should also be able to identify entities that require uncommon or rare language rules which are not likely to be hand crafted or possibly suppressed during the statistical rules extraction (due to their statistical insignificance). Ultimately, this NER is aimed at representing UTI in a structured form such that its content and semantic meaning can be understood and analyzed in a similar manner to a structured transactional data. For example representing news articles in a structured format such that one can query for the number of times that “Michael Phelps” has won the “freestyle” competition. The proposed NER approach follows the data-intensive model that scans through a large volume of example documents to find instances of known or similar solutions.

1.4.1. Language and Domain Independent

Different from other NER approaches, the data-intensive NER approach described in this dissertation does not require language or domain specific rules and neither require external linguistic resources (e.g. dictionary and thesaurus) for locating named entities within unstructured text. All the rules, heuristics and probability information used for NER are obtained from the annotated example corpus (also termed knowledgebase-KB). Unlike the statistical model NER approach, this approach does not need to be trained, it infers the information used for each entity recognition process directly from the KB. Also, because the KB is loosely coupled with the NER system, the candidate location heuristics are not dependent on the KB. Also; and the extracted rules and heuristics are transparent and human readable, hence can be easily modified by humans. It differs from the grammar-rule based approach in that it does not require hand crafted rules and the extension of its grammar does not require the skills of an expert computational linguistic; it can be extended simply by adding the document containing the desired grammar rule to the KB.

1.4.2. Semantic Named Entity Recognition

In addition to identifying named entities, this approach performs a high level NER by assigning semantic roles to the identified named entities. For example, given the sentence “Michael Phelps won the freestyle competition against Jason Lewis and Alain Bravo”, this approach identifies “Michael Phelps” as winner and “Jason Lewis” and “Alain Bravo” as competitors instead of simply recognizing them all as proper nouns or people. This can be of particular importance when there is the need to disambiguate among multiple named entities of the same group within a document. It also provides a way of capturing the real-world meaning of the content of a document in a structured form.

1.4.3. Loosely Coupled Language and Domain Knowledge with Entity Extraction Heuristics

Because the extraction heuristics and probability calculations are decoupled from the KB, users have the flexibility of introducing custom ontology, expanding or modifying existing ontology as well as sharing ontology and KB. The semantic role naming feature also provides the flexibility to further reclassify semantically named entities into more general groups e.g. person and location which can then be used for other purposes or in other domains; thereby making the KB re-useable in lower-level entity recognition.

Furthermore, instead of having to improve the NER system independently for different languages and domains, improvements made on the system using one language or domain can be directly transferred to other languages or domains.

1.5. Dissertation Chapters Outlines

This dissertation’s chapters are arranged as described below.

Chapter 2 contains review of related research in NER. Three categories of NER methods starting with the initial approaches to NER that requires manual programming of grammar-rules, to the NER approaches that use machine learning algorithms to create statistical models and algorithms for entity extraction, to systems that use features from both methods were reviewed.

In Chapter 3, the previous NER researches at UALR Information Science department that motivated this research into semantic NER were described. It also covers a data-intensive research that this NER approach draws closely from. Finally, an overview of the data-driven NER approach is presented.

Chapter 4 contains descriptions of the structure of the knowledgebase (KB), the KB fragments as well as the entities characteristics. In Chapter 5, how the KB fragments are used in the candidate location heuristics is illustrated and Chapter 6 details information on how the candidates can be selected as entities based on the ensemble of evidences from its characteristics and the KB. It portrays different methods of calculating the candidate score which is used to compare the candidates. Chapter 7 contains description of how the programming code for the NER was implemented.

In Chapter 8, the set-up of the experiments on the proposed NER was described, and the results and analysis were shown in several charts and tables. Chapter 9 contains discussions, the future work as well as the conclusion.

Chapter 2

Literature Review

Named Entity Recognition is a subtask of NLP that has been extensively studied for several decades. The increasing availability of large volume of high value unstructured textual information (e.g. social media) and the organizational need to automatically analyze and understand it has renewed interests in NER research. The different NER research approaches can be classified into 3 namely: grammar-based rule, machine learning and hybrid.

Early NER research involved the use of finite-state intuitive language and/or domain specific hand crafted grammar-based rules e.g. part of speech (POS), syntactic e.g. word precedence and orthographic features e.g. capitalization (Harris, 1962) (Joshi, et al., 1996). Research has also been done into the combination of linguistic resources such as dictionaries – WordNet (Fellbaum, 1998) and thesauri – Roget's (Emblen, 1970) with such systems.

2.1 Rules-Based NER Literature Review

Church (Church, 1980) is one of the pioneers of the use of finite state grammar for language understanding with the development of YAP. He presented the hypothesis that human exploitation of the complexity of an ideal language is subject to memory limitation, therefore a simplified computation might be adequate as a model of linguistic human performance. Abney (Abney,

1990) presented CASS, a multi-stage parser that uses Church’s POS program as input and uses 3 simple and inexpensive techniques (small question answering, direct error repairs and constituents structuring into major and minor) in the parsing process. Klein and Simmons (Klein, et al., 1963) presented a mechanical coder system capable of assigning POS tags to English word without the aid of a dictionary look-up. Applet et al (Appelt, et al., 1993) proposed FASTUS, a pattern matching based name recognition system that parses sentences into phrases which are pattern recognized by the lexical noun and verb groups. Hashemi et al. (Hashemi, et al., 2003) introduced a NER technique for extracting Names, Titles, and their associations.

2.2 Machine Learning Literature Review

Machine learning approach translates NER tasks into classification problems to which an array of machine learning models and can be applied. They create complex statistical and mostly non-transparent rules for entity recognition. Jelinek (Jelinek, 1990) statistically created sequences for entity recognition using the mutual information between two adjacent words that exceed predefined thresholds. The two types of machine learning models that are used for NER are supervised and unsupervised learning. Supervised NER learning involves the use of annotated/labeled examples to train machine learning models how to classify the features annotated in the examples. On the other hand, unsupervised NER learning does not involve the use of annotated input examples; the machine learning model is given unlabelled/raw data and trained without feedback. It seeks to build data representations and summarize key features of the training data (Mansouri, et al., 2008).

Supervised NER learning can identify and classify nouns into more specific groups like persons, organizations, locations etc.; however it requires a large amount of training data in order to achieve a good performance. Unsupervised learning is not very widely used in NER research, the data representation in its output are more widely used for text categorization (Joachims, 1997) (Srinivasan, 2002) and information retrieval (Kuflik, et al., 2006) (Ohbuchi, et al., 2006).

Supervised machine learning has attracted a lot of attention in recent years and has been extensively used in applications like Markov Chain (Shannon, 1948), Hidden Markov Model (HMM) (Bikel, et al., 1997), Maximum Entropy Model (MEM) (Borthwick, et al., 1998) , Support Vector Machine (SVM) (Mayfield, et al., 2003) and decision trees (Black, et al., 2002). These different supervised learning applications have different speeds and capabilities, for example HMM approaches have very short training times (few seconds) while MEM, SVM and decision trees can be used for contextual information modeling. Memory-based learning also known as “lazy” learning stores the training data in memory and extrapolates the class of most similar memory item(s) to the test item (Tjong Kim Sang, 2002) (De Meulder, et al., 2003) (Hendrickx, et al.).

2.2.1 Markov-Based NER Literature Review

HMMs have been used as probabilistic tools to model sequential data and determine optimal linguistic patterns for NER tasks. Zhou and Su (Zhou, et al., 2002) presented a NER system for names, times and numerical quantities that integrates different evidences using HMM. Nymbel, a name-finder based on HMM was proposed by Bikel et al (Bikel, et al., 1997). Taskar et al (Taskar, et al., 2002) introduced the relational Markov network (RMNs) which can be used in a joint probabilistic model for the classification of multiple related entities. Merialdo (Merialdo, 1994) trained a text tagger by applying hidden Markov process to untagged examples according to Maximum Likelihood principles.

2.2.2 Maximum Entropy NER Literature Review

Maximum entropy model, unlike HMM, can define conditional probability of state sequences of observed arbitrary overlapping features (Jing, et al., 2008).Borthwick et al (Borthwick, et al., 1998) presented the use of maximum entropy in exploring diverse knowledge sources for NER. McCallum et al presented an information extraction tool that uses maximum entropy framework to a set of models representing a state given a couple of observational sequences. Chieu and Ng (Chieu, et al., 2002) presented a maximum entropy based NER that extract named entities using a single classifier. Curran and Clark (Curran, et al., 2003) demonstrated a language independent maximum entropy named entity tagger tested on English, German and Dutch documents.

2.2.3 Support Vector Machine NER Literature Review

Mayfield et al used support vector machine (SVM) to efficiently select the number of features and simultaneously limit overtraining in a lattice-based approach to named entity recognition (Mayfield, et al., 2003). Zhang et al conducted a comparative study between SVM and structural SVM NER approach for extracting names from bibliographic contents (Zhang, et al., 2010). Li et al. (Li, et al., 2009) presented two adapted SVM methods for information extraction; they described the SVMUM – SVM with uneven margin and SVM active learning as ways of dealing with imbalanced training data and the cost of labeled training data respectively.

2.2.4 Decision Tree Based NER Literature Review

Paliouras et al. (Paliouras, et al., 2000) used decision trees to automatically acquire named-entity recognition and classification “grammar” from text data as a way of reducing the amount of required manual customization. Brill’s (Brill, 1995) transformation-based error-driven NER generates a simple rule-based approach to learning of linguistic knowledge by applying ordered transformation list learned from manually annotated corpus to parsed text obtained from an initial-state annotator; his approach offers transformation list even when an equivalent tree does not exist for a set of primitive queries.

2.2.5 Class-Based ER Literature Review

Class-based models are statistical language models that help in generalizing word meaning. It groups words with similar meaning or syntactic functions thereby increasing the features space of each word beyond the words surrounding the entity. Brown et al. (Brown, et al., 1990) presented the n-gram language model and statistical algorithms for assigning words to classes. Ward and Issar (Ward, et al., 1996) proposed a finite state networks approach of expanding words into sequences in a language class model. Maskey et al. (Maskey, et al., 2008) presented a named entity translation approach using syntax-based rules with non-terminals as named entity types.

2.3 Hybrid NER Literature Review

Hybrid NER methods aim to combine the strength of the rule based and machine learning methods. Several such systems have been proposed in the Message Understanding Conferences (MUC) (Mikheev, et al., 1998) (Srihari, et al., 2000) and Conference on Computational Natural Language Learning (CoNLL) (Zhang, et al., 2009) (Hong, et al., 2009) (Bharati, et al., 2009). Fisher et al. (Fisher, et al., 1995) applied several machine learning based components, manual coding as well as linguistic and lexical resources in the information extraction system at the University of Massachusetts. Black et al. (Black, et al., 1998) developed FACILE a rule-based system that uses database lookup of lexical resources but does not use training techniques for knowledge-based categorization of news in four languages (English, German, Italian and Spanish) at the University of Manchester. Miller et al. (Miller, et al., 1998) introduced SIFT; a fully trained information extraction system capable of performing NER tasks using statistical language models trained on annotated data alone.

A key factor in a system's ability to assign names to entities depends on its definition of context in which the entities are extracted. Sliding window methods define the context of the target word in terms of the neighboring words within a limited distance, "window" (Yarowsky, 1992) (Schütze, 1992) (Daelemans, et al., 1997) (Beeferman, 1998). Systems that consider context at the sentential level also known as shallow parsers e.g. CASS, ANNIE, SEXTANT, FASTUS identify entities in general categories without attaching semantic meaning or structure to them (Abney, 1997) (Bontcheva, et al., 2002). On the other hand, systems that consider context at the document level also known as deep parsers e.g. MINIPAR RASP identify entities in a structured manner and attach semantic meaning to them (Lin, 1998) (Briscoe, et al., 2002) (Hobbs, et al., 1993). Chiang et al. (Chiang, et al., 2008) used an extractor (FASTUS) to identify named entities and relationships from the text at the sentential level.

2.4 Collaborative NER Environment

IBM's UIMA[2] (Ferrucci, et al., 2004) and the University of Sheffield's GATE[3] (Cunningham, et al., 2002) are both collaborative open source environments that allows for different NER tools to be integrated and shared among users, developers and researchers. These environments increase the

[2] UIMA - Unstructured Information Management Architecture
[3] GATE - General Architecture for Text Engineering

level of synergy among different NER actors and stakeholders from various language and domain backgrounds. GATE currently contains up to 50 plugins from different NER researchers and developers who make it possible for users and other actors to share experience and re-use modules (GATE project team).

2.5 Summary

Several of the existing NER approaches have achieved remarkable performance. For example Mikheev's (Mikheev, et al., 1999) system achieved a 93.39% performance on the MUC-6 (single language) test and Che's (Che, et al., 2009) system achieved an 83.29% precision in CoNLL-10 multilingual semantic entity labeling challenge. The proposed system introduces a new approach to NER i.e. an approach without the use of hand-coded of grammar rules and/or machine learning. The use of statistical information in this approach is to rank selected candidates in order of likelihood of correctness. Furthermore, the meaning of semantic in NER is extended beyond the generalization of entities as people, location, date etc. to more specific semantic roles within the document e.g. place of birth, bride, groom etc.

Chapter 3

A Data-Intensive Approach to Named Entity Recognition

This chapter describes the motivation for the research approach pursued in this dissertation in addition to related research works preceding this approach. It also introduces the proposed data-intensive approach to NER.

3.1 Motivation to Data-Intensive Approach Named Entity Recognition

Research into the use of open source document for entity resolution, identification, disambiguation as well as updating entity catalogs and proprietary repositories at the Department of Information Science of University of Arkansas at Little Rock (UALR) by Talburt, Wu, Pierce etc. (Talburt, et al., July, 2007) (Talburt, et al., 2007) (Wu, et al., 2007) (Chiang, et al., 2008) (Hashemi, et al., 2002) (Hashemi, et al., 2003) are precursors to this NER research. In order for organizations to effectively take advantage of the massive amount of publicly available UTI (especially on the Internet), it is necessary to apply new tools that work effectively on unstructured text and/or transform the text into structured data (e.g. relational tables) so that existing analytical tools can be applied to it and also that it can be directly comparable to existing structured information within the organization. Some of their research involved the use of online obituary announcements information to identify new entities and append data to existing entities in repositories.

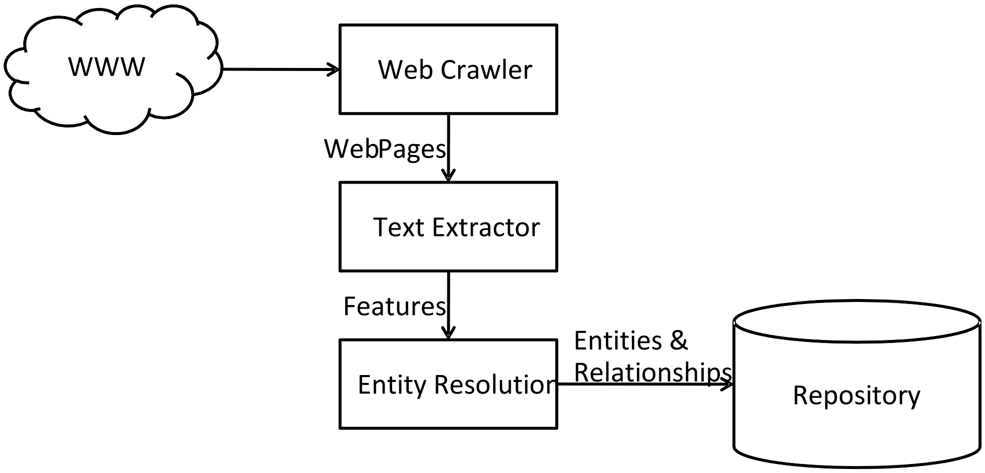


Figure 3.1: Entity identification process (Chiang, et al., 2008)

Their research required not only general identification of entities as persons, location etc within UTI, it was essential to assign their semantic role within the document e.g. bride, groom, bridesmaids etc in a wedding announcement. Several NER techniques and tools were researched and implemented for identifying the people entities; however, standard tools/methods for assigning semantic roles did not exist. They used an extractor (FASTUS) to identify named entities and relationships from the text and then applied hand crafted pattern matching techniques to the text around the identified entities to determine their semantic roles within the document. For example, “HAMPTON − John Doe, 80, of Cantrell Road, died Thursday, Dec. 30, 2004, at Hampton Regional Hospital.” matches the following pattern, “[LOC][NM][AG]?["of"]?[DP]["died"|"Died"|"passed away"|"Passed away"][DD]” where [LOC] – Hampton, [NM] – John Doe, [AG] – 80, [DP] – Cantrell Road, [DD] – Dec. 30, 2004 are the outputs. This technique yielded precision and recall rates of 37.2% and 20.1% respectively when applied to 31 obituary announcements.

The precision and recall rates increased to 71.4% and 41.3% respectively after further language and grammar specific rules were implemented in the pattern matching (Wu, et al., 2007).

Although the approach yielded interesting results, they also discussed some major drawbacks of the approach e.g. the amount of work needed to rework the code in order to accommodate different grammar rules and exceptions as well as its specificity to language and domain. For example the identification of rare names is prone to a high degree of error, the presence of typos, spelling mistakes or certain slangs in the target text results in pattern matching failures. The variations in punctuations (e.g. father-in-law and fatherinlaw) and abbreviations (e.g. AZ and Ariz.) are also among other factors that reduced the efficiency of their NER approach. Furthermore, modifying the code in order to accommodate a new matching pattern may lead to errors in identifying others.

3.2 Preceding Related Research

The data-intensive NER approach presented in this paper is preceded by Talburt and Bell's Bayesian identification of "floating address lines" using only information from an annotated knowledgebase-KB (Talburt, et al., 2000). They automatically classified US postal standard address lines by functions (e.g. individual name, business name, street address, etc.). The classification is done without any semantic knowledge of the words on each line; each address line is assigned a function based on the estimates of conditional probability distribution of its words with reference to a large corpus of addresses where each line was expert-coded as one of 7 functional types. The function of a line in the target address is inferred by looking up each phrase in the line in the frequency table (compiled from the KB) and accumulating a score for each of the possible types for each line in the address. A simple analysis of the line-by-type scoring matrix is used to assign a type assignment for each line.

They used a corpus compiled from 100,000 expert-coded addresses and tested it against 23 million addresses. Their approach yielded identification accuracy increase of 7% compared to the original, rule-based and language specific production system which used only a few small generic name tables. Inefficiencies in the system such as poor performance in identifying names recorded in last-name-first order, and city-state lines that did not have zip codes was easily rectified by adding more data (that contain examples of instances of poor performance) to the knowledgebase corpora. This form of system improvement method is cheap because it does not require program code modification. Furthermore, the likelihood that the process of improving performance to one class of entity would lead to regression in the performance of another is greatly reduced.

Their approach assumes that each address in the target data was already organized into a list of discrete address lines, and that each line could be classified into one of the seven line types. However when dealing with free text, there may be no distinction between lines, any phrase potentially represents an entity reference and there is no assumption that any entity is contained in the text; all these makes the extraction and identification problems more complex.

3.3 Overview of a Data-Intensive Approach to Named Entity Recognition

A data-intensive NER approach that does not require language specific rules, lexical resources or machine learning for locating and assigning semantic roles to entities within UTI is proposed. This NER system obtains all the required information for entity extraction from annotated example corpus (also known as knowledgebase) containing the same/similar document family as the target document. The entity location does not involve linguistic or domain specific heuristics hence it can perform seamless across different languages and domains. These heuristics also do not require probabilistic algorithms and are human readable as well as easily extended by adding new documents to the knowledgebase (KB). Because the entire data in the KB is used and not abstracted using statistical/probabilistic methods, the increase in the amount of unique example documents available is directly proportional to the number of opportunities for making inferences hence the system performance.

The proposed data-intensive NER approach features the use of KB(s) from which the language, domain and ontology information are extracted, methods for analyzing the properties of entities annotated in the KB, heuristics for locating candidates and candidate scoring methods all of which are loosely coupled (see Figure 3.1). This NER approach identifies candidate boundaries and semantic roles by direct character-to-character comparisons with boundaries and strings surrounding annotated entities in the KB. It uses the knowledge that particular strings surround an entity class in the KB to recognize instances of the same entity type in the target documents. It therefore assigns entity class to candidates based on the entity class associated with the surrounding strings in the KB. The string value of the candidate is then compared with those of the same entity class in the KB (e.g. the string value of a bride candidate is compared with all the values of bride entity annotated in the KB). If enough evidence of same/similar characteristics of the candidate is found in the KB, then it is considered as a possible entity. A score based on the ensemble of all the candidate characteristics is used for ranking and filtering the most likely correct candidates.

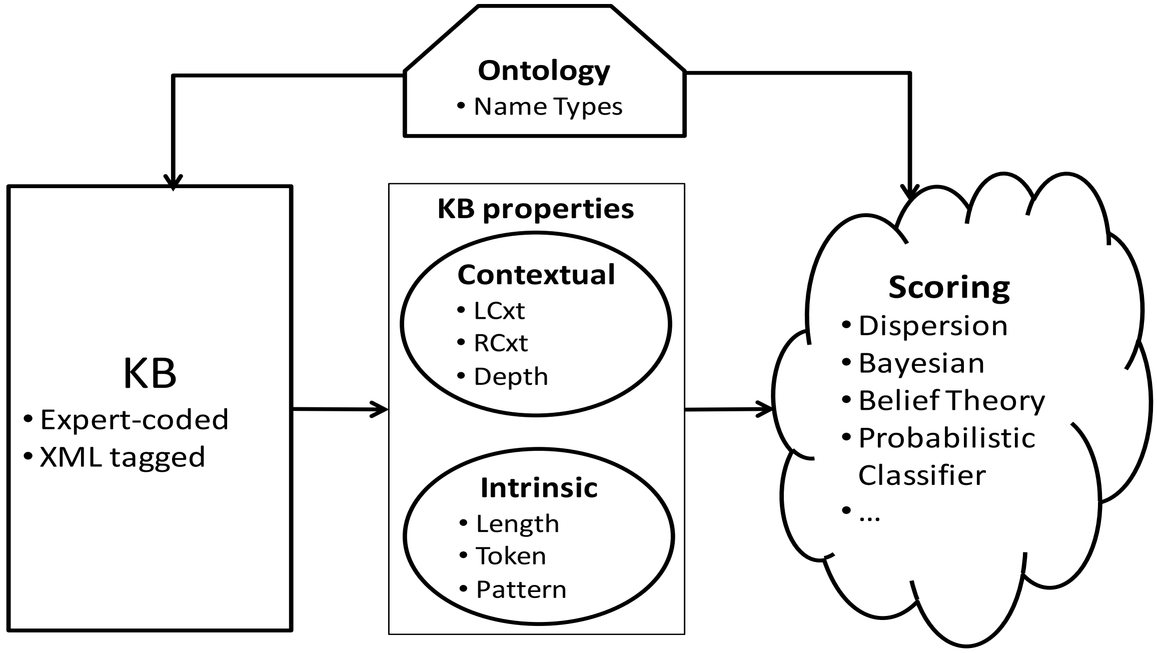


Figure 3.2: Overview of a Data-Intensive Approach to NER

This approach differs from the previous NER approaches in that the system neither requires handcrafted grammar rules nor use probabilistic methods to decide how an entity should be located or what semantic role it is assigned. Furthermore, it does not use any prior domain or language knowledge other than the distinction between alphabet, numeric, space and other characters[4]. A basic character-to-character matching is performed using all the available data in the KB and not abstracted/statistical sample. This approach is considered data-intensive because the more examples of data available, the greater the opportunity for finding attributes that can be used for entity location or instances for which the same/similar solution is known. As this approach requires a large volume of data and the computing resources to rapidly navigate through them, it is natural that it would not have been a viable approach decades ago when high performance computers were relatively very expensive and data was not ubiquitous as is the case today.

[4] We assume that in every language, there is a distinction between alphabet, numeric, space and other characters. In the case that a language does not have these distinctions, we have no information on how our NER system would perform.

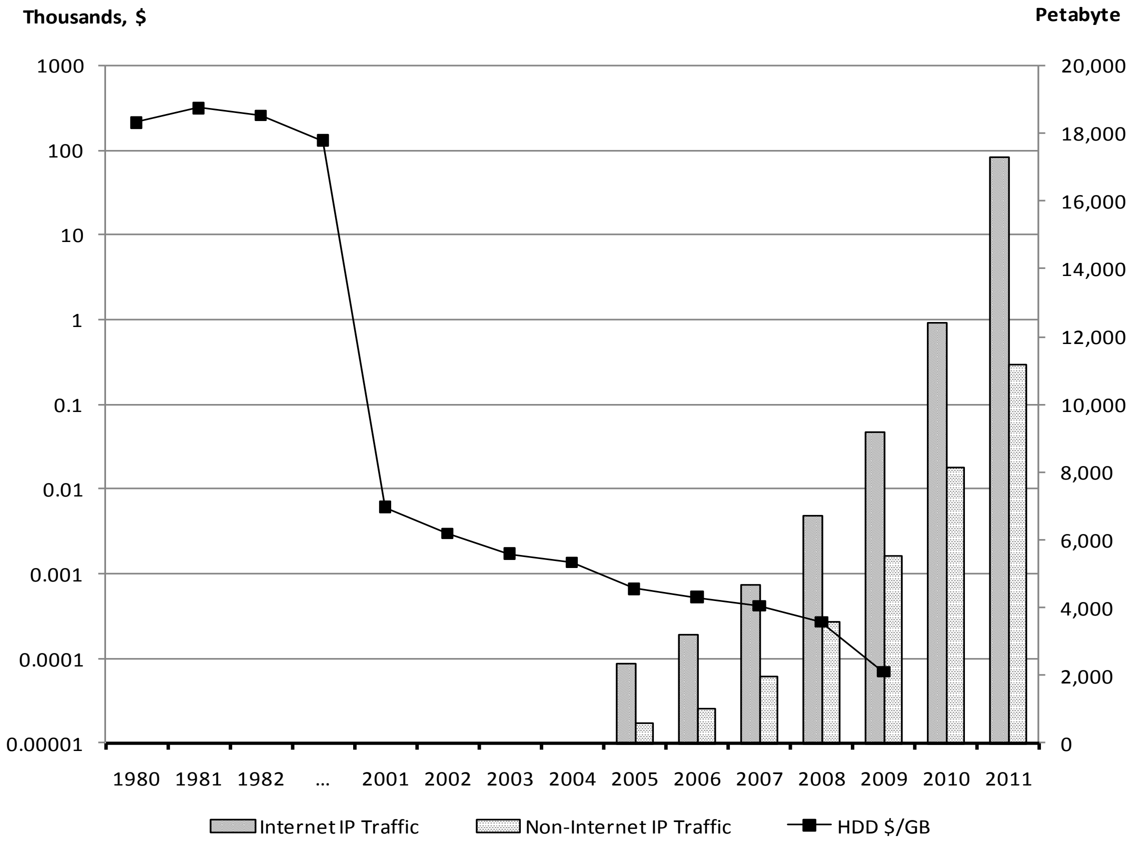


Figure 3.3: Global IP Traffic and Hard Drive Cost per Gigabyte since 1980. Global IP traffic (Cisco Systems, 2007) estimated increased between 2005 and 2009 is about 400 percent while the estimated decrease in the cost of hard drive storage (Nova Scotia) is about 90 percent in the same period (Kurzweil). The amount of IP traffic is estimated to have grown by over 800 percent between 1980 and 2011 and cost of hard drive fallen about 100 percent since 1980 (see data in appendix Table A-1 and Table A-2).

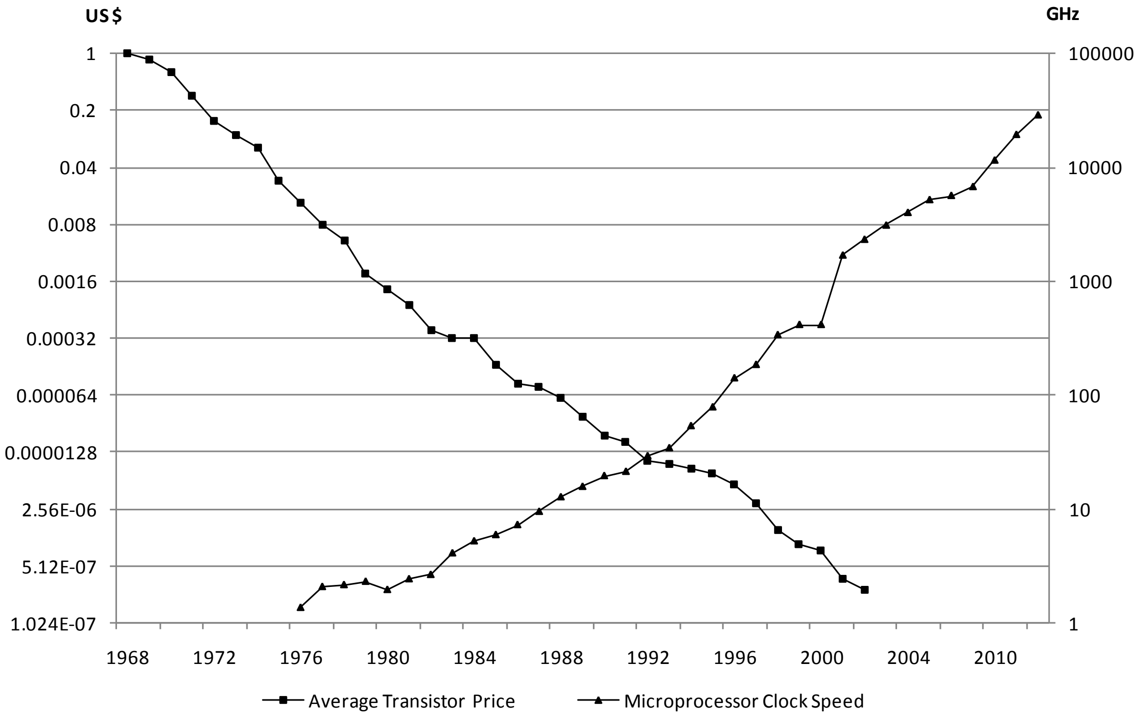


Figure 3.4: Cost of Microprocessor compared with its speed 1968-2010.

The microprocessor speed has increased by over 2GHz since 1976 while the price has fallen from US$1 in 1980 to less than a hundredth thousand of a dollar (Kurzweil). In 2003, the Itanium processor contained 410 million transistors (Intel Corporation). At the cost of about US$110 in 2003, it would have cost US$410 million in 1980.

The purist implementation of this NER approach would require massive amount of data and computational power which are usually out of the reach of individuals or small organizations. This form of implementation requires absolute evidence of all candidate characteristics in the KB in order for it to be considered as possibly correct. For example if the string “Little Rock” is identified in the test document as a location, but the KB does not contain the exact same string annotated as a location, the system would not accept it as a candidate even if it has exactly the same surrounding strings as in the KB. This implies that the KB must contain the annotated names of every possible location it can be used to extract. Even for organizations with the required resources, the purist implementation might be too extreme and not provide an efficient use of resources. An advantage of this form of implementation is that the use of any statistical information can almost be eliminated in choosing the most likely correct candidates; frequency counts may be enough to establish a reliable confidence measure for the correctness likelihood of candidates.

For individuals and organizations without access to massive amounts of data and computing resources or that do not wish to make such amount of investments in a NER tool, it is still possible to apply this data-intensive NER approach on a limited data and computing resources budget. The economic implementation uses simple statistical methods to establish reasonable ranges for the different entity properties. In the previous example with “Little Rock” as a candidate belonging to the location entity class, if the entity values of location in the KB indicate for instance that the length of a city name is between 5 and 15 characters and that it has between 1 and 2 words, “Little Rock” would be included among the candidates to be considered as possibly correct. As will be described in later chapters, the use of such simple statistical information can significantly bridge the performance gap between the purist and economic implementations.

In either case, all the operations required for NER are relatively simple and can be understood without advanced mathematics education. The experiments (also described in a later section) were performed using the economic implementation of the system. In the following three chapters, the different components of the proposed NER system namely the knowledgebase, candidate location heuristics as well as the candidate scoring and filtering methods are described in terms of their features and functions within the system.

Chapter 4

Knowledgebase

In this paper, the annotated example corpus from which the NER system obtains information for locating and assigning semantic roles to entities is referred to as the knowledgebase (KB). Arguably, the robustness of the KB in this data-intensive approach is much more critical to system performance relative to other NER systems for three reasons:

i.) Unlike the machine learning approach, it does not abstract the KB using statistical or probabilistic methods, hence it is more difficult for it to summarize or generalize conditions for entity extraction. Whatever information is not explicitly found in the KB is not used for NER.

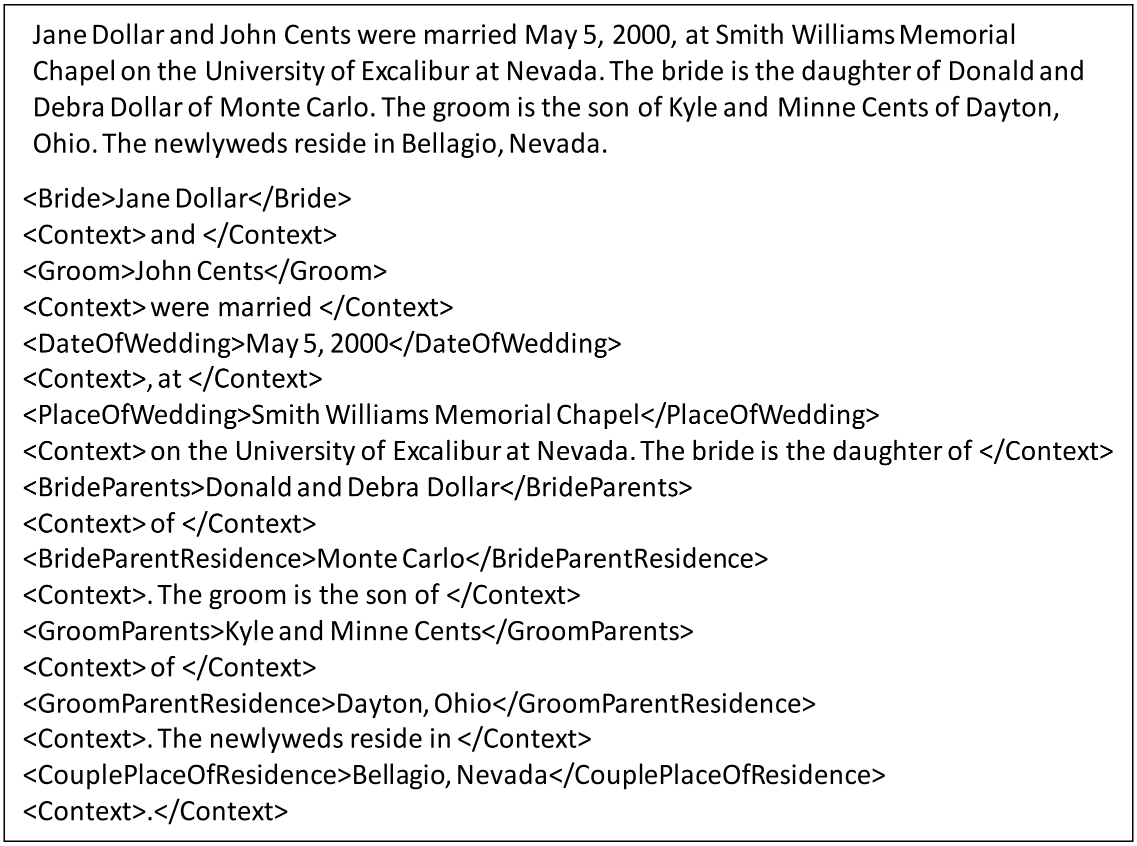
ii.) In contrast to grammar-based approach, it does not use hand written codes which could provide it with information for recognizing entities for which examples cannot be inferred from the KB.

iii.) Similar to the second reason, it does not use lexical resources (e.g. dictionary, thesaurus and gazette) from which it could obtain information not contained in the KB.

For these reasons, the KB can be considered the “brain-box” of the system. The performance of the system is largely dependent on the robustness of the KB and the level of similarity between the family of documents it contains and the target documents e.g. a KB containing wedding announcements can be expected to perform poorly on target documents about car advertisements. The annotated objects in the KB can be indicated using different formats such as XML tags and indexed catalogs. In this dissertation, the annotation is described from the perspective of XML tags representation.

4.1 KB Structure

The KB is a collection of documents annotated by a person(s) with knowledge of document language and domain. It generally indicates strings representing entities of interest and all other strings in the text. Figure 4.1 is an example of a wedding announcement KB document.



Jane Dollar and John Cents were married May 5, 2000, at Smith Williams Memorial Chapel on the University of Excalibur at Nevada. The bride is the daughter of Donald and Debra Dollar of Monte Carlo. The groom is the son of Kyle and Minne Cents of Dayton, Ohio. The newlyweds reside in Bellagio, Nevada.

```
<Bride>Jane Dollar</Bride>
<Context> and </Context>
<Groom>John Cents</Groom>
<Context> were married </Context>
<DateOfWedding>May 5, 2000</DateOfWedding>
<Context>, at </Context>
<PlaceOfWedding>Smith Williams Memorial Chapel</PlaceOfWedding>
<Context> on the University of Excalibur at Nevada. The bride is the daughter of </Context>
<BrideParents>Donald and Debra Dollar</BrideParents>
<Context> of </Context>
<BrideParentResidence>Monte Carlo</BrideParentResidence>
<Context>. The groom is the son of </Context>
<GroomParents>Kyle and Minne Cents</GroomParents>
<Context> of </Context>
<GroomParentResidence>Dayton, Ohio</GroomParentResidence>
<Context>. The newlyweds reside in </Context>
<CouplePlaceOfResidence>Bellagio, Nevada</CouplePlaceOfResidence>
<Context>.</Context>
```

Figure 4.1: Example of a marked-up wedding announcement

Based on the type of annotation, the contents of the KB can be classified into two broad categories namely entity and context.

- Entity: a string(s) representing information of interest.
- Context: string(s) surrounding an entity.

A knowledgebase fragment is the combination of an entity and its contexts on both sides. It is the basic unit by which the entity location heuristics transact with the KB. In other words, candidates are located and assigned a semantic role using the information contained in a fragment. Information about the boundaries (both sides) of a candidate as well as its semantic role(s) is contained in the contexts while the information about the character composition of the entity class is contained in the entity value.

```
<Context> were married </Context>
          <DateOfWedding>May 5, 2000</DateOfWedding>
<Context>, at </Context>
```

Figure 4.2: Example of a fragment from Figure 4.1

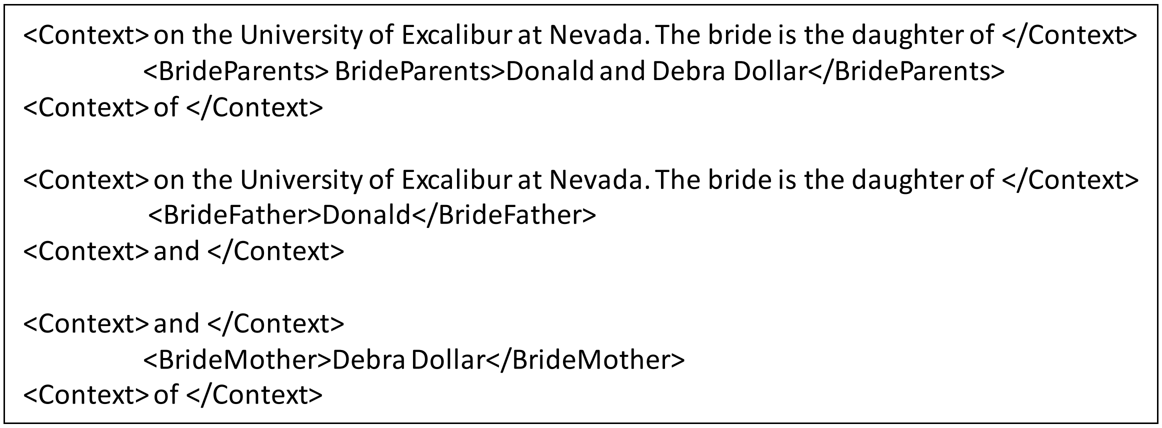
Because fragments require 2 contexts in order to identify the boundaries of a candidate, every entity in the KB is required to be surrounded by contexts on both sides. Hence, the use of filler is introduced to represent contexts in documents where the first or last strings of a document represent an entity e.g. entity Bride in Figure 4.1. Although these fillers might not have any semantic meaning, they denote the important information that the entity is located at the start or end of the document. Therefore, given *n* number of entities in a KB document, there must exist (*n+1)* contexts.

The use of nested tagging system to indicate different semantic roles or to group a set of entities can also be handled by this system. In Figure 4.1 for example, the entities BrideFather and BrideMother may simultaneously be considered part of the string that constitute the entity BrideParents.

```
<Context> on the University of Excalibur at Nevada. The bride is the daughter of </Context>
<BrideParents>
		<BrideFather>Donald</BrideFather>
		<Context> and </Context>
		<BrideMother>Debra Dollar</BrideMother>
</BrideParents>
<Context> of </Context>
```

Figure 4.3: Example of nested entity tags

The system creates 3 different fragments from the above nested tags and can search for individual entities independently e.g. even if the BrideParents entity cannot be recognized; it may still be possible to recognize the BrideMother entity.



```
<Context> on the University of Excalibur at Nevada. The bride is the daughter of </Context>
            <BrideParents> BrideParents>Donald and Debra Dollar</BrideParents>
<Context> of </Context>

<Context> on the University of Excalibur at Nevada. The bride is the daughter of </Context>
             <BrideFather>Donald</BrideFather>
<Context> and </Context>

<Context> and </Context>
            <BrideMother>Debra Dollar</BrideMother>
<Context> of </Context>
```

Figure 4.4: KB Fragments from nested entity tags in Figure 4.3

Although this tagging flexibility allows users to customize the KB for different entity types, the use of a foreign KB requires some knowledge of the tags usage (metadata). For instance, a travel journalist may identify the entity class “place” as a geographic location while a poet may identify the same entity class as a state of mind within the same family of documents. The availability of metadata on the other hand increases the ease of reusability of a KB. Because of the semantic roles (high level) assigned to the tagged entities, they may be used to extract more general (lower level) entity types. For example, the Bride, Groom, BrideParents and GroomParents may be generalized as people and used within a larger concept to extract such entities.

The domain information encoded in the KB using the entity tags is directly translated into the ontology to be used by the system in locating and structuring the entities references. For instance, the number of occurrence of entity class A in each of the KB document could be used in guiding the entity location heuristics on when to terminate the search for the particular entity type within the target document. The ontology also provides the database architecture framework for storing the recognized entities in a structured format (e.g. relational database).

As can be deducted from the descriptions above, the KB is structured in such a way that is loosely coupled with the rest of the NER system, thus it allows users the flexibility in designing their individual KBs, reuse and share them (given that the tags metadata are available).

4.2 Entity Properties

The entity properties are the characteristics that were defined for the annotated entities in the KB for the purpose of systematically locating, validating and comparing candidates with one another as well as with KB entities. These entity properties are divided into two categories namely contextual and intrinsic properties.

4.2.1 Contextual Entity Property

Contextual entity property refers to the characteristics of the environment in which an entity is located within a document. It describes an entity relative to the words (strings) that surround it as well as its starting position relative to the document length. This property is measured using two main parameters – context and depth.

4.2.1.1 Contexts

Contexts refer to the string(s) surrounding the annotated entities on both sides. The system leverages the contexts as its dictionary and infers all the language and domain information needed for the NER tasks from it. Examples of contexts are depicted in Figure 4.1 (XML tagged as “context”). They are usually used to determine the boundary locations and semantic roles of entity references. They are further classified based on the side of the context on which they occur:

- Left Context (LCxt): the context preceding (before) a labeled entity in the KB
- Right Context (RCxt): the context following (after) a labeled entity in the KB.

4.2.1.2 Depth (Dpth)

The depth indicates the start position of the entity within the document relative to the document size (number of characters).

$$\text{Dpth} = \frac{\text{index position of entity value in KB document}}{\text{number of characters in KB document}}$$

4.2.2 Intrinsic Entity Property

Intrinsic entity property is the characteristics of the annotated entity value. It is used to describe characters that compose the entity value. It serves as a glossary from which the NER system infers information about the known values of different entity types. It provides further information for disambiguating candidates after they have been extracted using the contextual property. This property is measured using 3 parameters – length, token and pattern.

- Length (Leng): the number of characters in the entity value
- Token (Tokn): the number of word(s) in the entity value (i.e. the number of spaces plus one)
- Character Composition (Char): the differentiation between the alphabetic, numeric and other characters that compose the entity value.

  Three methods were explored for evaluating the character composition.

  i.) The characters contained in the entity value are considered regardless of order.

  ii.) This method transforms each of the alphabetic and numeric characters in the entity value into a single alphabetic and numeric representation respectively (e.g. Jan. 18, 2010 is transformed into Aaa. 99, 9999) and considers them in the exact order.

  iii.) In this method, the entity value is transformed in the same manner as the second method, however the count of the alphabetic, numeric and punctuations are considered regardless of order.

These character composition evaluations were developed keeping in mind that target documents may not have the exact value as the KB entities. These terms and parameters are later used when describing the entity location heuristics and candidates scoring in the following chapters.

4.3 Feasibility of a Functional KB

A few decades ago before corporations like Google, Yahoo!, Bing and AOL with massive amounts of data introduced services and solutions based “big data” approach, the thought of building a suitable KB for the proposed NER approach might have seem absurd. Nevertheless among the initial thoughts when attempting to annotate a KB big enough such that it contains enough example documents that can be used for NER (within a family of document) without the aid of external lexical resources, machine learning techniques or handcrafted grammar based rules is the feasibility of such an endeavor. Because the richness and flexibility of natural language allows for a single document to be written in several ways while still portraying the same meaning, capturing all these variations using a finite set of documents is a daunting task. Furthermore, if the required KB size (number of documents) for obtaining a good performance is infinitely large, then the task might be impossible or uneconomic, hence this approach to NER not implementable.

The need for the proposed NER approach to function efficiently using a relatively small KB is of particular importance to small scale organizations without resources comparable to some of the organizations previously mentioned. Although a definitive empirical evidence of such a KB (due to limited resources) is not provided, methods of efficiently utilizing the KB data such that the critical information for performing NER tasks on a large number of test documents can be obtained from a relatively few documents is described and demonstrated.

Furthermore, avenues for getting example documents labeled considerably low cost are discussed.

Because, the required information for locating and validating candidates are obtained from individual KB fragments, the requirement that the KB must contain documents virtually identical to the target document may be reduced to the requirement that it must contain the same fragments in the KB. Figure 4.5 and Figure 4.6 show proportions of unique left and right contexts of the “decedent” entity type in a KB of 100 non-identical obituary announcements respectively. By treating the each announcement as fragments, the 100 unique documents all share 28 LCxts and 22 RCxts for the decedent entity class. In other words, it is possible to locate the decedent entity type from all 100 announcements using the 28 and 22 unique LCxts and RCxts respectively. Further examination reveals that only 7 unique LCxts account for 20% and 6 unique RCxt account for 82% of all the decedent LCxt and RCxt respectively.

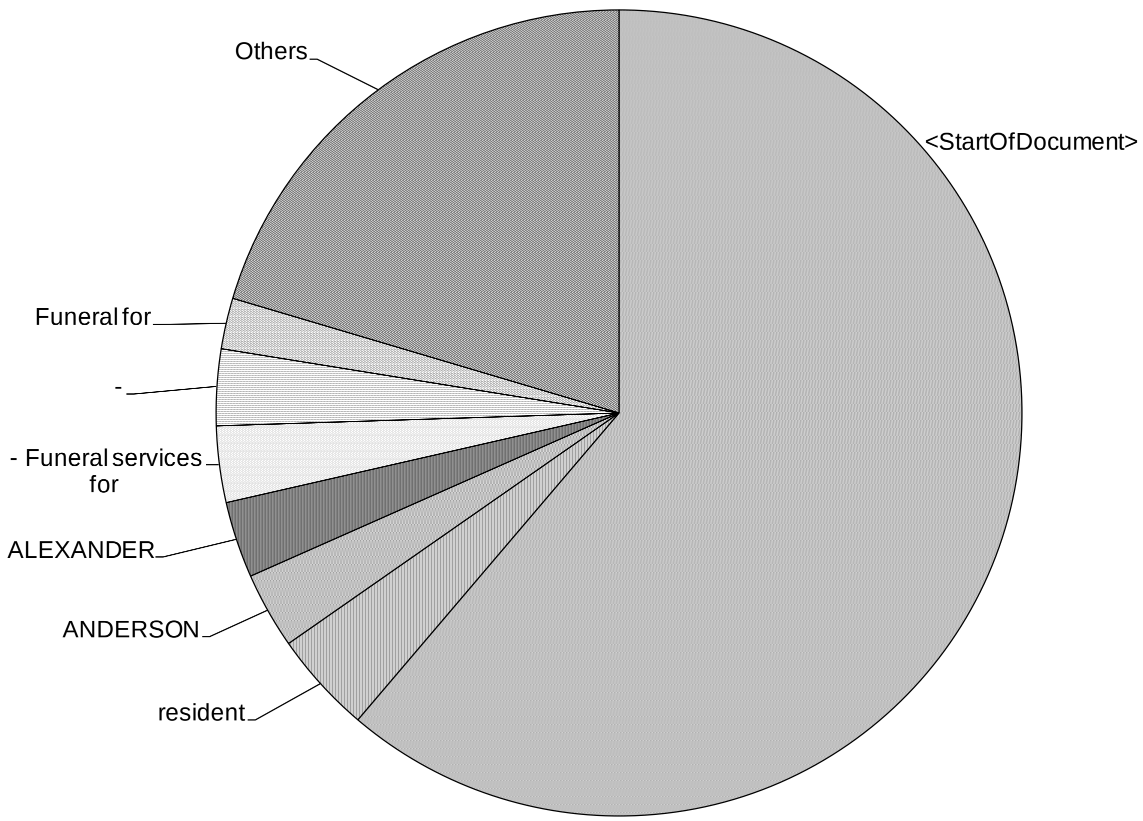


Figure 4.5: Proportions of unique left contexts of “decedent” entity type in a KB of 100 obituary announcements.

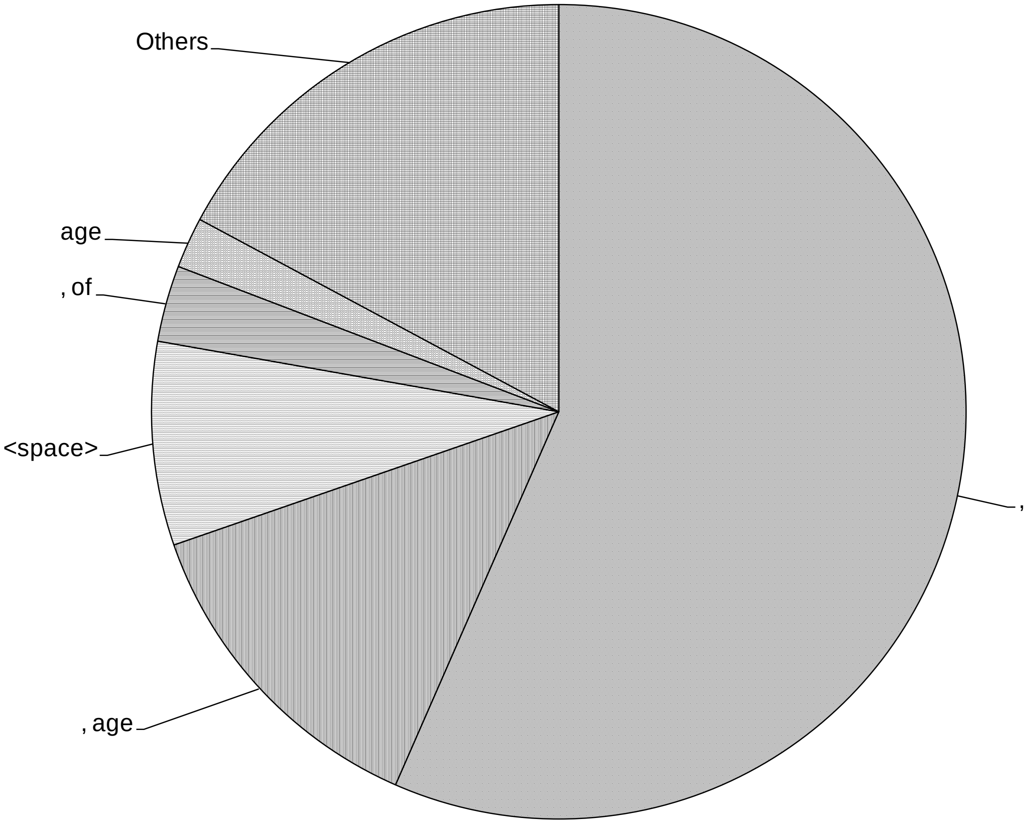


Figure 4.6: Proportions of unique right contexts of “decedent” entity type in a KB of 100 obituary announcements.

Furthermore, by introducing the technique of using the contexts of fragments of the same entity types interchangeably, target documents do not need to contain exact KB fragments but only a combination for an entity type. This implies that the number of opportunities of using *n* unique KB fragments of entity type *α* for locating entities in target documents is n^2.

$$\left.\begin{matrix} LCxt_1 \\ LCxt_2 \\ LCxt_3 \\ \dots \\ LCxt_{n-1} \\ LCxt_n \end{matrix}\right\} EntityType_{\alpha} \left\{\begin{matrix} RCxt_1 \\ RCxt_2 \\ RCxt_3 \\ \dots \\ RCxt_{n-1} \\ RCxt_n \end{matrix}\right.$$

$$n\ fragments_x \xrightarrow{usage} n^2\ fragments_x$$

Figure 4.7: The Multiplier Effect of using Contexts (of the same entity type) Interchangeably

In addition, the technique of partial context matching which would be further discussed in the next chapter is introduced in order to avoid searching for the entire characters of very long contexts. By carefully matching the appropriate portions of the KB fragments with target documents, it is possible to quickly arrive at a KB size (number of documents) where there is a diminishing return on the inclusion of new documents. By treating the KB documents as fragments and by allowing the contexts of fragments of the same entity types to be used interchangeably, it is possible to use a relatively small KB size (number of documents) to recognize entities from a relatively large test dataset.

Typically, anyone with a good knowledge of the document language can annotate the example documents unless they contain entities very particular to a subject domain. Even in those cases, it might still be cheaper to train less skilled people to perform the tagging compared to having an expert perform the task. Services such as Amazon Mechanical Turk (Amazon.com) which enables easy access to crowd sourcing for performing human intelligence tasks at a relatively much lower cost compared to a subject domain expert or computer programmer makes the compilation of a robust KB within the reach of small organizations.

Chapter 5

Candidate Location Heuristics

Candidate location heuristics refers to the systematic processes of using the entity contextual properties to find candidates within target documents. As previously mentioned, the techniques used to locate candidates consider fragments as sequences of characters and not words (does not know the semantic meaning of the words). Because it does not use the semantic information that may be contained in the words it can perform efficiently seamlessly across different languages and domains. On the other hand, a KB can only be expected to perform well only if the target documents belong to the same family of documents contained in the KB.

The candidate location heuristics leverages the contextual properties as its lexical resources and infers all the language and domain information needed for candidate location from them. It is based on the research that two documents are similar depending on the number of words they have in common (Salton, et al., 1983) which has been extensively used in information retrieval tasks. The application of this concept in the proposed NER system is that, two entities are similar or equivalent depending on the number of words their contexts have in common. Therefore if the region of the target document corresponding to the entity depth is searched for the LCxt and RCxt such that there is a string between them, that string is designated as a candidate of the same entity class as that the

KB entity surrounded by the contexts. This process is repeated using all the available fragments to extract every possible candidate from the target document(s).

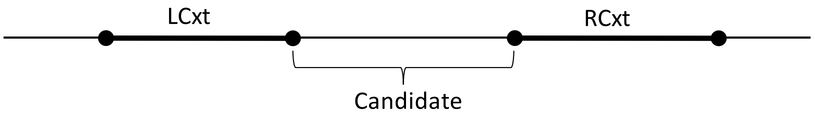


Figure 5.1: Relative locations of the contexts and candidate in a test document.

For example if LCxt=“ and ” RCxt=“ were married ” was found in a document containing “Olivia Oyl and Popeye Sailor were married in Habor Chapel…”. The string “Popeye Sailor” is identified as a possible candidate of groom entity class.

5.1 Context Depth

The context depth is a parameter used to measure of the region of the target document relative to the KB document where particular entities occur. Although entities may be located sporadically in some documents in which case the depth parameter would be of little/no use in disambiguating among entities, this parameter can be very useful if the occurrence of entities with ambiguous contexts are generally located within a particular region of the document e.g. first 20 percent of the document.

5.2 Context Matching Method

Since in the purist implementation of this NER approach, the entire LCxt and RCxt are to be matched absolutely, the method with which they are matched against the target document matters very little as long as they are matched in the appropriate orientation (i.e. relative to each other and the candidate value) and there is a string to be considered as a candidate in the middle (Figure 5.1). However, if the KB is not very large finding an exact long context might be impossible. Thus a method of context matching that allows for flexibility in performing the character matching procedure especially if a context cannot be matched in its entirety is introduced.

5.2.1 Partial Context Matching

One of the challenges encountered during this research was the handling of long contexts for which a direct character-to-character match in the target document is not likely to exist or would require a substantial portion of the target and KB documents to be identical (which is not usually feasible). The proposed solution to this challenge involves using only a portion of the context during the matching procedure; it is termed partial context matching. In deciding the portion of the context to match, Schütze's (Schütze, 1992) research in which he assigns meaning to a context based on the set of words that occur in proximity was subscribed. Schütze extended the information retrieval assumption that two documents are similar based on the number of words they have in common (Salton, et al., 1983) by stipulating that words that occur in close proximity within a context (say a window of 50 words) can be used to represent the context. This is extended further by hypothesizing that words that occur in closest proximity to an entity contributes more to its semantics role(s) than farther words.

Therefore the partial context matching is started from the characters closest to the annotated KB entity. For example, in Figure 5.2 the context matching is performed in the direction of the arrows. The minimum number of characters to match during the partial context matching is a parameter that is determined by the user. The higher the minimum length required for a partial context, the higher the precision will be at the expense of recall and vice versa. If the number of minimum characters set by the user is matched before a mismatch occurs, the partially matched context is accepted and the candidate assigned an entity class otherwise it is discarded.

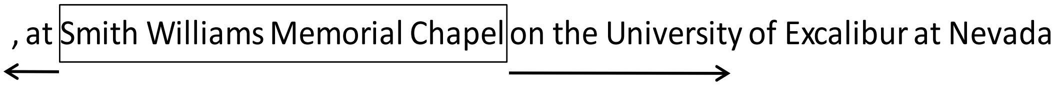

, at
Smith Williams Memorial Chapel
on the University of Excalibur at Nevada

Figure 5.2: Context Matching Directions

When using partial context matching, it may be useful to ensure that the partial context does not end in the middle of a word. For instance in Figure 5.2, that the partial context is not “ on the Un” as this increases the chance of context ambiguity since there may be another context with equal string value that references a different entity type in the KB. It is however also possible that the entire context or partial context that does not end in the middle of a word may be ambiguous (i.e. associated to a different entity type in the KB), this problem is addressed under context disambiguation below. The determination of the ambiguity of a partial context that ends in the middle of a word would require a time consuming iterative scan of portions of the all the KB contexts that can be avoided for performance speed.

The use of partial context matching also helps in extending the reach of the KB as that the number of test documents in which a partial context could be found can be expected to be greater than those in which the entire context can be found. For example in Figure 5.3, the decedent entity type in an obituary document could be located with the partial context “ service for “ in six different documents with different left contexts for decedent. A major drawback of the partial context matching as previously mentioned is that it could introduce context ambiguity into the system. It is nevertheless an effective method of implementing this NER approach on an economic scale.

$$\left.\begin{array}{c}\text{"Funeral service for"}\\[1em] \text{"Graveside service for"}\\[1em] \text{"Memorial service for"}\\[1em] \begin{array}{l}\text{"There will be a private}\\ \text{family service for"}\end{array}\\[1em] \text{"Burial service for"}\\[1em] \text{"Final service for"}\end{array}\right\}\text{Decedent} \Rightarrow \text{"service for"}\}\text{Decedent}$$

Figure 5.3: Context Matching Approximation for the left context of decedent

5.2.2 Context Usage Methods

Other than the one-to-one usage of the KB contexts i.e. attempt to use n number of contexts pairs to locate entities n number of time, two other methods of context utilization were explored.

5.2.2.1 Using LCxt and RCxt Interchangeably

As briefly mentioned in chapter 4, the usage of the left and right context interchangeable increases the number of fragments contexts for matching by a power of 2 thereby significantly increasing the number of target documents from which the KB can extract candidates. To avoid repetition, the reader is referred to section 4.3 of this dissertation.

5.2.2.2 Building probabilistic classifiers using the contextual properties.

From a probabilistic classifier perspective, the process of finding a candidate using the contextual properties parameters can be described as a decision tree i.e. a sequence of decisions. Although the use of machine learning algorithms for NER is not in the scope of this NER research, a probabilistic classifier can be used to determine statistically efficient sequences in which the different parameters can be used for candidate location (Figure 5.4).

This method is implemented using the C4.5 classifier[5] (Quinlan, 1992). The J48 class which is an implementation of C4.5 in Weka® an open source machine learning software (The University of Waikato) was included in the NER application for this purpose. The contextual properties decision tree was built

[5] An ID3 classifier would perform similarly.

using the contextual properties parameter values, and was traversed in order to determine the heuristics sequence for extracting a candidate. One of the most immediate observations with this method is that statistically insignificant parameters are not used in building of the decision tree. For example, in Figure 5.4, the decision tree does not include the RCxt in the entity location process. The elimination of such properties may make it impossible for the decision tree rules to successfully identify candidates e.g. inability to determine the candidate value boundaries. For decision tree branches where both the LCxt and RCxt were not used, the missing parameter is artificially introduced by adding the missing parameter from all fragments of the same entity class.

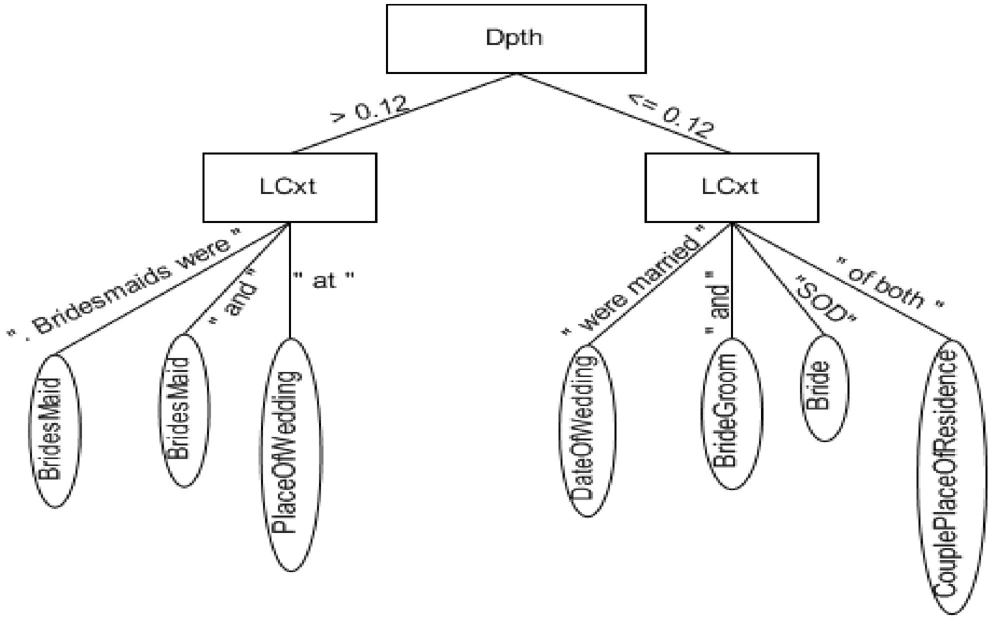


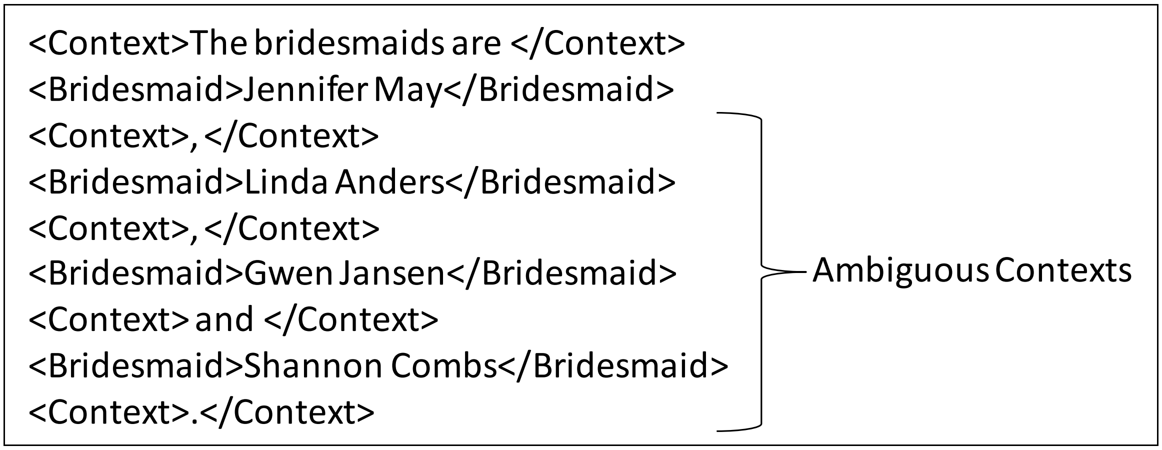
Figure 5.4: Example of Decision Tree built by J48 algorithm using the contextual attributes[6].

[6] SOD refers to Start of Document

For instance, if Depth is <= 0.12 and LCxt = “ were married ”, any candidate extracted using this (partial) condition is classified as DateOfWedding.

5.2.3 Context Disambiguation

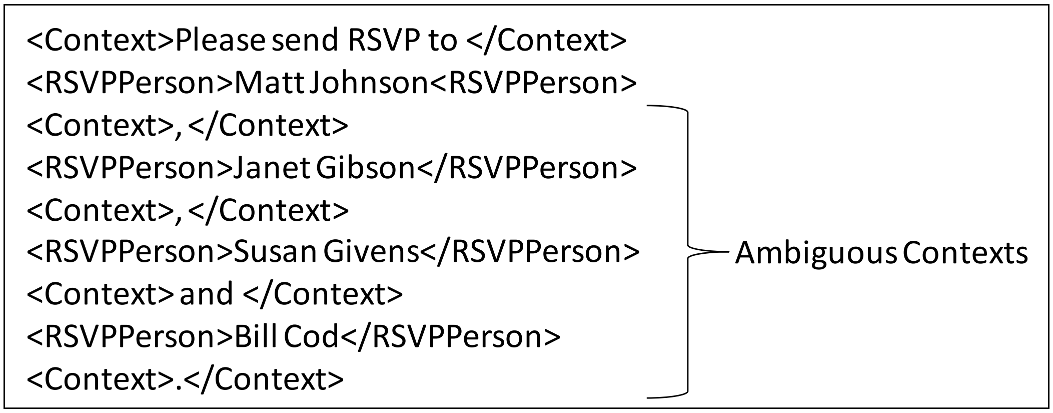
A weakness of the KB structure is the possibility of having identical contexts referencing multiple entity classes. For example in Figure 5.5 (a) and (b) LCxt and RCxt with the value “, ” reference both bridesmaid and RSVPPerson entity types within the same document. Since the same strings indicating a candidate value extracted from the target document by one of them in can also be extracted by the other, deciding the entity class to which the candidate really belongs (entity resolution) can be very difficult and sometimes impossible due to the unavailability of definitive disambiguating characteristics. Naturally, the disambiguation problem is more complex if both the LCxt and RCxt are ambiguous compare to if only one is ambiguous. If these context ambiguities are not resolved, they may be compounded by further ambiguities introduced by the use of partial context matching.



```
<Context>The bridesmaids are </Context>
<Bridesmaid>Jennifer May</Bridesmaid>
<Context>, </Context>
<Bridesmaid>Linda Anders</Bridesmaid>
<Context>, </Context>
<Bridesmaid>Gwen Jansen</Bridesmaid>
<Context> and </Context>
<Bridesmaid>Shannon Combs</Bridesmaid>
<Context>.</Context>
```

Ambiguous Contexts

Figure 5.5 (a): Ambiguous contexts for Bridesmaid



```
<Context>Please send RSVP to </Context>
<RSVPPerson>Matt Johnson<RSVPPerson>
<Context>, </Context>
<RSVPPerson>Janet Gibson</RSVPPerson>
<Context>, </Context>
<RSVPPerson>Susan Givens</RSVPPerson>
<Context> and </Context>
<RSVPPerson>Bill Cod</RSVPPerson>
<Context>.</Context>
```

Ambiguous Contexts

Figure 5.6 (b): Ambiguous contexts for RSVPPerson

The context disambiguation feature is introduced as an attempt to solve this problem. This feature prevents the use of contexts pairs that reference more than one entity type in the KB for candidate extraction. It augments the LCxt with the preceding entity class tag and/or the RCxt with the immediately following entity class tag depending on which is ambiguous. This allows the knowledge of the preceding/following entity class to be considered when extracting and disambiguating candidates with ambiguous contexts.

<Context>***{Bridesmaid}***, </Context>
<Bridesmaid>Linda Anders</Bridesmaid>
<Context>, ***{Bridesmaid}***</Context>

<Context>***{Bridesmaid}***, </Context>
<Bridesmaid>Gwen Jansen</Bridesmaid>
<Context> and ***{Bridesmaid}***</Context>

Figure 5.7: Example of disambiguated contexts for bridesmaid using entity tags from Figure 5.5 (a)

Because some the disambiguated contexts require the prior recognition of entities within the target document, the NER process is slightly different from the regular procedure of extracting all the possible candidates and then deciding which ones are the most likely correct ones. When using this feature, candidates without disambiguated contexts are first extracted and the most likely correct ones chosen. That information is then inserted into the test document so that a match using the disambiguated context is possible. For example, “Jennifer May” and/or “Shannon Combs” in Figure 5.5 (a) must be identified as bridesmaid before a match for the disambiguated contexts in Figure 5.7 can be found.

For obvious reasons, a partial match that ends in the middle of an entity name used to augment a context is not allowed (since it may also be ambiguous). Another method of augmenting ambiguous context that was considered was the use of the KB document stings (context and entity value) instead of the entity type as shown in Figure 5.8.

```
<Context>The bridesmaids are Jennifer May, </Context>
            <Bridesmaid>Linda Anders</Bridesmaid>
<Context>, </Context>
```

Figure 5.8: Example of disambiguated context for bridesmaid using the document content from Figure 5.5 (a).

Using this form of context disambiguation would require the string “Jennifer May” or “May” to be contained at the exact same position in the target document in order to get a successful match. This type of requirement forces the test document to be similar to the example documents than absolutely necessary. On the other hand, using the disambiguation in Figure 5.7 allows the disambiguated context a much greater flexibility in that it may be found regardless of the value of the bridesmaid entity class.

A limitation of this solution is that the failure to recognize an entity may prevent the identification of others. Also, an error in the recognition of a single entity can easily propagate into recognition error(s) in subsequently recognized entities. It is important to note that the context disambiguation procedure may still produce ambiguous candidates; however the information about the preceding or following entity provide valuable information for resolving such candidates.

5.2.3.1 Bootstrapping

While the context disambiguation feature reduces the complexity of distinguishing between candidates, it may result in inability to extract any candidate. For instance if several of the contexts in a document have been disambiguated, the failure to extract any first candidate would lead to the failure to extract adjacent candidates with ambiguous context. The bootstrapping feature is triggered if during a candidate location process (i.e. attempt to locate candidates using all available fragments) no candidate is returned. It relaxes the disambiguation feature by allowing only one of the contexts (LCxt or RCxt) to be ambiguous. For example, if the RCxt had previously been disambiguated, it may be returned to its ambiguous state and used for candidate location if the LCxt used along with it is not ambiguous or vice versa. This procedure can be started with either of the contexts (left or right). If bootstrapping with the LCxt and RCxt (regardless of order) fails consecutively to return any candidate, the entity loaction process is terminated.

5.3 Entity Type Document

The Entity Type Document (ETD) is used to declare ontology information of the KB domain. The declared information in the ETD include the number of occurrence of each entity class, sequence in which the entity class occur (i.e. transition between entity classes - preceding and following entity class) etc. within the KB. The ETD is analogous to Document Type Definition (DTD) used for markup declarations for SGML-family markup languages and its name also stems from it. This information may be given explicitly by the user or can be automatically extracted from the KB. The ETD can help in reducing the likelihood of error by for example, avoiding the extraction of candidate belonging to different entity classes more than the number of occurrence declared. Consequently, it can improve the runtime of the extraction process. Furthermore, the information on entity class transition provides additional information that can be used for candidate disambiguation. For example, if the ETD indicates that the entity “bride” is always followed by the entity “groom”, this information could be used to disambiguate between two different possible “groom” candidates depending on their relative location to the bride in the document.

This feature has not been greatly explored yet and is among the future work tasks. However experiments were performed in which the number of occurrence of entity types in each document is considered.

Chapter 6

Candidate Selection

A candidate is a word or phrase extracted from unstructured text by the NER process but is yet to be declared as correct. It is also described using the same characteristics used to describe the KB entities i.e. contextual and intrinsic properties. One of the consequences of using fragments, partial contexts and interchanging the LCxt and RCxt for entity location is that candidates with different characteristics may be purported to represent the same entity/entity class. The identification of the most likely correct candidate(s) from the multitude of possible candidates is therefore a form of belief function. This belief function reflects the ensemble of evidence from both the contextual and intrinsic properties. In other words, it represents the amount of evidence found in the KB that a candidate_i having one or any combination of Leng_i, Tokn_i, Char_i, Dpth_i, LCxt_i, and RCxt_i, characteristics belongs to a particular the entity class. The value of the belief function for each candidate is termed the candidate score.

6.1 Candidate and Entity Comparison Measures

In order to estimate the amount of evidence that can be found in the KB about the correctness of a particular candidate, the candidate must be compared with entities of the same class within the KB. This comparison between candidates and entities is done by comparing their characteristics. Hence, if a candidate compares well with an entity of the same class in terms similarity in any combination of Leng, Tokn, Char, Dpth, LCxt and RCxt, there is a high probability that it is correct.

Illustrated below is how the intrinsic and contextual properties of candidates and entities of the same class can be compared (these comparison techniques are used in the experiments described later).

6.1.1 Length Similarity

The length similarity measures if the number of characters in the string 1 is equal to the number of characters in string 2.

$$\text{Leng similarity} = \begin{cases} 1 & \text{string } 1 = \text{string } 2 \\ 0 & \text{string } 1 \neq \text{string } 2 \end{cases}$$

6.1.2 Token Similarity

Token similarity compares the number of space character(s) in string 1 to that of string 2.

$$\text{Tokn similarity} = \begin{cases} 1 & \text{string } 1 = \text{string } 2 \\ 0 & \text{string } 1 \neq \text{string } 2 \end{cases}$$

6.1.3 Character Composition Similarity

3 different techniques for assessing the character composition of entity values were described in section 4.2.2. The comparison of the candidate value character composition varies according to the technique adopted. Below are descriptions of how the comparison for each of the technique can be implemented. The different comparisons may be performed with or without taking into account the different alphabet cases (capitalization) e.g. the string "Apple" may be considered as "Aaaaa" or "aaaaa"; the representation must however be consistent through the entire process. Also, as with the other comparisons, the comparisons are carried out only among entities and candidates of the same class.

i.) The characters contained in the string are checked against known characters for a particular entity type regardless of order.

Char similarity = $\frac{\text{number of candidate value characters found in the KB entity}}{\text{number of characters in candidate value}}$

ii.) The alphabet and numeric characters contained in the string are transformed into a single alphabetic and numeric representation respectively e.g. Jan. 18, 2010 is transformed into Aaa. 99, 9999.

Char similarity = function of the distance between the compared strings e.g. Levenshtein distance (Levenshtein, 1966)

iii.) The string value is transformed in the same manner as the second method. However, the counts of each character type (alphabet, number and punctuation) are compared e.g. Jan. 18, 2010 is transformed into {alphabet count: 3, number count:.6, punctuation count: 4}

Char similarity = Average of character type comparisons

Where

Character type comparison = $\begin{cases} 1 & \text{string } 1 = \text{string } 2 \\ 0 & \text{string } 1 \neq \text{string } 2 \end{cases}$

6.1.4 Depth Similarity

The depth similarity compares the relative index of string1 within its source document to the relative index of string 2 within its source document.

Depth similarity = $\begin{cases} 1 & \text{string } 1 = \text{string } 2 \\ 0 & \text{string } 1 \neq \text{string } 2 \end{cases}$

6.1.5 Context Ambiguity

The contexts of entities and/or candidate are compared not just based on their character(s) composition, but on the likelihood that they reference more than one class of entity in the KB. Context ambiguity is the conditional probability of a set of contexts identifying more than one entity class in the KB.

Context ambiguity = $\frac{\text{count of contexts in KB identifying a particular entity class}}{\text{total count of context in KB}}$

6.2 Range for Characteristics Comparison

As mentioned earlier, the purist implementation of the proposed NER system would involved the use of KB(s) with a significantly large number of example documents with which candidates can be compared. The high number of entities in such large KB will allow for a reasonable likelihood that direct characteristics comparison would yield a hit. However, if the size of the KB is small, the likelihood that exact candidate characteristics will be found in the KB is significantly lower.

The use of range of tolerance within which characteristics comparison are considered equivalent was introduced. This range is determined by allowing for elbow room (ε) around the mean value of the KB property parameter (μ) such that the range R can be defined as e.g. [μ-ε, μ+ε]. For example, if the mean of the length of the bride entities in the KB is equal to $\mu_{\mathrm{bride}}^{\mathrm{length}}$, then the comparison range for the length of the bride candidates, $\mathrm{R}_{\mathrm{bride}}^{\mathrm{length}}$ can be described as [$\mu_{\mathrm{bride}}^{\mathrm{length}}$-$\varepsilon$, $\mu_{\mathrm{bride}}^{\mathrm{length}}$+$\varepsilon$] The greater the value of ε, the higher the chances of identifying wrong candidates as correct.

A technique for assigning value to ε is the use of the standard deviation (δ). This is done by calculating the standard deviation of the values of individual characteristics for each entity class contained in the KB, and then used to determine the value of ε for corresponding candidate entity class and characteristics comparisons. Because this technique uses the distribution of the characteristics values in the KB, it allows users to statistically estimate the proportion of the values that fall within the range and avoids errors of using arbitrary ε value. The experiments described in the next chapter used 3 standard deviations as the value of ε to ensure that only comparisons that indicate extreme outliers are rejected.

6.3 Candidate Scoring Methods

There are several methods by which candidate score can be calculated. An array of statistical tools may be used to develop probabilities or evidence theories that can be used for candidate scoring. A high score indicates a high likelihood that a candidate is correct. It is important to note that if a strict fragment/document matching is implemented (as would be in a purist implementation), the likelihood of extracting wrong candidate would be relatively low and the use of frequency may be enough to establish reasonable candidate scores.

Described below are the methods explored in calculating the candidate score namely: Bayesian, Dispersion, Dempster-Shafer belief theory and the C4.5 classifier. These methods described below are used for the experiments illustrated in the next chapter.

6.3.1 Bayesian Candidate Scoring Method

The Bayesian scoring method calculates the candidate score as the conditional probability of a candidate simultaneously having all the same characteristics and belonging to the same class as entities in the KB i.e. the candidate score is the ratio of the number of KB entities with the same characteristics and class as the candidate to those that have the same characteristics. Figure 6.1 shows the population of KB entities (A) with the same characteristics as a candidate$_i$ of the same entity class.

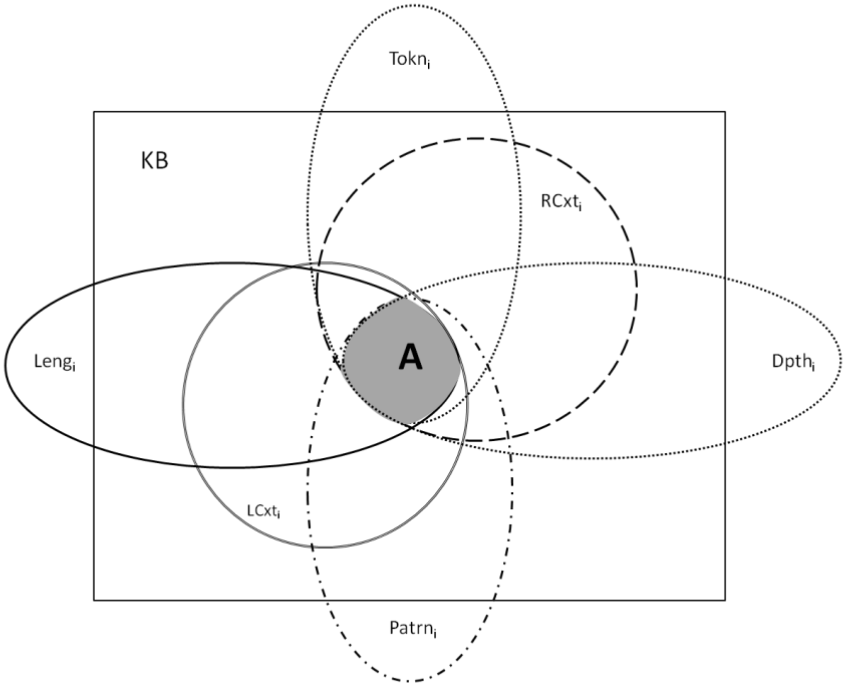


Figure 6.1: Intersection of entities drawn from the KB using the intrinsic candidate characteristics is labeled A.

A = $LCxt_i \cap RCxt_i \cap Leng_i \cap Tokn_i \cap Patrn_i \cap Dpth_i$.

Where $LCxt_i$, $RCxt_i$, $Leng_i$, $Tokn_i$, $Patrn_i$ and $Dpth_i$ are the characteristics of candidate$_i$

$$\text{Candidate Score}_{\text{Bayesian}} = \frac{\#\text{correct entities of } A_\alpha}{\text{size of set } A}$$

Where A_α is the sub-set of the set A belonging to type α

A strict implementation of the Bayesian scoring method (i.e. all characteristics must be exact match) has the possibility of achieving a high precision, however at the expense of recall.

6.3.2 Dispersion Candidate Scoring Method

The dispersion method calculates the candidate score as the product of the context ambiguity and the probability density of, Leng, Tokn, Patrn and Dpth referencing the candidate's type. The probability density of $\text{characteristic}_{i,j}$ (i indicates i-th candidate and j indicates j-th characteristic) referencing entity type α is a function of its z-score relative to the mean and standard deviation of the characteristic_j of all KB entities of type α. Hence it gives a measure of the likelihood that $\text{characteristic}_{i,j}$ is close to the population mean.

$$\text{Candidate Strength}_{\text{Dispersion}} = f(\delta_{\text{Leng}}) * f(\delta_{\text{Tokn}}) * f(\delta_{\text{Patrn}}) * f(\delta_{\text{Dpth}}) * \text{context ambiguity}$$

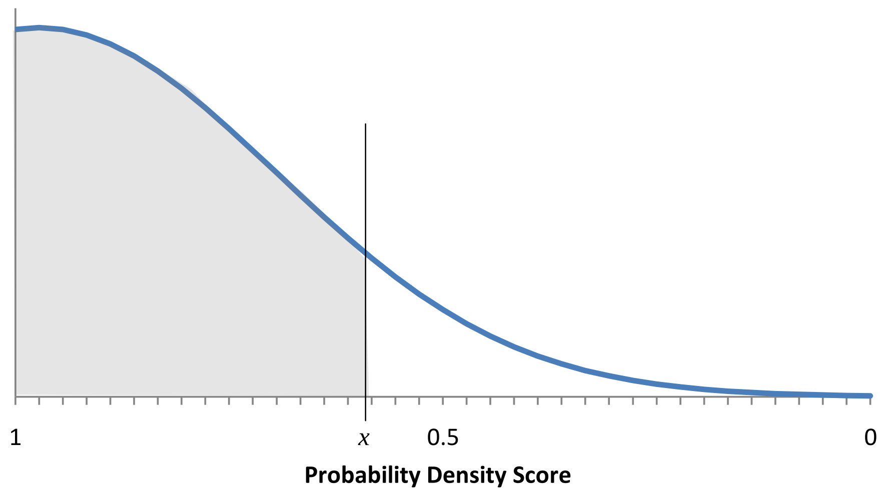


Figure 6.2: Probability Density Function Chart. The closer x is to the mean the higher the probability of its correctness

This dispersion scoring method can be very effective in identifying outliers in the candidate list. However, a low (or zero) score from any characteristic comparison will have a heavy impact on the overall candidate score e.g. a zero token probability density would result in a null candidate score.

6.3.3 Dempster-Shafer Belief Theory Candidate Scoring Method

Dempster-Shafer belief theory allows for the gradual increase/decrease of the candidate score based on the evidence provided by individual characteristics (Shafer, 1976) thereby reducing the heavy impact individual evidence might have on the entire score. Using this scoring method ensures that the candidate strength is not nullified when only a single attribute score is/close to zero. The candidate score is calculated as the product of the belief from the contextual and intrinsic evidences.

Candidate Score $_{\text{Dempster-Shafer}}$ = Contextual Belief * Intrinsic Belief[7]

Where:

Contextual Belief=1 - (1 – LCxt $_{\text{probability}}$) * (1 – RCxt $_{\text{probability}}$) * (1 – Dpth $_{\text{probability}}$)

Intrinsic Belief=1 - (1 – Leng $_{\text{probability}}$) * (1 – Tokn $_{\text{probability}}$) * (1 – Patrn $_{\text{probability}}$)

Where

The LCxt $_{\text{probability}}$, RCxt $_{\text{probability}}$ etc are the conditional probabilities of each candidate characteristic identifying entity class α in the KB (given that the candidate is of class α)

6.3.4 Probabilistic Classifier Candidate Scoring Method

This method uses decision trees built using probabilistic classifiers for locating candidates from unstructured text and calculating their scores. The trees are built such that the leaf nodes represent the entity class associated and the entity characteristics represented by the internal nodes. In the first step, a decision tree built using the contextual property is used to locate candidates from the UTI (see section 5.2.2.2 & Figure 5.4). And in the second step, another decision tree is built using the intrinsic property is used to verify if the entity class assigned by the contextual decision tree is confirmed by the candidate's intrinsic properties (Figure 6.3). An extracted string is considered a candidate only if both decision trees assign it to the same entity class.

[7] The candidate score is equal to zero only if all the evidence from the contextual or intrinsic properties indicate it to be zero

The probability distribution of the KB properties estimated by the decision trees are used in calculating the candidate score. It is calculated as the product of the contextual and intrinsic scores, where the intrinsic and contextual scores are calculated as the ratio of the correctly classified leaf node instance(s) to the total leaf node instances.

$$\text{Candidate Score}_{\text{Probabilistic Classifier}} = \text{Classifier score}_{\text{Contextual}} * \text{Classifier score}_{\text{Intrinsic}}$$

Where:

$$\text{Classifier score} = \frac{\text{Correctly classified Leaf instances count}}{\text{Total leaf instances count}}$$

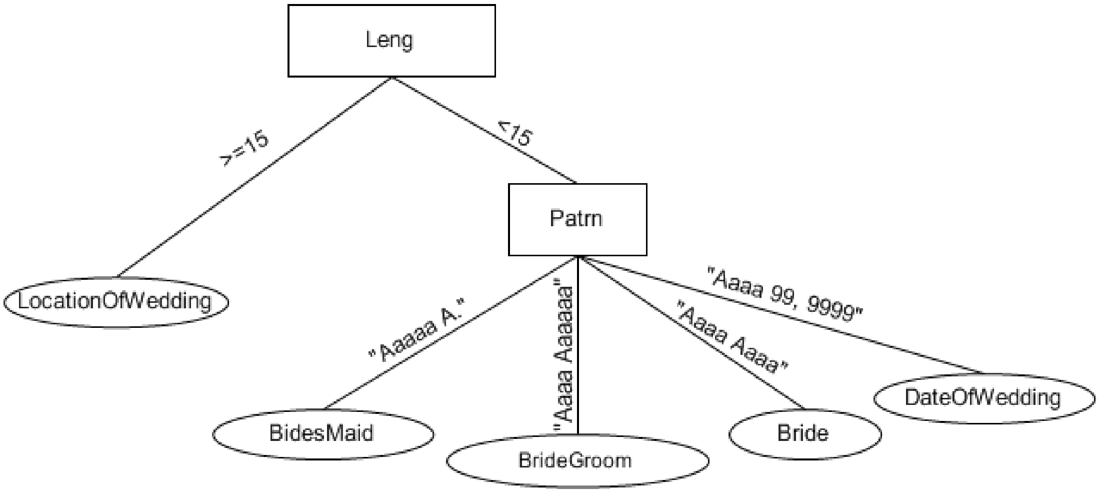


Figure 6.3: Example of Decision Tree built by J48 algorithm using the intrinsic attributes.

For example, if the length of the candidate value is less than 15 and its pattern is equivalent to “Aaaa 99, 9999”, then the candidate is classified as bride.

An advantage of this method compared to the others is that it is very fast in processing a relatively large KB since it efficiently eliminates redundancies and builds “statistically optimal” decision trees. On the other hand, it may eliminate rare characteristics combination thereby making it difficult/impossible to locate candidates with such characteristics. Furthermore, none usage of some of the entity properties e.g. LCxt and RCxt might make candidate extraction impossible, if the generated decision tree is not modified to include them (described in section 5.2.2.2).

6.4 Other Candidate Scoring Considerations

As mentioned earlier, the user may use any preferred method(s) for scoring the candidates. Some other information that may be considered for the scoring process includes:

- Context Identification: This feature is useful when disambiguated contexts are used for entity location. Context identification prevents the portion of the target document used as contexts in the identification of any entity from being recognized as (or part) of a candidate value. For example, given the target document “`... three sons, James Lee Smith and his wife, Pat, of...`” The recognition of “`James Lee Smith`” which transforms the target document into “`... three sons, <Child> and his wife, Pat, of...`” increases the likelihood of recognizing “`his wife, Pat`” as a child entity class; mainly due to its relatively strong LCxt value “`<Child>`

`and `"[8]. This feature prevents "`his wife, Pat`" from being recognized as a candidate if any part of "`his wife `" is part of the context used to recognize "`James Lee Smith`".

- <u>Boundary Characters</u>: In addition to using the contexts to determine the candidate value boundaries, the beginning and ending characters of the KB entities can be used to identify string values to be accepted as candidate values.
- <u>Sister Candidates</u>: The number of candidates with the same value and belonging to the same entity class but extracted using different combinations of unambiguous LCxt and RCxt may be considered as an indicator that a candidate is likely correct. This information can be used to add a multiplier effect to the candidate score e.g.

$$\text{Candidate Score} = (\text{Candidate Score}_1)^{\kappa}$$

Where: κ is the number of candidates with the same entity class and value but different context combination(s).

6.5 Candidate Filtering

Candidate filtering is the process of identifying the most likely correct candidates from the list of extracted candidates. The candidate scores are used in comparing the different candidates during the elimination procedure. The filtering process can be implemented by choosing successive candidates with the

[8] Because disambiguated contexts are generally less ambiguity, they contribute to more to the candidate score than regular contexts.

highest score in the list and eliminating every other candidate with overlapping entity value. For example given a target document containing the text *“…were married May 5, 2000, at…”* and two candidates *“May 5”* and “*May 5, 2000*” both belonging to the date of wedding entity class extracted from the document. Assuming that “May 5, 2000” has a higher score than “May 5”, “May 5” is eliminated because its location in the document intersects the location of “May 5, 2000”. This procedure is repeated until all intersecting candidates of lower scores have been removed. The remaining candidates are then considered as entities from the test document.

Because high candidate scores implies high likelihood of correctness, setting a high enough candidate score threshold can help in increasing the number of correct candidates returned by the system. When disambiguated contexts are used, a good threshold value may prevent the recognition of wrong candidates which may make the recognition of correct candidates impossible. Methods for choosing ideal threshold values have not been fully researched and it may very well be strongly dependent on the KB. However, it may be possible to determine the threshold value by performing NER experiments on several of the KB documents and observing the scores of the highest scored correct and wrong candidates. Another method that can be explored is the visual inspection of the experiments results to determine the candidate scores where the candidate characteristics are most dissimilar to known KB entities.

Chapter 7

Code Implementation

The presented NER approach was implemented using java programming language. One of the main goals during implementation was to ensure that instances of the program can run in parallel in the sense that a single KB can be used to extract entities from several target documents simultaneously and that several consecutive entity location processes can be carried out on a target document.

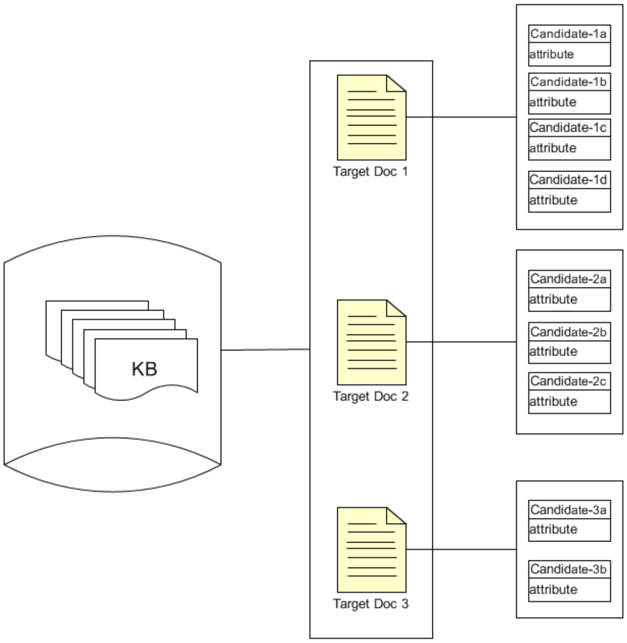
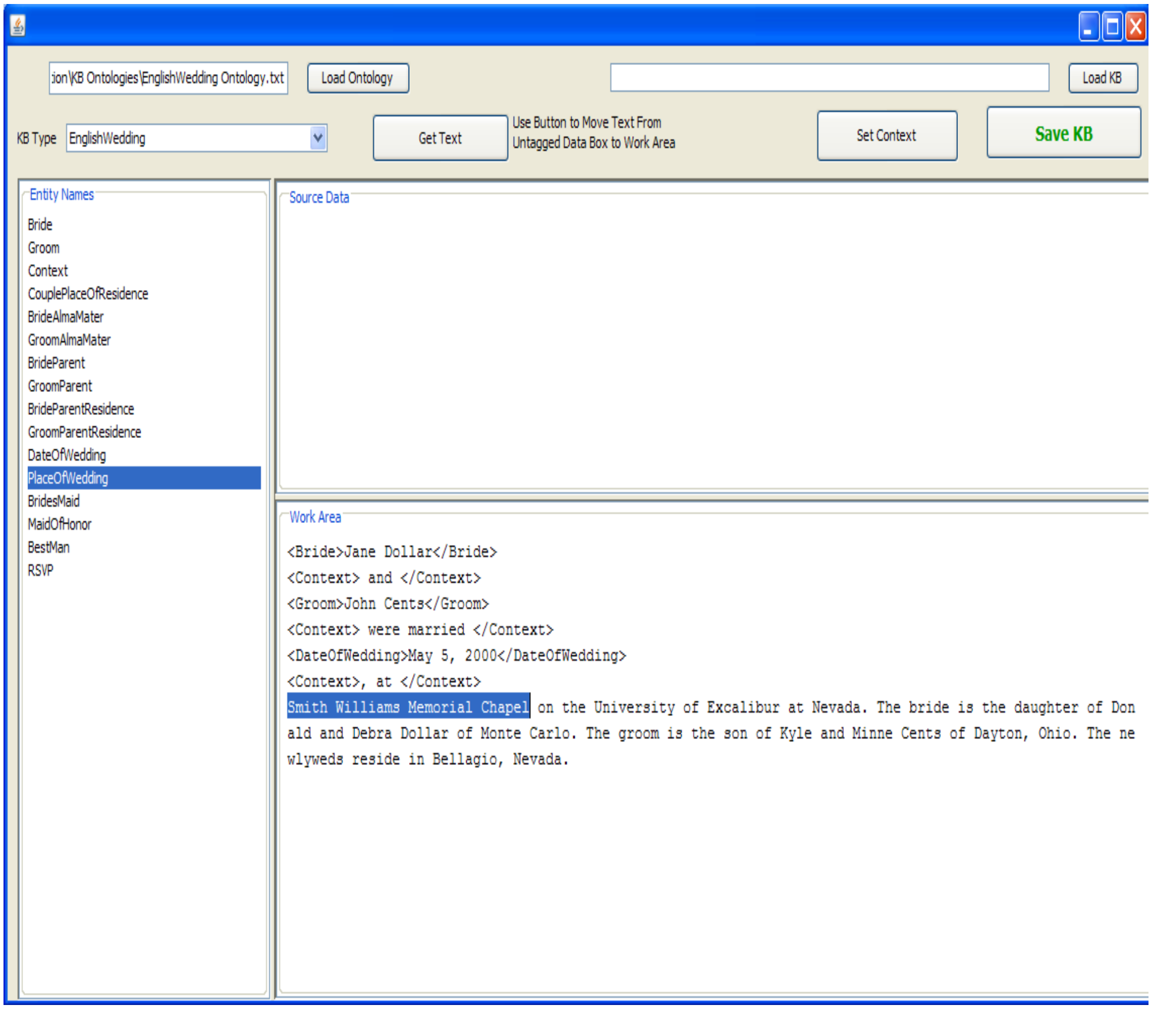


Figure 7.1: Simultaneously using the KB to extract entities from multiple target documents.

7.1 Preparing the KB Dataset

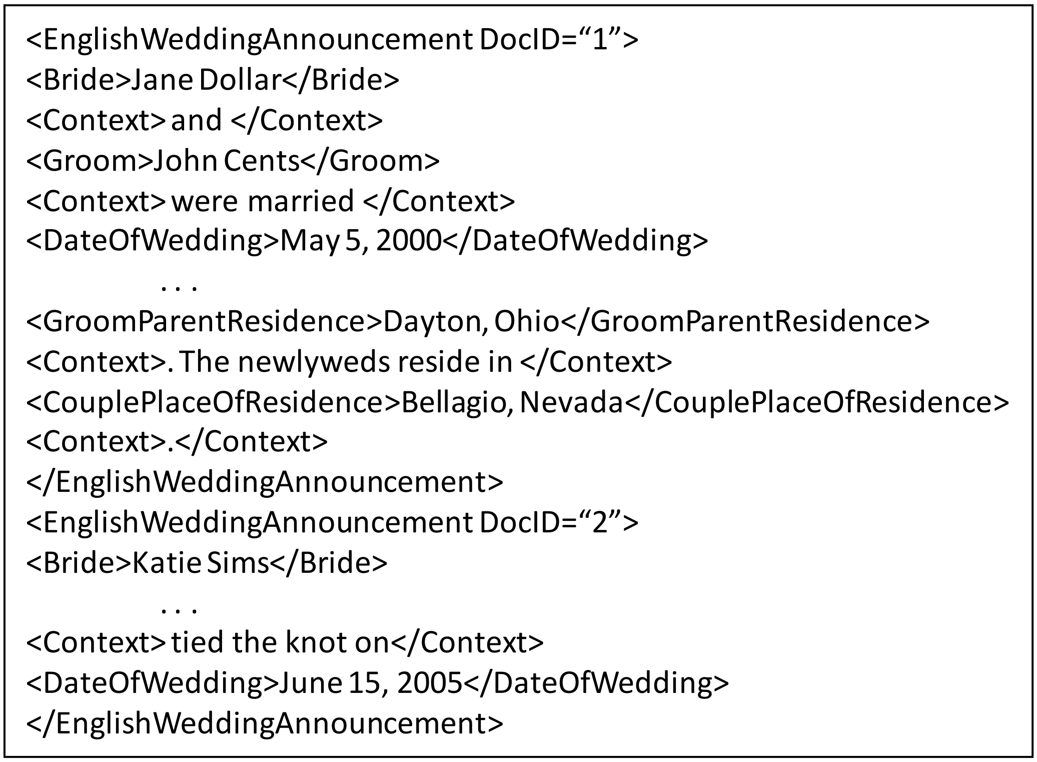
Preparation of the dataset involves annotation of the unstructured data to indicate entities of interests. This task was achieved by using a graphical user interface that allows users to load different ontology and annotate strings in the KB documents text by highlighting and selecting the corresponding entity class. The dataset preparation also involves the unique numbering of each the individual document that comprises the KB so as to facilitate easy identification. In addition to the collection of potentially valuable information about the entity and context transition within each document, the numbering provides a way of uniquely identifying of the source of individual KB entities which helps in checking the correctness of the candidates when KB documents used as target documents during the experiments and testing.



```
<Bride>Jane Dollar</Bride>
<Context> and </Context>
<Groom>John Cents</Groom>
<Context> were married </Context>
<DateOfWedding>May 5, 2000</DateOfWedding>
<Context>, at </Context>
Smith Williams Memorial Chapel on the University of Excalibur at Nevada. The bride is the daughter of Don
ald and Debra Dollar of Monte Carlo. The groom is the son of Kyle and Minne Cents of Dayton, Ohio. The ne
wlyweds reside in Bellagio, Nevada.
```

Figure 7.2: Example of a KB Annotator

The ontology includes both the type/category of the document and the entity classes that it comprises e.g. document type is “wedding announcement” and the entity classes include “Bride”, “Groom”. This information is encoded into the KB document (e.g. Figure 7.3) and helps in selecting the KB documents categories to be used for NER from different target documents. In addition, extra spaces between words and other formatting such as bullets and tab are removed from the KB and target documents as they may induce unnecessary errors during the extraction process and provide no additional information for the NER process.



```
<EnglishWeddingAnnouncement DocID="1">
<Bride>Jane Dollar</Bride>
<Context> and </Context>
<Groom>John Cents</Groom>
<Context> were married </Context>
<DateOfWedding>May 5, 2000</DateOfWedding>
            . . .
<GroomParentResidence>Dayton, Ohio</GroomParentResidence>
<Context>. The newlyweds reside in </Context>
<CouplePlaceOfResidence>Bellagio, Nevada</CouplePlaceOfResidence>
<Context>.</Context>
</EnglishWeddingAnnouncement>
<EnglishWeddingAnnouncement DocID="2">
<Bride>Katie Sims</Bride>
            . . .
<Context> tied the knot on</Context>
<DateOfWedding>June 15, 2005</DateOfWedding>
</EnglishWeddingAnnouncement>
```

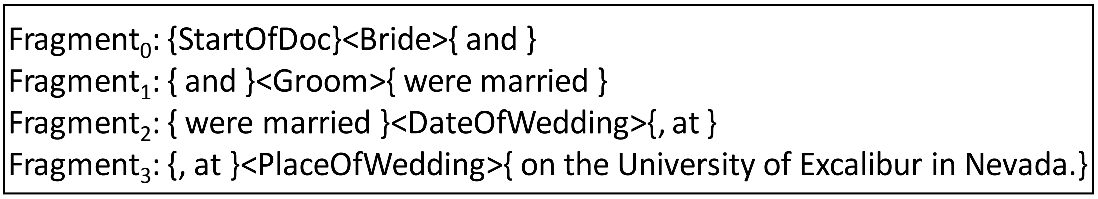
Figure 7.3: Example of 2 numbered and annotated KB Documents

7.2 Processing KB

After the KB is read by the NER application it is structured in such a way that it can be effectively used for entity extraction and its different properties and their values easily accessed. 3 main goals achieved during the KB processing phase are collecting the KB fragments, entities and entity statistics.

7.2.1 Fragment List

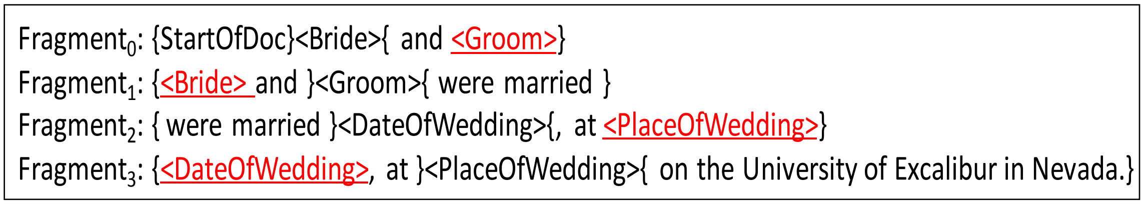
The fragment list is the collection of all the possible fragments from the KB. This process essentially organizes the KB as a series of fragments that would be used in the process of locating entities from target documents. A programming object was created in order to provide a consistent structure for each fragment and also to help in their organization. The object's attributes includes the KB document identification number, LCxt, RCxt, entity class, and depth of the entity value. In addition, information about the ambiguity of the LCxt and RCxt as well as their duplicity are also determined and stored as attributes of the fragment object.



Fragment$_0$: {StartOfDoc}<Bride>{ and }
Fragment$_1$: { and }<Groom>{ were married }
Fragment$_2$: { were married }<DateOfWedding>{, at }
Fragment$_3$: {, at }<PlaceOfWedding>{ on the University of Excalibur in Nevada.}

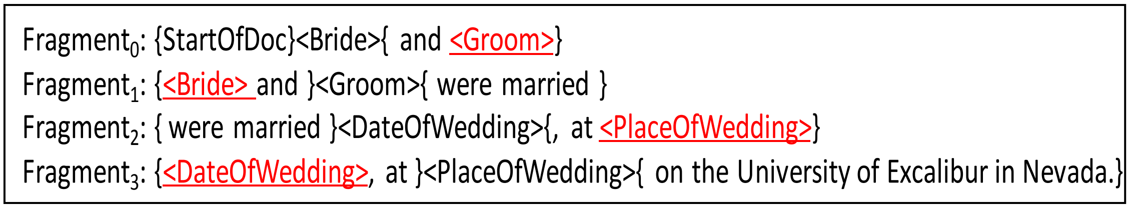
Figure 7.4: Example of indexed fragments

When the context disambiguation feature of the NER is used, 3 other sets of KB fragments are also created each for the three forms of context disambiguation: all contexts disambiguated, only LCxt disambiguated and only RCxt disambiguated (discussed in section 5.2.3.1 above).



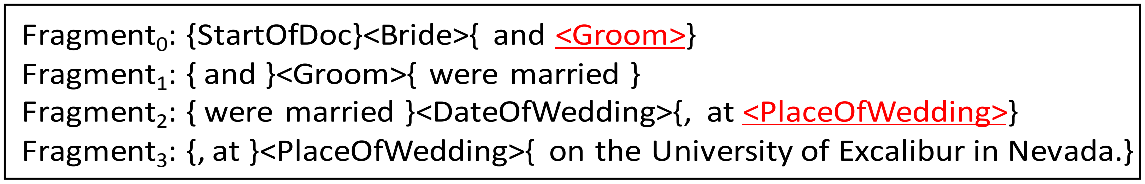
Fragment$_0$: {StartOfDoc}<Bride>{ and <Groom>}
Fragment$_1$: {<Bride> and }<Groom>{ were married }
Fragment$_2$: { were married }<DateOfWedding>{, at <PlaceOfWedding>}
Fragment$_3$: {<DateOfWedding>, at }<PlaceOfWedding>{ on the University of Excalibur in Nevada.}

Figure 7.5: Fragments with all contexts disambiguated



Fragment$_0$: {StartOfDoc}<Bride>{ and <Groom>}
Fragment$_1$: {<Bride> and }<Groom>{ were married }
Fragment$_2$: { were married }<DateOfWedding>{, at <PlaceOfWedding>}
Fragment$_3$: {<DateOfWedding>, at }<PlaceOfWedding>{ on the University of Excalibur in Nevada.}

Figure 7.6: Fragments with LCxt disambiguated



Fragment$_0$: {StartOfDoc}<Bride>{ and <Groom>}
Fragment$_1$: { and }<Groom>{ were married }
Fragment$_2$: { were married }<DateOfWedding>{, at <PlaceOfWedding>}
Fragment$_3$: {, at }<PlaceOfWedding>{ on the University of Excalibur in Nevada.}

Figure 7.7: Fragments with RCxt disambiguated

7.2.2 Entity List

The entity list comprise of all the entities in the KB. The information about each attribute is structured using a programming object designed specifically for this purpose. The entity object contains information about the annotated KB entity value along with its entity class, depth, preceding entity class and succeeding entity class in addition to the identification number of the KB document.

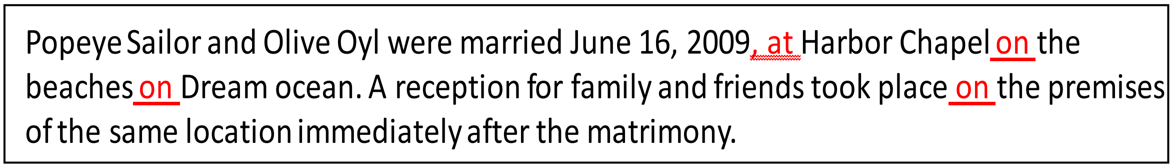
7.2.3 Entity Class Statistics List

The entity list statistics are obtained from the entity list described above. After all the KB entities have been identified, basic statistics information about each of the intrinsic properties and the depth (context statistics are calculated during the program execution) of each entity class are collected for use in the validation of the candidate value extracted using the contexts. Again, the statistical information for each entity class is stored as an individual programming object. The attributes of this programming object include the minimum, maximum, mean and standard deviation values for each KB property e.g. entity length and token.

7.3 Candidate Location

The candidate location process involves the use of the KB fragments to locate candidate values from within the target document. Depending on the type of scoring method used, the fragments list described in section 0 can be iteratively used to extract candidates (Bayesian, Dispersion and Dempster Shafer). When a scoring method like the probabilistic classifier is used, the decision tree is traversed in order to determine the candidate extraction steps.

In general, the test document is searched for strings matching the LCxt and RCxt starting in any order. Assuming that the context matching is started with the LCxt, the first occurrence of the LCxt (or part of it) is sought in the target document. If found, successive RCxt is sought allowing for the presumed candidate value to be at least one character long. After the RCxt search has been exhausted, the search procedure is repeated starting with the next LCxt occurrence in the target document thereby generating numerous potentially candidates. The number of identical candidates extracted and the speed of the process can be reduced by avoiding the use of duplicate LCxt and RCxt (attributes of the fragment object).



Popeye Sailor and Olive Oyl were married June 16, 2009, at Harbor Chapel on the beaches on Dream ocean. A reception for family and friends took place on the premises of the same location immediately after the matrimony.

Figure 7.8: Example of candidate extraction from target document

Candidate values that can be extracted using the fragment: {, at }<PlaceOfWedding>{ on } are “*Harbor Chapel*”, “*Harbor Chapel on the beaches*” and “*Harbor Chapel on the beaches on Dream ocean. A reception for family and friends took place*”.

The possible length of the candidate value can be controlled by using for example a function of the maximum length value of the entity class obtained from the entity statistics list. Because excessively long candidate values would be later ranked lower compared to those with more moderate length and consequently filtered out, limiting of the length of the extracted candidate mostly affects the execution time of the extraction process (reduce) without affecting its accuracy. Similarly, the region of the document searched can also be limited using the depth statistics.

The output of the candidate extraction process is a list of all the possible candidates extracted from the target document. The information about each candidate like other important information in NER process is stored using a programming object with attributes that include the value of the left and right context used for its location, value, entity class and depth.

7.4 Candidate Scoring and Filtering

After the candidates are extracted, the contextual and intrinsic properties of the candidate are used to calculate the degree of belief that the candidate is correct. As previously described, calculation of the candidate score is dependent on the scoring method chosen. If the probabilistic classifier method is used, a decision tree leaf that exhibits all the candidate intrinsic properties is searched for and used in determining the candidate score. Otherwise, each of the candidate characteristic is compared to the KB entity characteristics. Candidates are scored generally on the context, depth, length, token and character composition.

A separate process is then used to sort the candidates in the order of score magnitude and remove candidates with lower score and values intersecting that of candidates with higher scores.

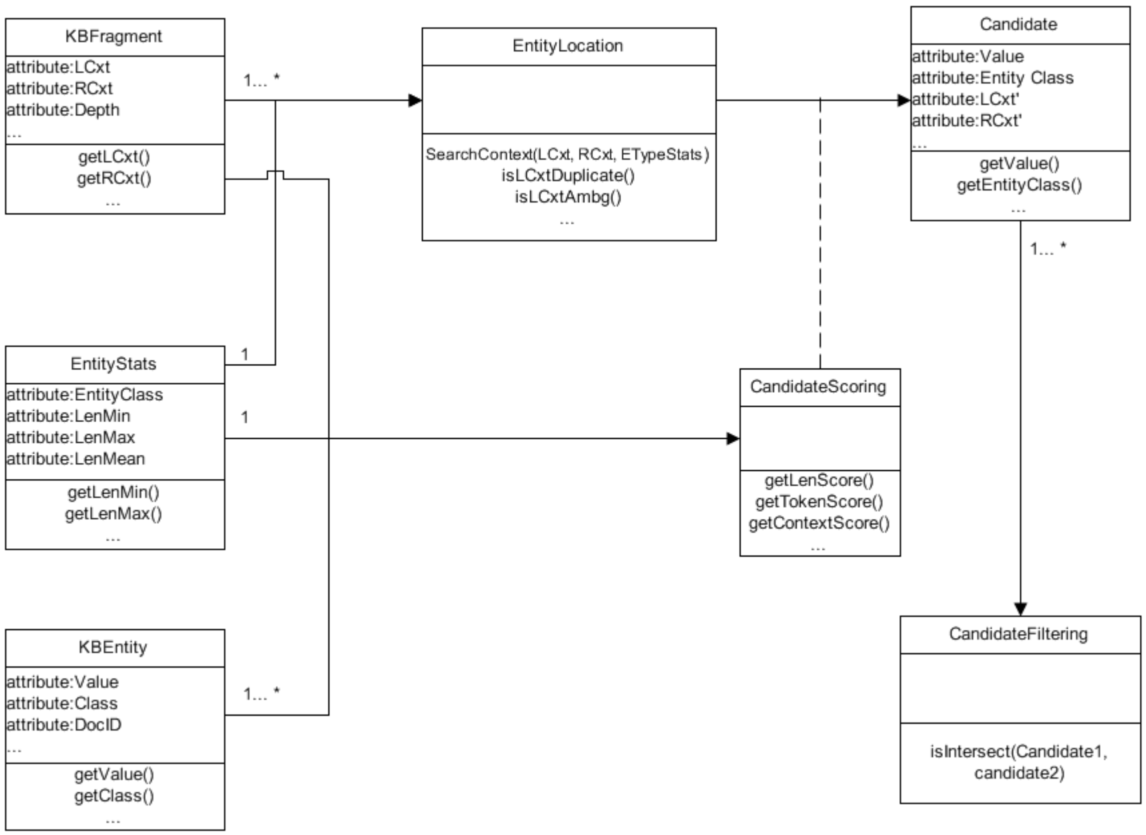


Figure 7.9: Overview of the Software Implementation

Chapter 8

Experimental Results

This chapter contains information on how the NER system performance can be measured, the different set-ups of experiments for evaluating the system and the analysis of the experiment results.

8.1 Performance Measurement

The performance of this NER system can be measured using a general framework for assessing information extraction and NER systems (Van Rijsbergen, 1979). The relations between the correct results and the results returned by the NER system are measured using precision and recall; and the relation between these two measurements is represented by the F-measure.

Precision is the measure of the correctly identified entities by the NER system (true positive) relative to the total number of entities in the target document.

$$\text{Precision} = \frac{\text{count of correct NER result}(s)}{\text{count of correct results}}$$

Recall is the proportion of correct entities within the total number of entities recognized by the NER system.

$$\text{Recall} = \frac{\text{count of correct NER results}}{\text{count of NER results}}$$

F-measure is the harmonic mean of the precision and recall. It gives an overall accuracy of the NER system.

$$\text{F-measure} = \frac{(\beta^2+1)*\text{precision}*\text{recall}}{\beta^2+\text{precision}+\text{recall}}$$

Where β indicates the relative importance of recall and precision.

$$\beta = \begin{cases} < 1 & \text{precision more important} \\ = 1 & \text{equal importance} \\ > 1 & \text{recall more important} \end{cases}$$

8.2 Experiment Set-up

The experiment set-up involved the use of 2 different KBs. The first KB contains 100 different obituary announcements in the English Language and the second KB contains 49 different news reports of crime incidents in the Spanish language. A number of documents from each of the KB were used as test documents while the rest were used as KB documents during experiment. The test documents were hidden from the NER system during the candidate extraction process, and were only used to identify the correct results from among the NER results set[9]. Each experiment was conducted using the four candidate scoring methods described in section 6.3.

8.2.1 Set-up 1: Experiments with English Documents

The first experiments were performed on English language obituary announcements obtained from Internet websites (See example: Figure B-1). For the purpose of this experiment, these annotated example documents are referred to as (KB1). 10 of the documents were used as test documents while the other 90 were used in an aggregate series of different KB sizes. The KB sizes used during the experiment were 20, 30, 40, 50, 60, 70, 80 and 90.

8.2.1.1 Relationship between NER performance and minimum number of context characters (English Obituary)

The purpose of this experiment is to determine the performance of the proposed data intensive NER system relative to different values of the minimum number of context characters (minCxtLen) allowed. This information would help in identifying optimal minCxtLen value for the system. minCxtLen values ranging from 2 to $\underline{\infty}$ (partial context match is not allowed) were used for recognizing entities in 10 different test documents and 7 different KBs (with different sizes) using the above discussed candidate scoring methods. minCxtLen value of 1 was omitted from the experiments because preliminary results revealed that the

[9] This allows us to easily verify if the validity of the system's choice.

results are worse compared to higher minCxtLen values. Moreover, the amount of execution time required is significantly much higher.

Figure 8.10, Figure 8.11 and Figure 8.12 show the average precision, recall and F-measure for the different minCxtLen values. The experiment was repeated for the context disambiguation feature turned on and off in order to account for the relationship between context disambiguation and minCxtLen value. The range within which different comparisons are accepted is 3 standard deviations of elbow room on each side of the characteristics mean (section 6.2). This comes from the established knowledge of statistics that the vast majority of the characteristics values fall within the 3 sigma range and values outside this range can potentially are often outliers and used to determine when a system goes out of statistical control.

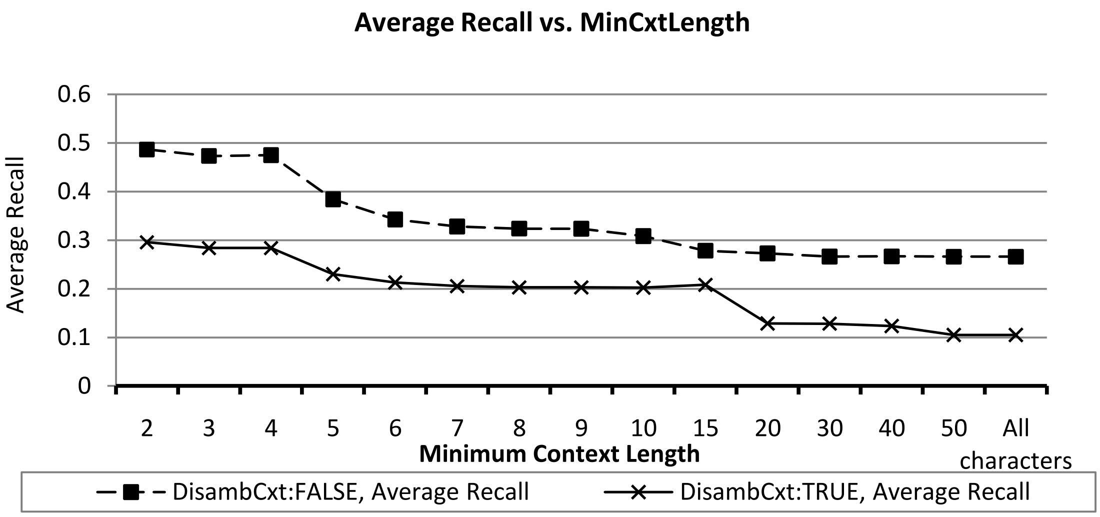


Figure 8.10: NER recall rate at different minimum context length values (English Obituary)

As depicted in Figure 8.10, the recall rates decreased as the value of minCxtLen increased. This decrease was expected because of the lesser opportunities of finding longer contexts within the test document; this leads to a decreased in the number of contexts used for candidate extraction. Generally, when the context disambiguation feature is turned on (DisambCxt: TRUE), the total number of contexts used for extraction is relatively lower compared to when the feature is turned off (DisambCxt: FALSE). Because the number of candidates returned when regular contexts (non-disambiguated context) are used is greater than when disambiguated context are used, the opportunity for finding correct candidates is greater. For example, as can be seen in Figure 8.11 the number of candidates returned by regular contexts is about 6 times larger than disambiguated contexts. This phenomenon is responsible for regular contexts usage having a better recall performance than disambiguated contexts in Figure 8.10.

Furthermore, the dependence of one candidate extraction on another through the disambiguation process significantly affects the number of candidates extracted with disambiguated contexts. For example in Figure 5.7, “Linda Anders” must be extracted as a bridesmaid in order for “Jennifer May” (preceding entity) and/or “Gwen Jansen” (succeeding entity) to be extracted as bridesmaid entity class.

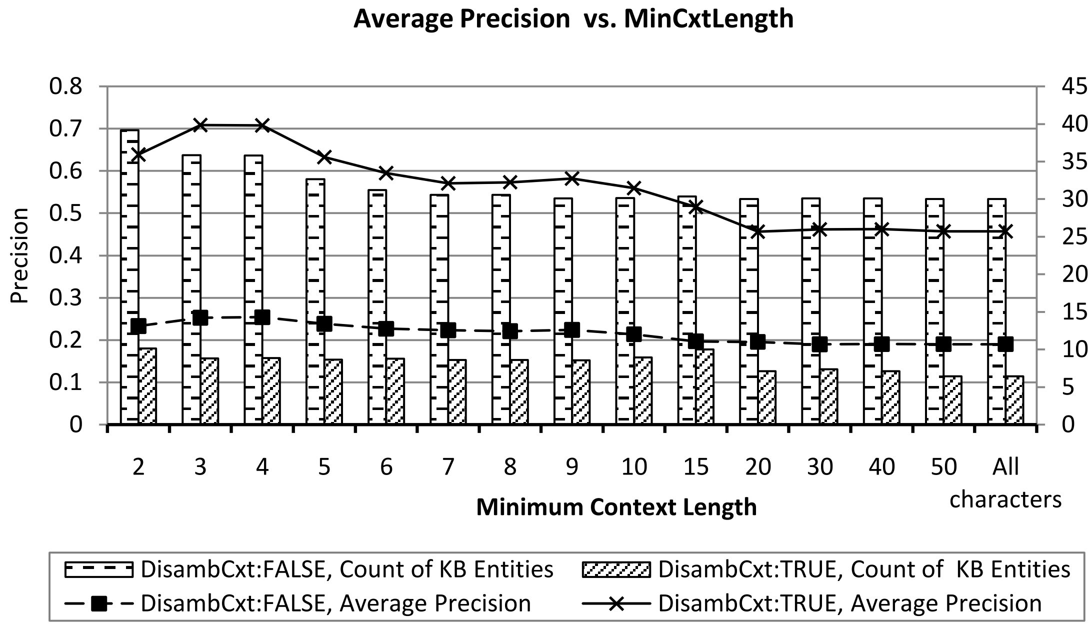


Figure 8.11: NER precision rate at different minimum context length values (English Obituary)

As expected, disambiguated contexts consistently yielded better precision than regular contexts. This performance can be attributed to the additional information contained in disambiguated contexts for positively identifying candidates. As can be observed in Figure 8.11, although the number of candidates returned by disambiguated context usage is significantly lower, the proportion of correct candidates is much higher.

Also observable from the figure is that the precision performance dropped and became flat as the value of minCxtLen increased. This drop in performance results from the inability of the system to match longer contexts and the leveled performance reflects the number of characters beyond which no new information is obtained from the KB i.e. no new context matches can be found. If the size of the KB is increased, the minCxtLen at which the precision rates become flat can be expected to increase due to greater opportunities to find matches.

The performance of the system peaks when it is able to extract candidates using contexts that are not very particular to any single document (e.g. contexts containing proper names and long sentences) and contain enough information for disambiguating between them. The minCxtLen value at which the system performs best may be dependent on language, domain and/or the KB size. Therefore the minCxtLen values of 3 and 4 where the precision peaked in the above experiment cannot be assumed to be universal.

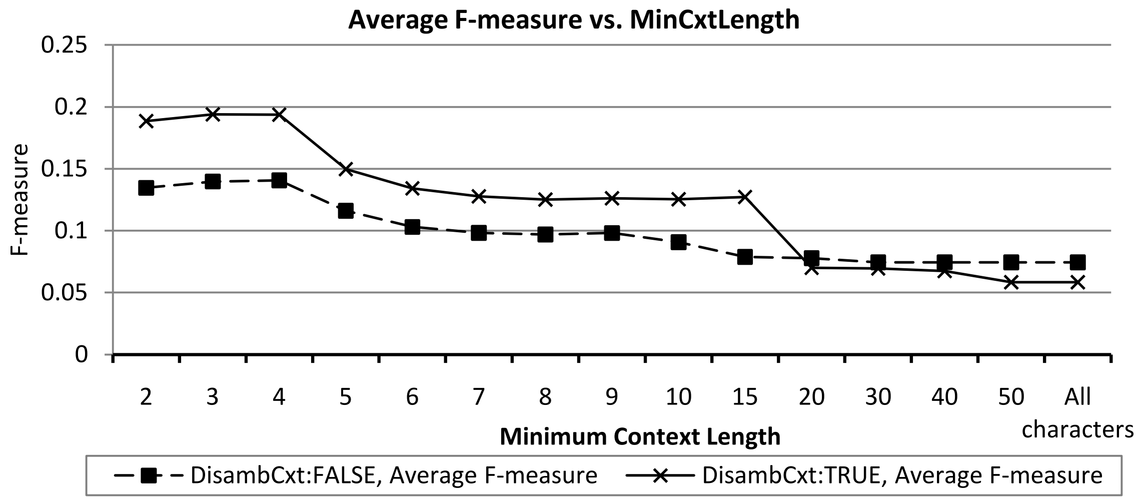


Figure 8.12: NER F-measure values at different minimum context length values (English Obituary)

The F-measure for both when regular and disambiguated contexts are used sharply decreased when the minCxtLen value exceeded 4. Based on this information, it can be concluded from Figure 8.12 that 4 is the optimal minCxtLen value for this particular KB (KB1) used in these experiments.

8.2.1.1.1 Results Discussion

Based on the total of 9600[10] NER experiments conducted and the results shown in Figure 8.10, Figure 8.11 and Figure 8.12, the system performed best when the value of minCxtLen was equal to 4. Hence, 4 has been chosen as the optimal minCxtLen value. Although, minCxtLen values of 2 and 3 result in similar performance, they have relatively longer execution times. Because the opportunity for matching long contexts in entirety is limited in small KBs, it is difficult to extensively examine the performance at high minCxtLen values (other than the entire context length). Therefore, a range of larger KBs should be considered in order to better understand the effect of minCxtLen on the performance.

Conducting similar experiments using KBs from different domains and languages would help in establishing if the optimal minCxtLen value obtained from these experiments can be considered universal. The above described experiments can help in establishing the optimal minCxtLen value of any KB in the absence of knowledge of the minCxtLen value recommended for the language or domain. Generally, it is suspected that the minCxtLen value is dependent on the average number of characters in a word for any given language and/or domain. Experiments with minCxtLen equal to1 were excluded

[10] 10 documents * 9 minCxtLen values * 8 KBs * 4 scoring methods * 2 context disambiguation settings

from the results and analysis because of the considerably longer execution time and relatively poor performance.

8.2.1.2 NER performance at different KB sizes relative to type of contexts used for candidate extraction (English Obituary)

Using the optimal minCxtLen value, 4 obtained from the above experiments, the performance of the NER for different KB sizes were examined for both regular and disambiguated contexts.

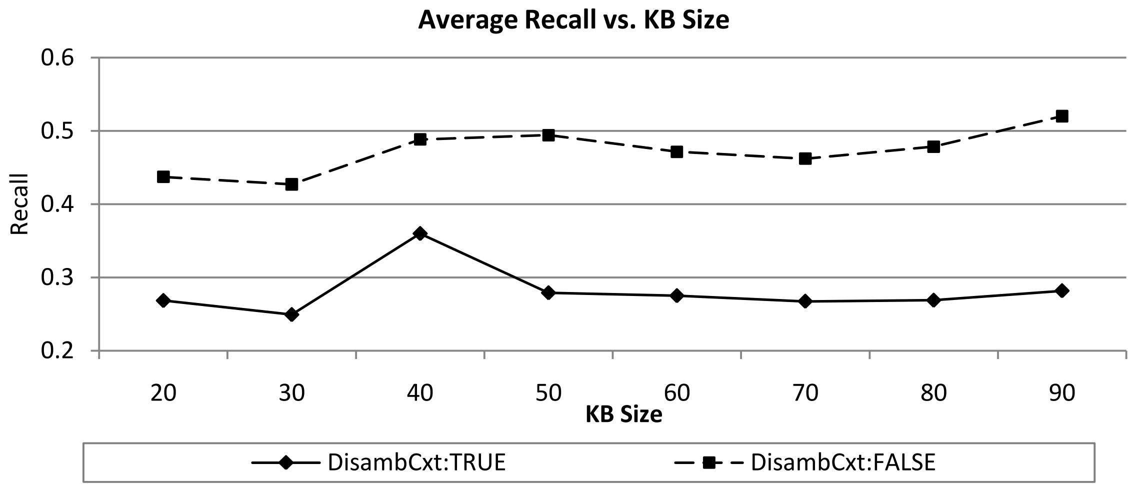


Figure 8.13: NER recall rates at different KB sizes (English Obituary)

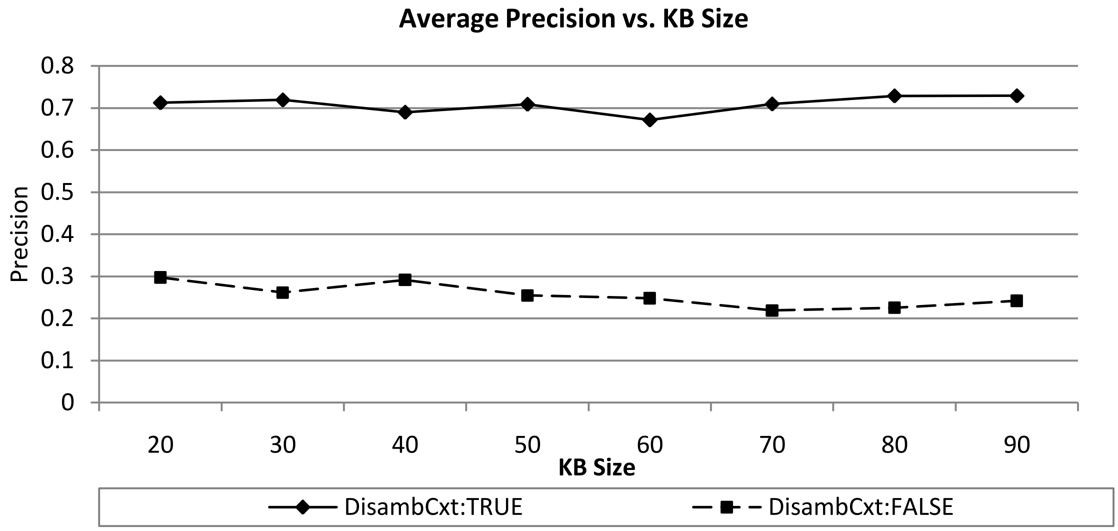


Figure 8.14: NER precision rates at different KB sizes (English Obituary)

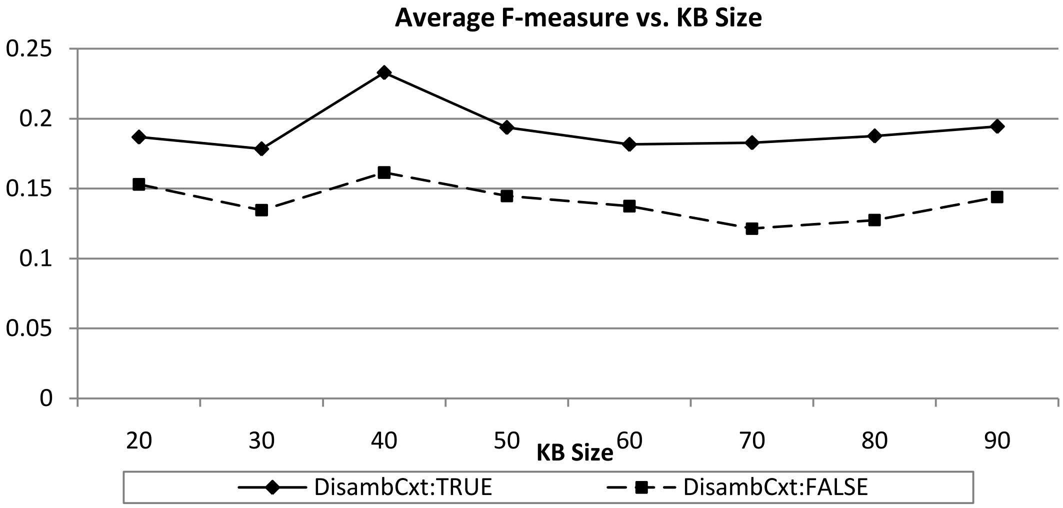


Figure 8.15: NER F-measure values at different KB sizes (English Obituary)

Although the upward trend in performance of the system with increasing KB size is not very visible in the above figures, the use of disambiguated contexts shows higher performance as the size of the KB increased compared to regular context usage. The increase in the size of the KB not only provides more opportunities to match more contexts, it also introduces the amount of ambiguous contexts available in the KB. Furthermore, the range of values of KB entities widens due to the increase in the number and classes of new entities. These two factors significantly contribute to the poor performance of the system when regular contexts are used for candidate recognition, especially since the levels of ambiguity of contexts are not considered in this method of the recognition process. At the same time, the use of disambiguated contexts does not completely eliminate this problem. However the system has more information with which to disambiguate among candidates.

The use of regular contexts shows a higher recall performance for the same reason as explained for Figure 8.10 and vice-versa for precision as explained for Figure 8.11. The sudden peak for KB size 40 is a result of randomness that occurred in the composition of the KB. The slow increase in the system performance (relative to the KB size) suggests among other factors that either a relatively much larger KB would be required in order to implement this NER approach efficiently or the context disambiguation methods needs to be enhanced. More research should be carried out to examine the causes and types of ambiguous contexts and more effective ways to manage them.

The wider similarity comparison range resulting from the increase in the number of KB entities resulted in a greater number of wrong candidate values were not successfully filtered. Decreasing the similarity comparison window as the KB size increases might provide solution to this problem.

8.2.1.2.1 Results and Discussion

A very visible improvement in the system performance resulting from KB size increase was not observed, although some improvement was noticed with the use of disambiguated contexts. The use of larger KBs and better handing of ambiguous contexts can help in better understanding this relationship. Based on the analysis of the experiment results, one can conclude that the use of disambiguated contexts is preferable to regular contexts usage.

8.2.1.3 NER performance by Candidate Scoring Method (English Obituary)

The performances of the NER using the 4 candidate scoring methods previously described were compared. The experiments with the minCxtLen value of 4 and the use of disambiguated context were considered.

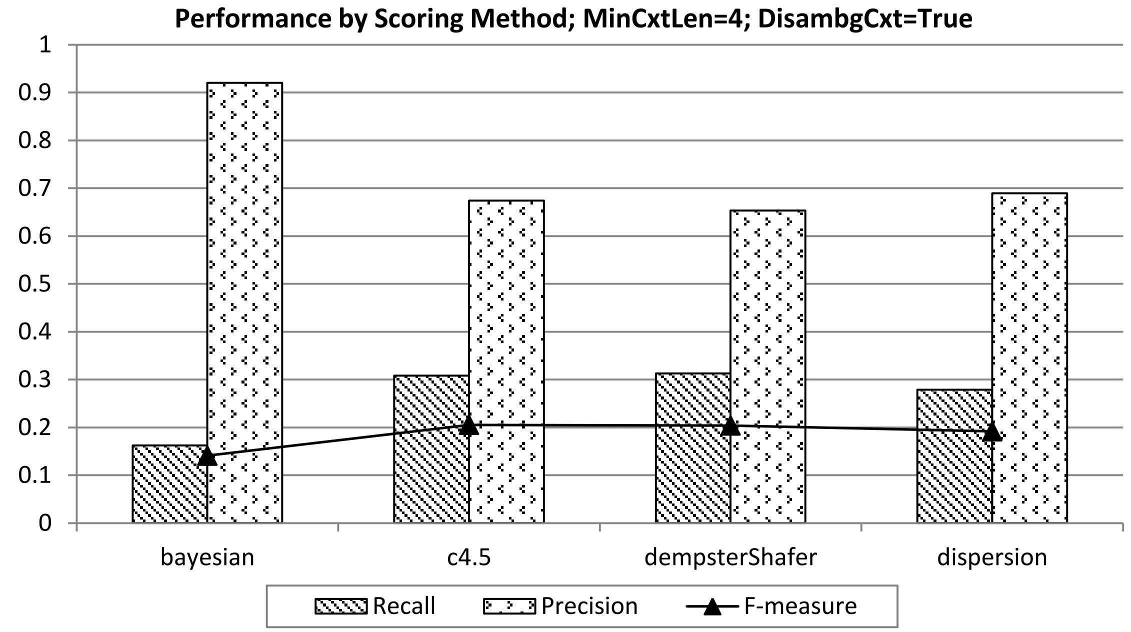


Figure 8.16: NER performance across different Candidate scoring methods (English Obituary)

Figure 8.16 shows the overall performance of the NER system relative to the different scoring methods examined. It can be observed that the Bayesian scoring method has the poorest f-measure performance. This poor performance is attributable mainly to the use of interchangeable contexts; the inability of the Bayesian scoring method to find fragments having both the LCxt and RCxt as the candidates, forces it to consider such candidates as incorrect. All other candidate scoring methods behaved comparatively similarly.

8.2.1.3.1 <u>Results and Discussion</u>

As shown in Figure B-3, Figure B-4 and Figure B-5 the performance of the different candidate scoring methods generally increased relative to the size of the

KB. The Bayesian method due to its very strict scoring conditions has the highest precision, however at the expense of recall.

8.2.1.4 Performance by Entity Class (English Obituary)

The 12 entity classes annotated in KB1 are Child, Decedent, Decedent Age, Decedent Date of Birth, Decedent Date of Death, Decedent Last Residence, Decedent Place of Birth, Grand Child, Parent, Sibling, Spouse and Spouse of Child. Regardless of the contexts used to locate and assign entity class to the candidate, the intrinsic characteristics of many of the candidate values are identical. As the intrinsic measurement is designed to help in identifying candidate values that have very similar character composition to KB entities of the same class, entity classes with unique and/or homogenous character compositions (e.g. age and dates) are expected to significantly benefit from this measurement.

Below, the measurement of the system relative to the entity classes extracted for both when regular and disambiguated contexts are used were compared. Again, only candidates extracted using the minCxtLen of 4 from the previous experiments were considered.

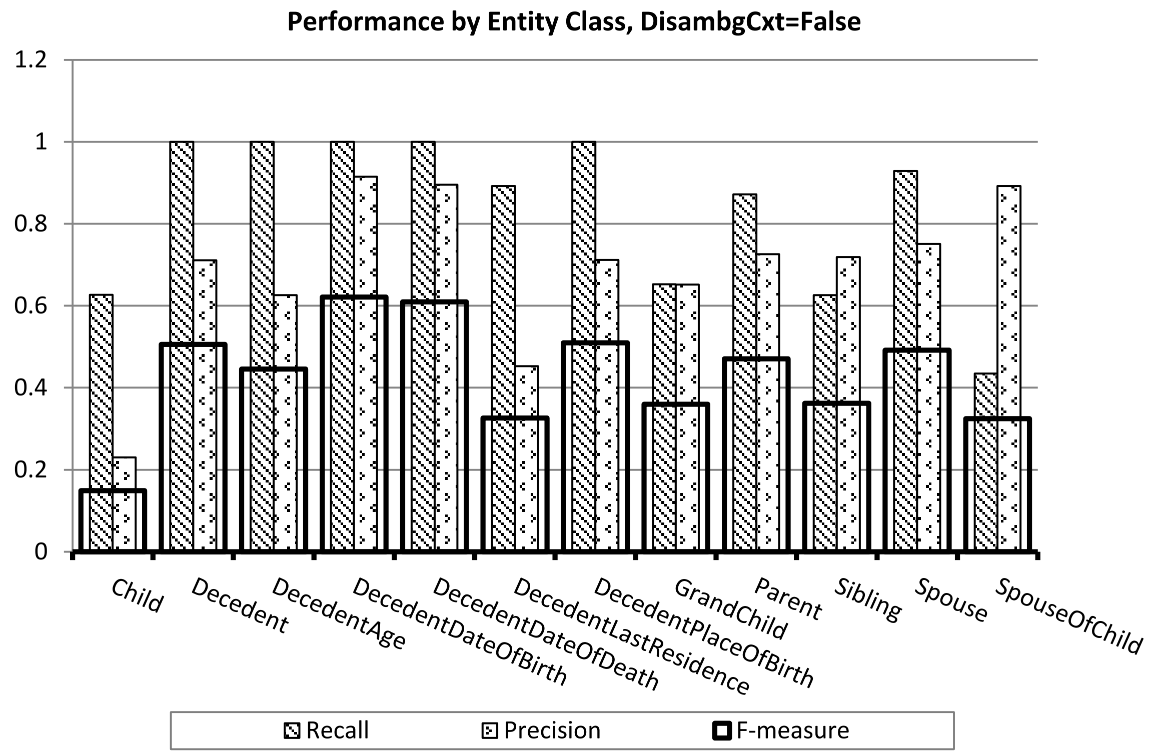


Figure 8.17: Performance of the NER system by Entity Class with regular contexts usage (English Obituary).

As can be observed in Figure 8.17, the date entity classes have the highest performances. Decedent age was expected to have a similar performance due to the homogeneity in its character composition (numeric) within the KB. However, because the intrinsic measurement is only taken after the candidate has been located using the contexts, it is difficult to attribute the poorer than expected performance of the decedent age solely to the intrinsic measurement.

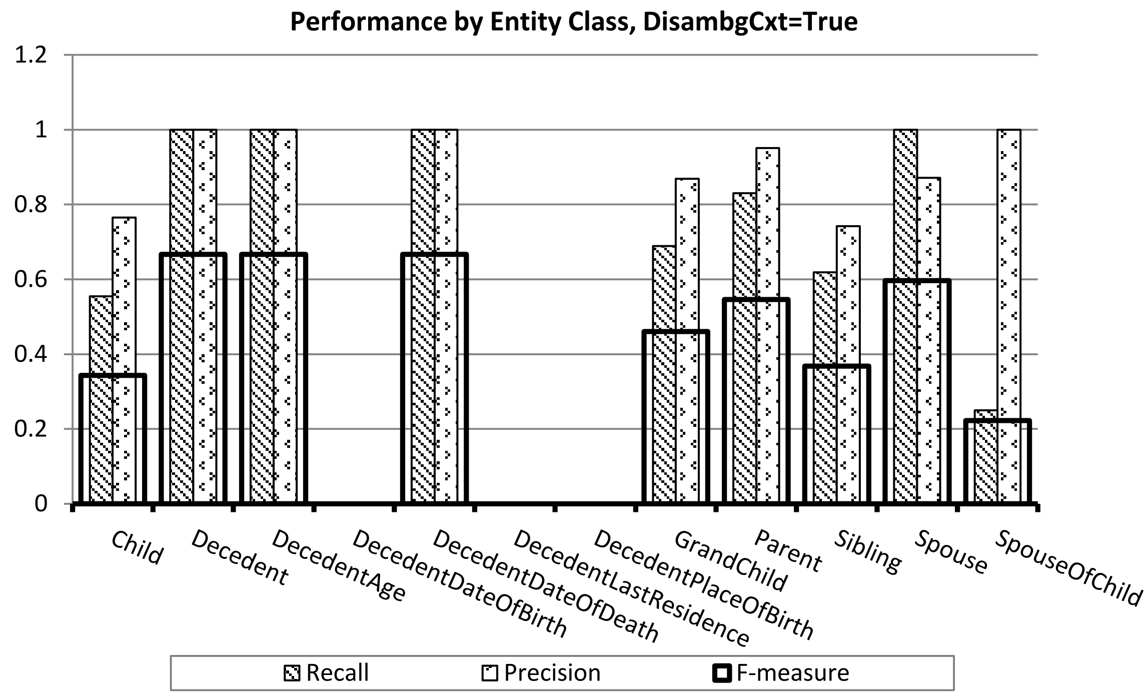


Figure 8.18: Performance of the NER system by Entity Class with disambiguated contexts usage (English Obituary).

As expected, the system performance for decedent age entity class is among the highest. This better performance relative to Figure 8.17 indicates the importance of the additional disambiguating information contained in the contexts. Because the effective use of disambiguated contexts depend on the prior recognition of the entity class used to argument certain contexts, the inability to recognize such entities makes the use of those contexts impossible. Consequently the recognition of other entities may be impossible. This factor is responsible for the inability of the system to extract the date of birth, last residence and place of birth entity classes. These 3 entity classes occur in very close proximity to one another in the KB and are often appended to the contexts for disambiguation. The inability in successfully identity one of them led to its inability to locate others.

The performance for parent, sibling and child entity classes in both experiments are similar (after the context differences are accounted for) because of the similarity in their character compositions. Other entity classes which have mostly single token (word) values (e.g. last residence, spouse of child and grandchild) reveal a much lower performance. This low performance can be attributed to their relatively less disambiguating token properties and character type composition. Majority of these entity types in the KB only have a single token and all alphabet characters, which makes them undistinguishable from any other word in the text.

The high performance of the decedent entity type is attributable more to the contribution of its contextual rather than its intrinsic properties. Although, the contexts play a strong role in the candidate correctness likelihood, the similarity of this entity values to many other values decreases the likelihood of the effectiveness of the intrinsic measurement in effectively filtering out candidates.

8.2.1.4.1 Results and Discussion

Considering that the user may have deep knowledge of the domain and ontology, the intrinsic characteristics of certain entity types may be used interchangeably with one another e.g. Decedent name and parent name. If the intrinsic measurements are understood to be poor, more emphasis may be placed on the level of ambiguity of the contexts used in locating the concerned entity classes.

8.2.1.5 Relationship between the Range of Comparison and NER Performance (English Obituary)

In this experiment, the NER performance using different similarity comparison ranges were examined. A good understanding of the behavior of these ranges can help in better disambiguating between candidates as the KB size changes. A comparison range too small will potentially cause the rejection of candidates that do not have almost identical values to those in the KB. On the other hand, excessively wide range risks high recall and low precision rates.

The NER system performance was examined using the Dempster-Shafer scoring method, minCxtLen value of 4 and disambiguated contexts (highest performances from the previous experiment).

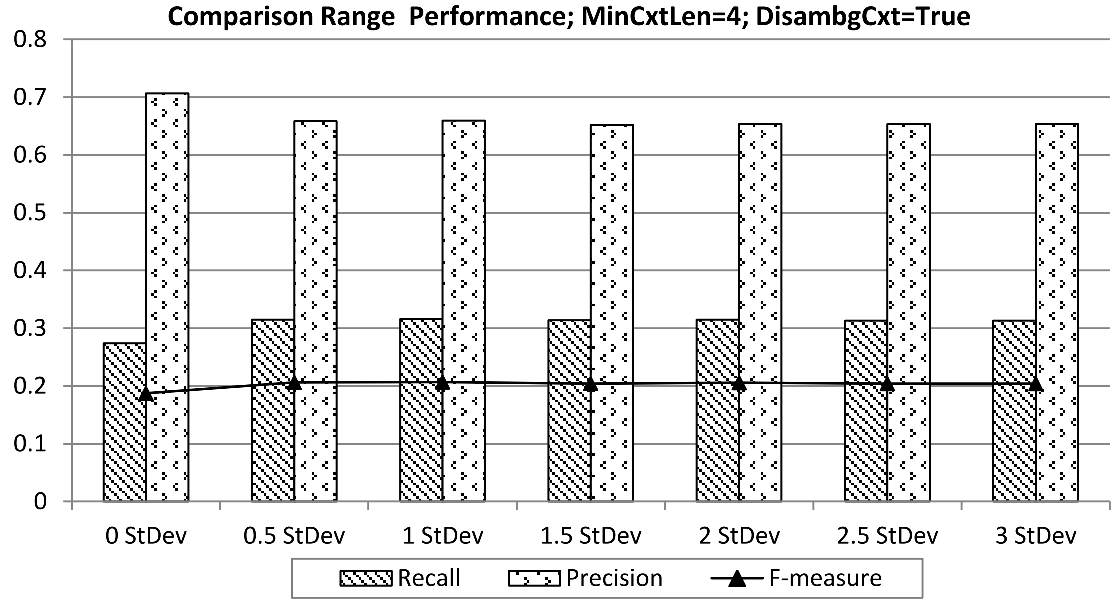


Figure 8.19: NER performance in relation to comparison range (English Obituary)

The result shown in Figure 8.19 it indicates that the system performs uniformly for almost all the different range values tested. These results are the consequence of the approximations that occur when the range is translated into actual comparison values. For example, ranges of 1, 1.5 and 2 standard deviation may translate to token ranges of [1.5, 2.5], [1.3, 2.7] and [1.1, 2.9] respectively. However, since a word counts are integers, all the above ranges are converted to [1, 3] tokens, implying that the system performance of all the different ranges would be identical.

As evident from Figure 8.20, Figure 8.21 and Figure 8.22 this range translation varies almost identically at different KB sizes with the exception of 0 standard deviation factor. Unfortunately, it is difficult if not impossible to conclude any meaningful information from this experiment. Different methods to vary the comparison ranges would be researched during future studies.

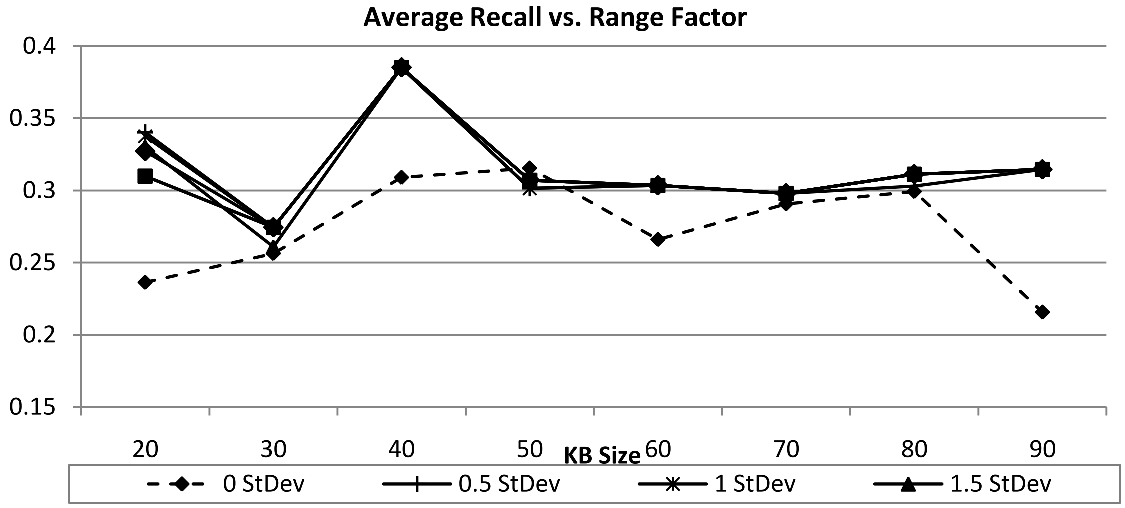


Figure 8.20: Recall performance in relation to comparison range (English Obituary)

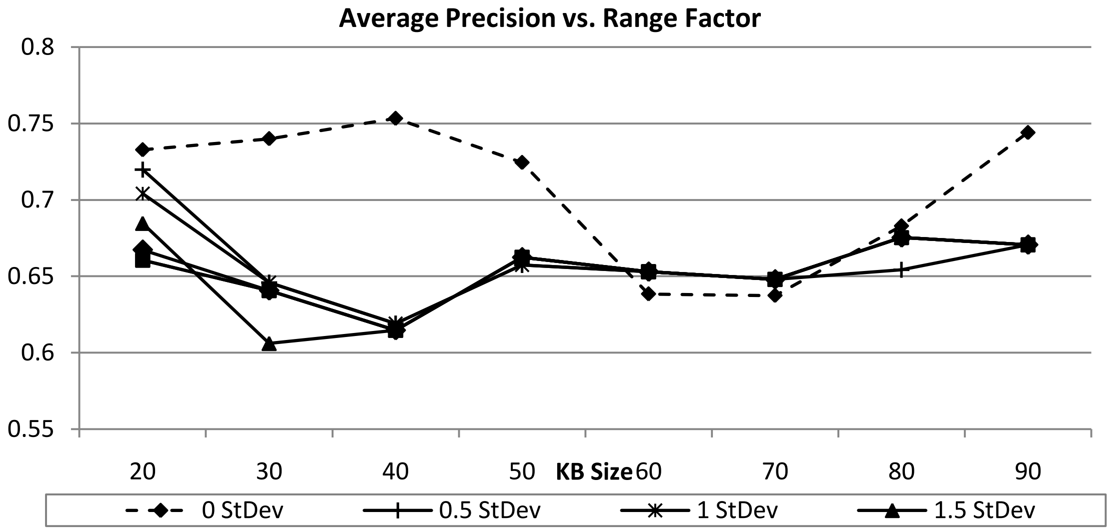


Figure 8.21: Precision performance in relation to comparison range (English Obituary)

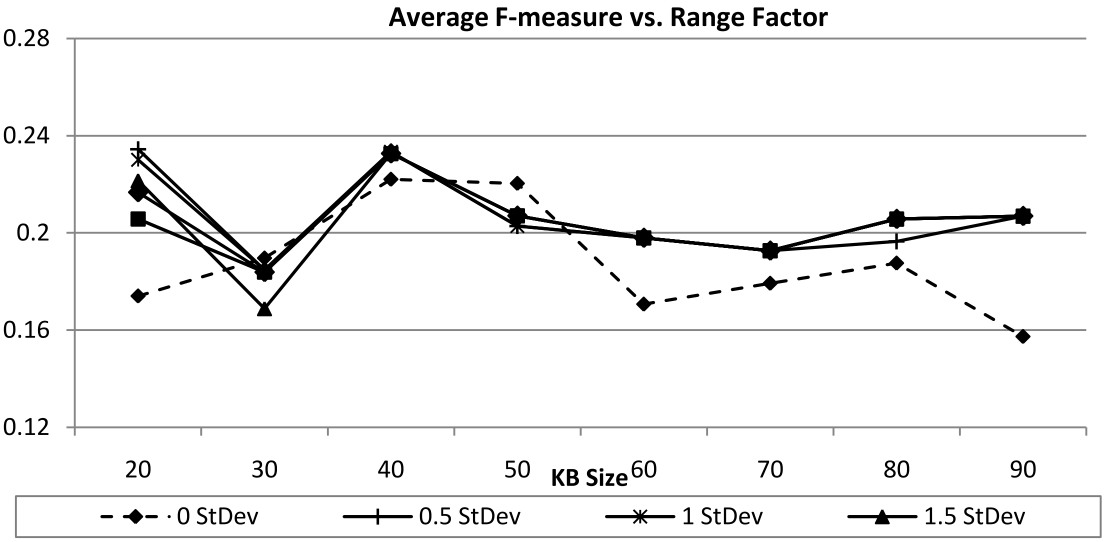


Figure 8.22: F-measure performance in relation to comparison range (English Obituary)

8.2.1.5.1 Results and Discussion

This experiment is quite inconclusive as a definitive trend in the behaviors of the different comparison ranges for different KB sizes cannot be identified. Larger KBs and other methods of varying the range might help in empirically determining the behavior of this system attribute. This would be taken up during future studies.

8.2.2 Set-up 2: : Experiments with Spanish Documents

The second round of experiments was performed with 49 Spanish language crime reports (See example: Figure B-2) also obtained from Internet websites (KB2). Several of the experiments performed on KB1 were repeated i.e. experiments to determine the optimal minCxtLen value, examine the performance relative to KB sizes, scoring methods and entity classes.

8.2.2.1 *Relationship between NER performance and minimum number of context characters (Spanish Crime Report)*

The experiments described below were conducted using a comparison range factor of 3 standard deviations.

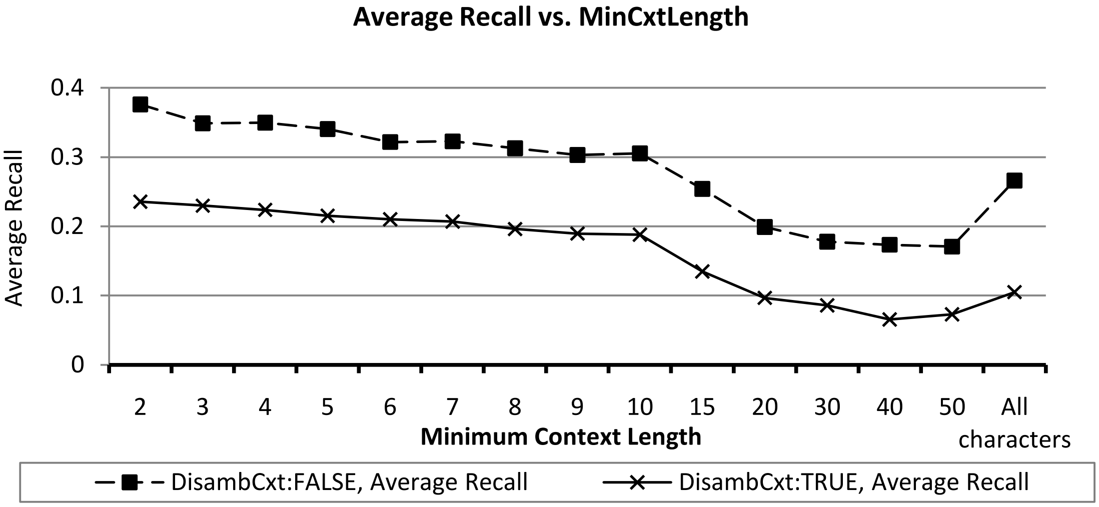


Figure 8.23: NER recall rate at different minimum context length values (Spanish Crime Report)

Similar to the recall rate of the English obituary announcements, regular contexts usage have higher performance compared to disambiguated contexts usage. However, unlike the KB1 experiments the recall suddenly increased with the use of the entire contexts. Although the reason for this increase is not fully understood, it is evident that more information (correct candidates) was extracted when the entire contexts were used compared to when 50 context characters were used. Also Figure 8.24 indicates that when regular contexts were used, the number of candidates returned increased as the value of minCxtLen increased (contrary to the English documents). An explanation for this observed phenomenon is that the higher number of candidates originally extracted using shorter contexts were dropped during the NER process due to zero scores resulting from the level of context ambiguity and the failure to pass the intrinsic measurements.

In Figure 8.24, the use of disambiguated contexts again outperformed regular contexts usage. Similar to the recall chart above, the increase in the precision rate at higher minCxtLen values would require further investigation.

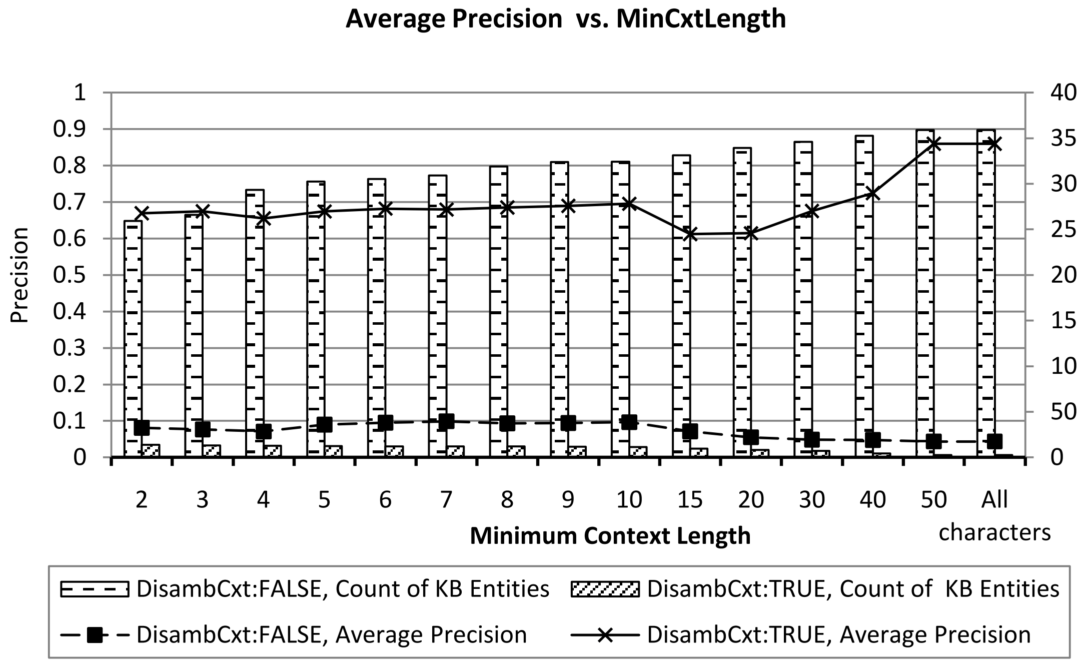


Figure 8.24: NER precision rate at different minimum context length values (Spanish Crime Report)

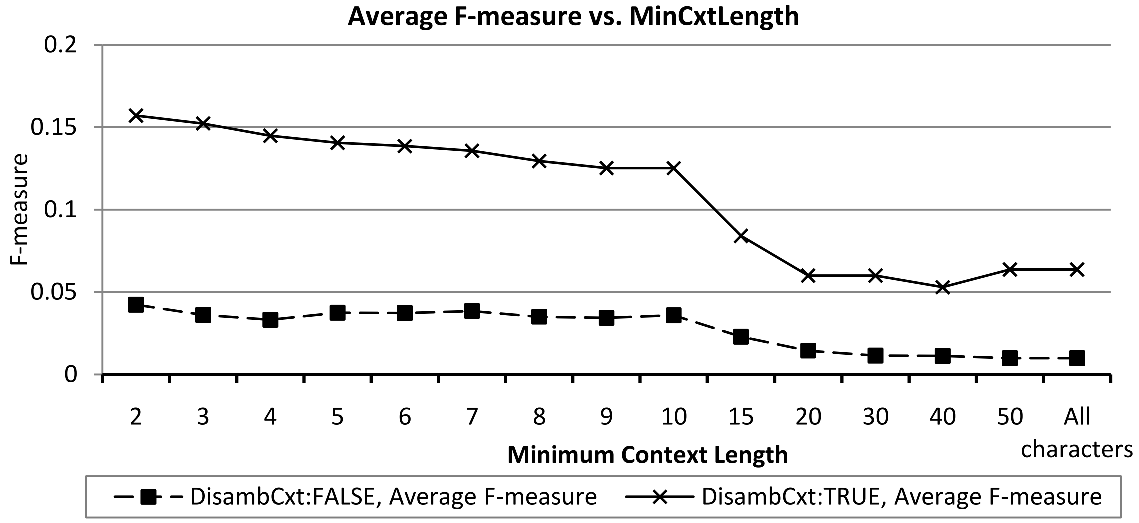


Figure 8.25: NER F-Measure rate at different minimum context length values (Spanish Crime Report)

As stated during the analysis of the KB1 minCxtLen experiments, the optimal minCxtLen value may be language and or domain dependent. Figure 8.25 indicates that KB2 performs best when the minCxtLen value is equal to 2. A conclusion that can be drawn from both the KB1 and KB2 experiments is that the minCxtLen value varies from one KB to the other. Consequently, an experiment to determine an optimal minCxtLen value for a KB is advisable before it is used for candidate extraction.

8.2.2.2 ***NER performance at different KB sizes relative to type of contexts used for candidate extraction*** **(Spanish Crime Report)**

The improvement in performance as the size of the KB increases is more visible in KB2 experiments even as the general behavior of the system remains the same i.e. regular contexts usage have higher recall than disambiguated contexts and vice-versa for precision.

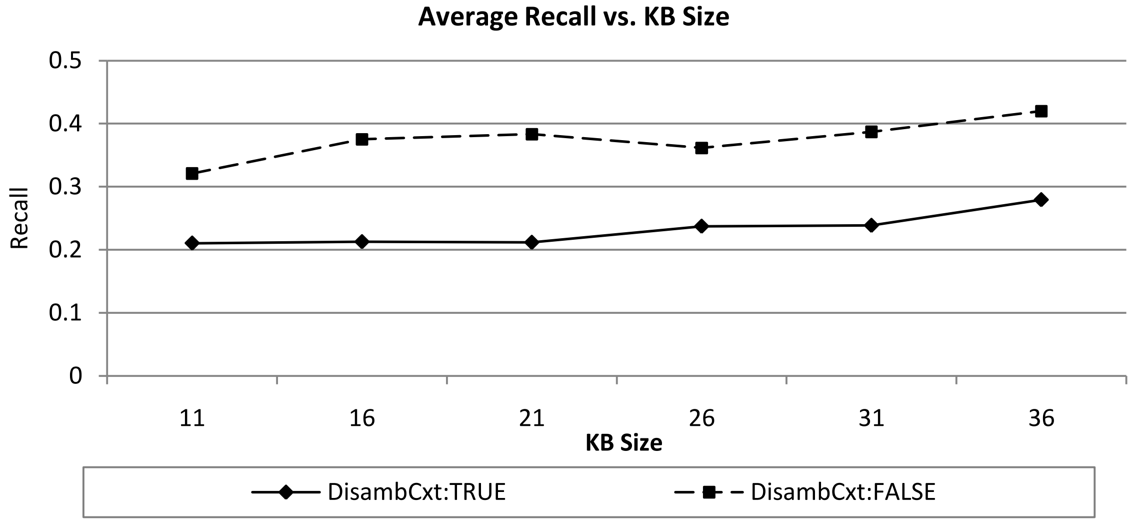


Figure 8.26: NER recall rates at different KB sizes (Spanish Crime Report)

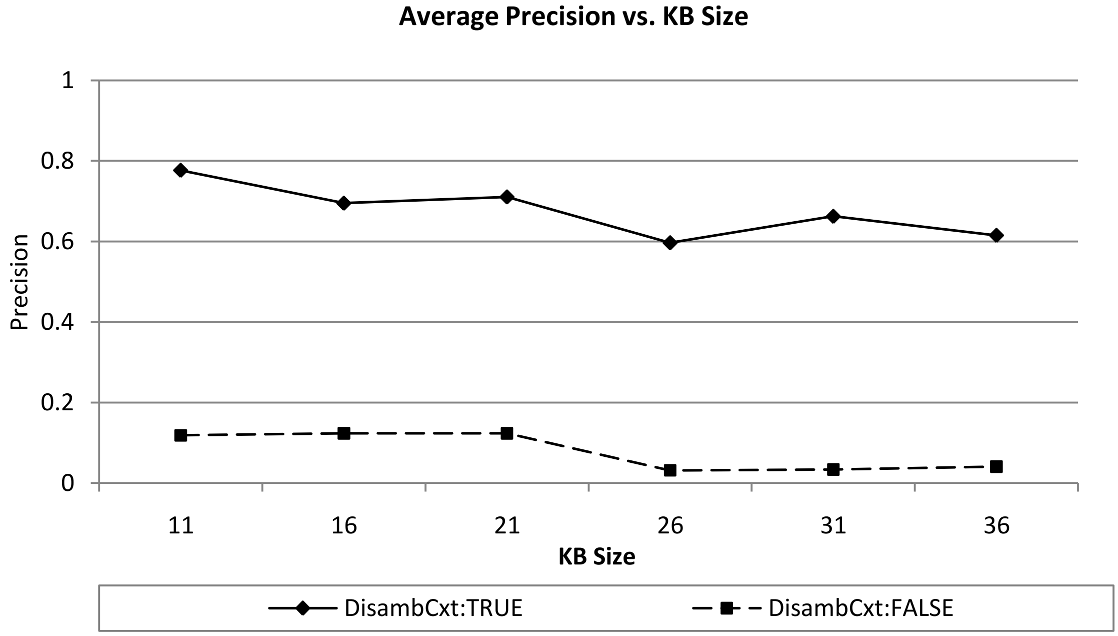


Figure 8.27: NER precision rates at different KB sizes (Spanish Crime Report)

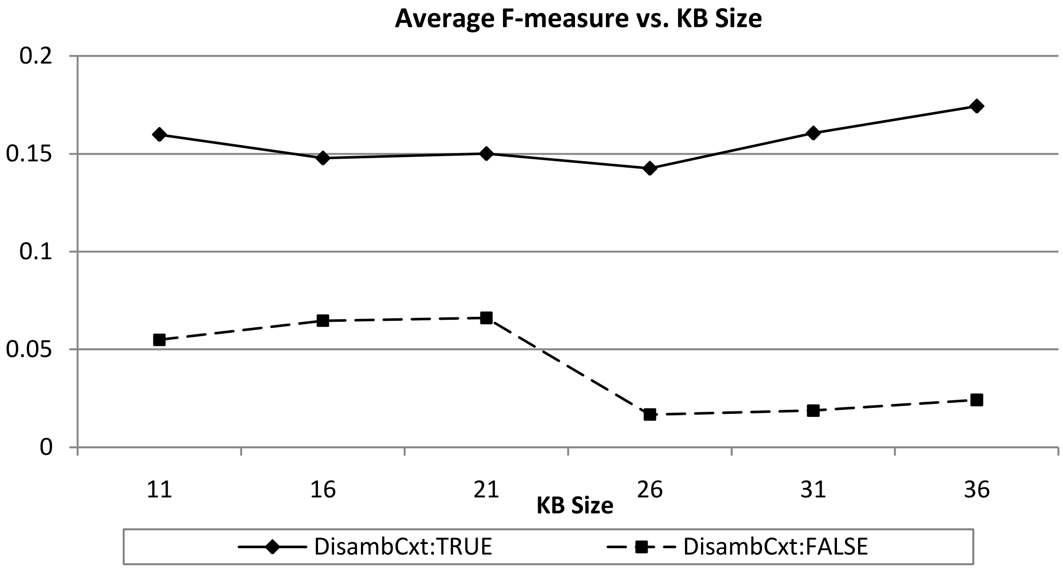


Figure 8.28: NER F-measure values at different KB sizes (Spanish Crime Report)

The sudden drop in the performance at KB size 26 can be explained using the same reason for the sudden spike in performance in KB1 (KB size 40). Using the average of several KBs of the same size would help in observing a smoother curve modulation.

8.2.2.3 *NER performance by Candidate Scoring Method (Spanish Crime Report)*

The performance of the different scoring methods is identical in both KB1 and KB2 experiments.

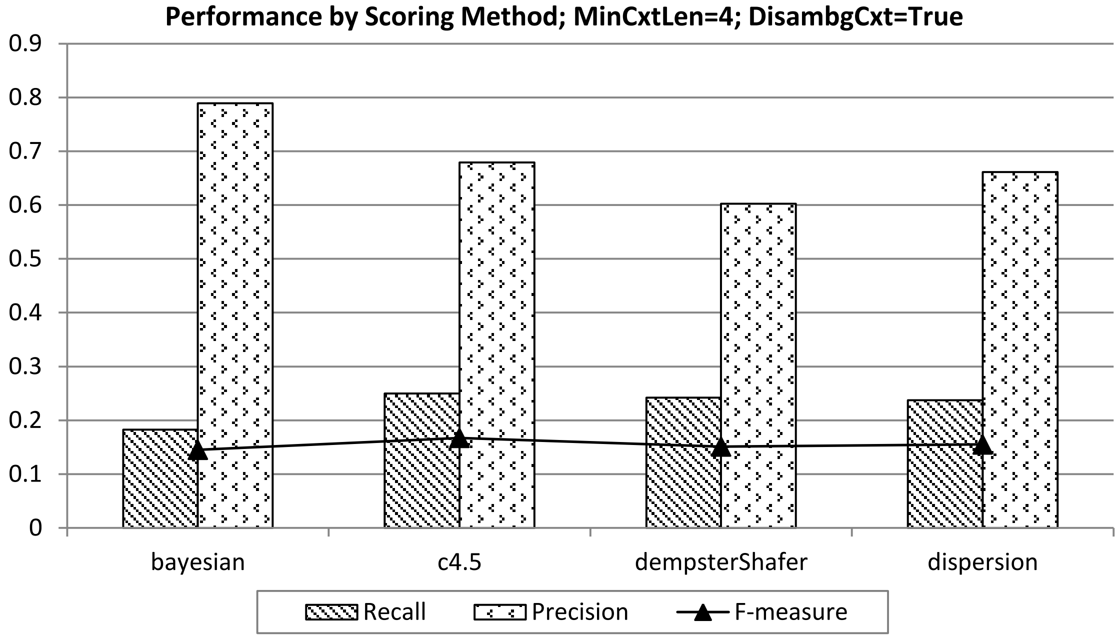


Figure 8.29: NER performance across different Candidate scoring methods (Spanish Crime Report)

8.2.2.4 Performance by Entity Class (Spanish Crime Report)

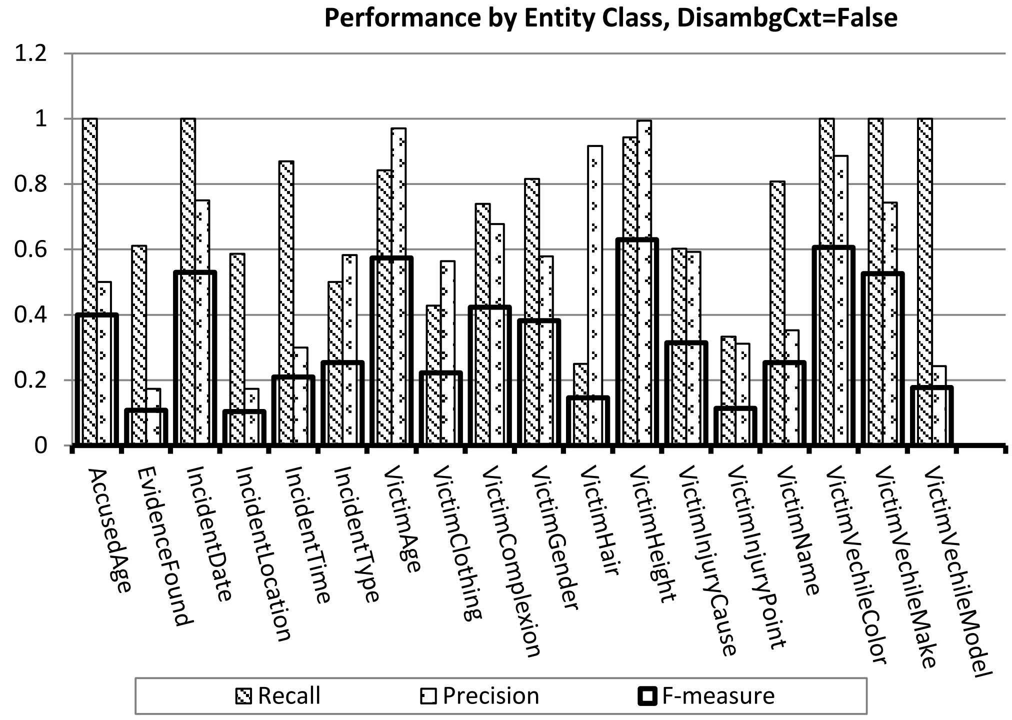


Figure 8.30: Performance of the NER system by Entity Class with regular contexts usage (Spanish Crime Report)

Entity classes with unique and/or homogenous character composition exhibits high performance just as observed in KB1 experiments; specifically, the date and height entity classes have high scores as expected. Although the age entity class also performed relatively well, a poorer performance compared to KB1 was expected because in KB2 the ages were often declared as ranges and not just numeric characters as in KB1. Figure 8.31 shows that similar to KB1, the use of disambiguated contexts although results in general better performance, it fails to extract some entity classes for the same reasons described in section 8.2.1.4.

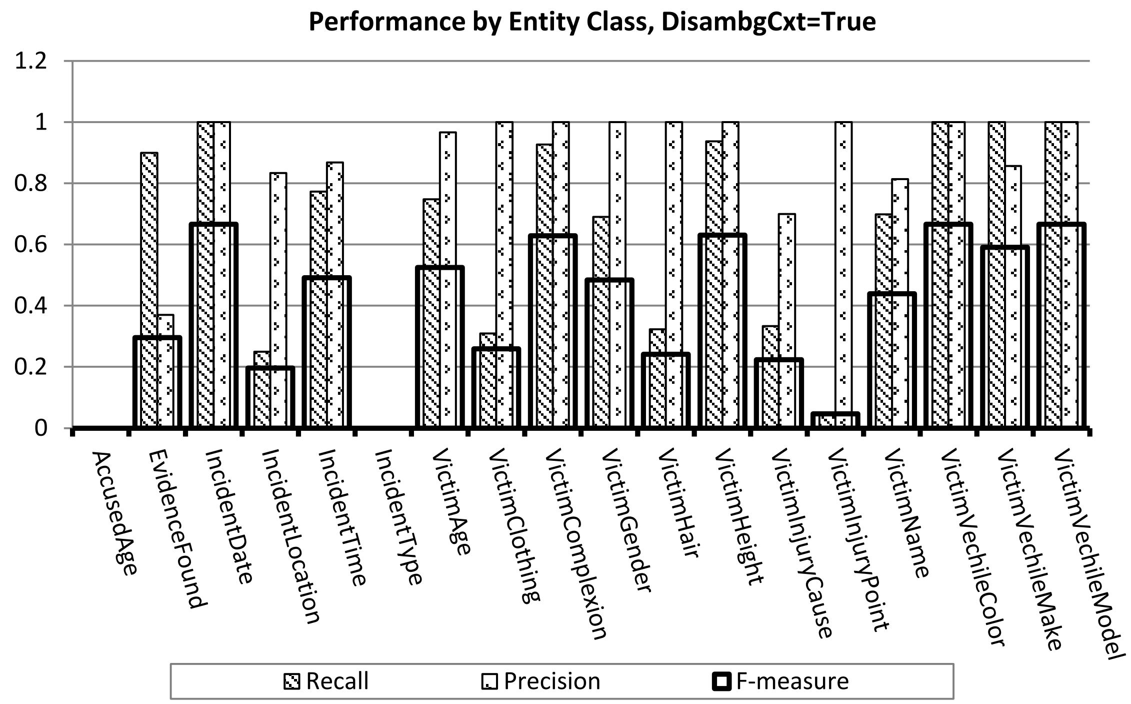


Figure 8.31: Performance of the NER system by Entity Class with disambiguated contexts usage (Spanish Crime Report)

8.2.2.5 Relationship between the Range of Comparison and NER Performance (Spanish Crime Report)

As can be observed from Figure 8.33, Figure 8.34 and Figure 8 35, all the different comparison ranges behaved similarly except for the 0 standard deviation comparison range factor. These behaviors are similar to that exhibited during the KB1 experiment and can be similarly explained.

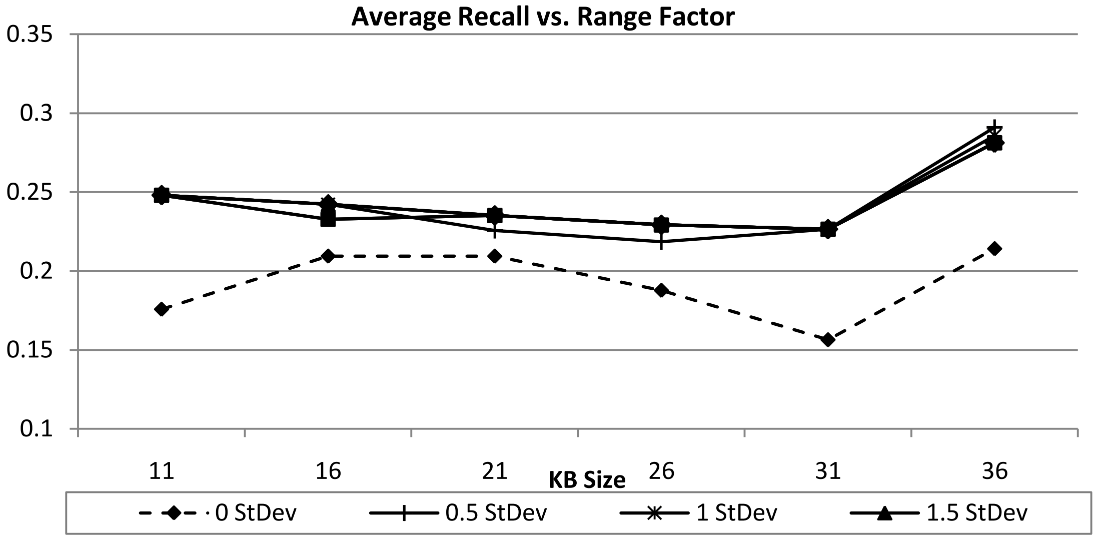


Figure 8.32: Recall performance in relation to comparison range

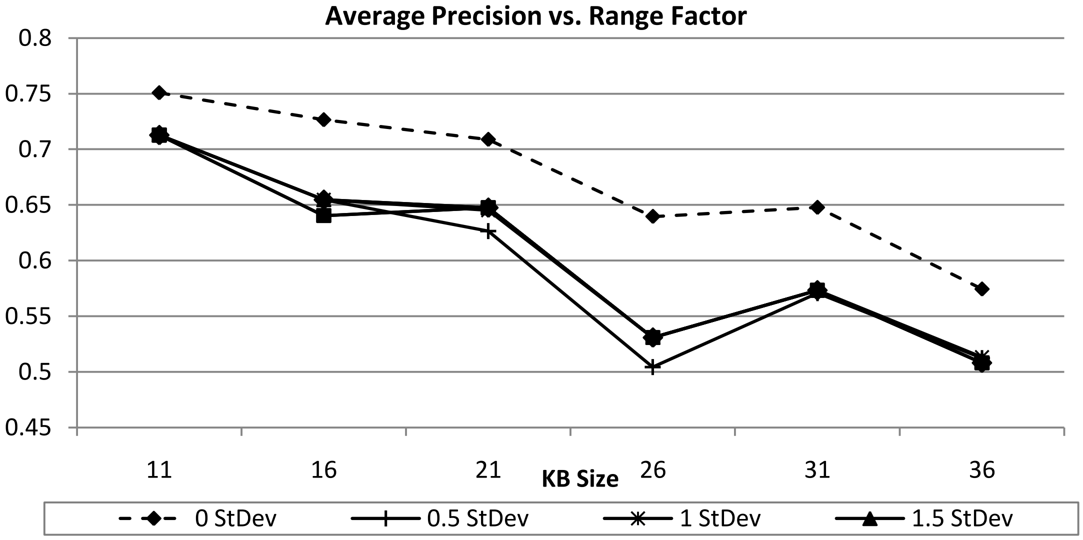


Figure 8.33: Precision performance in relation to comparison range

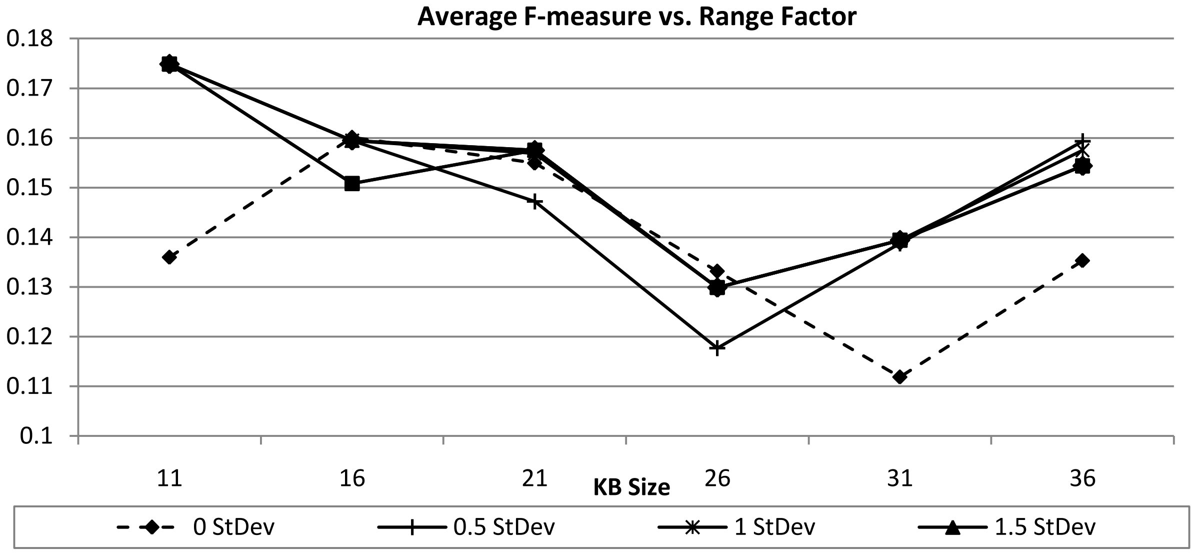


Figure 8.34: F-measure performance in relation to comparison range

8.3 General Performance Evaluation

In general, relatively high precision scores were achieved during the experiment i.e. the probability that an extracted candidate is correct is very high. On the other hand the recall rate was much poorer, meaning that the likelihood of extracting all the entities in the target document is low. The best F-measure performance recorded was 0.44 (recall=0.62, precision=0.9)[11] and the worst F-measure recorded was 0.008 (recall=0.0.48, precision=0.09)[12]. As previously stated at the introduction, the unavailability of publicly shared semantically annotated corpus makes the direct comparison of these results sets to those of other NER systems very difficult.

[11] Document 10, KB size=40, minCxtLen=5 and scoring method = dispersion.

[12] Document 6, KB size=50, minCxtLen=30 and scoring method = dempsterShafer.

8.4 Language and Domain Independence Test

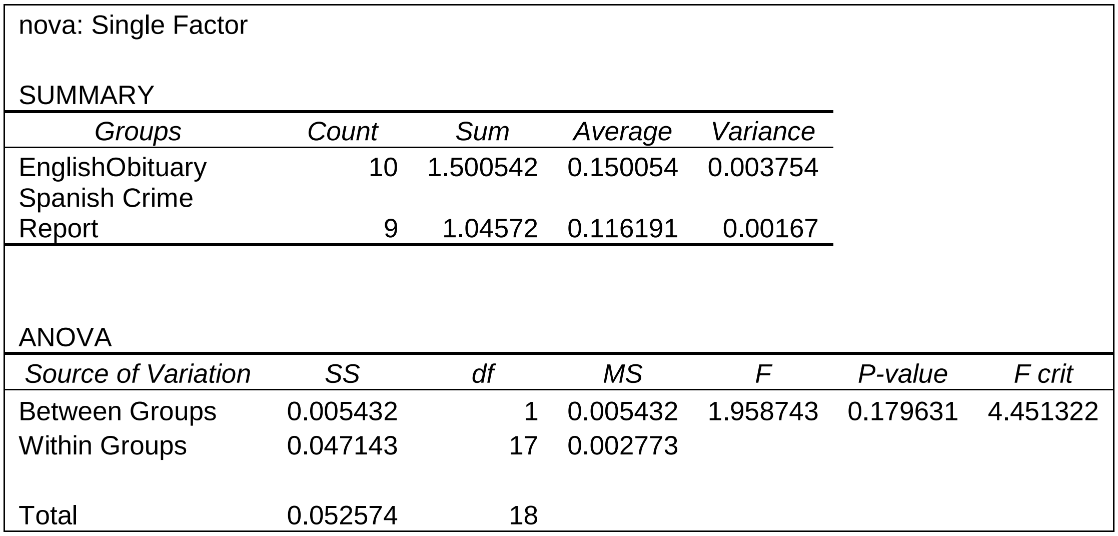
In order to examine the similarity of the NER performances between the two different languages and domains, the F-measure values of the NER performance on the English language obituaries and Spanish crime reports were compared. The average F-measure values of the NER performance on 9 documents using 6 different identical KB sizes of the two different sets of data, 15 different minCxtLen values and disambgCxt=true setting were compared (see Table 8.1).

DocID	English Obituary	Spanish Crime Report
1	0.116884	0.075129
2	0.077143	0.177939
3	0.122092	0.0625
4	0.12674	0.086581
5	0.164032	0.116797
6	0.162152	0.142969
7	0.120626	0
8	0.174916	0.170876
9	0.131406	0.096397
10	0.30455	0.116531

Table 8.1: Average F-measure values of 9 documents for the 2 different data sets.

The null hypothesis is that the means of the two experiments are equal. If this null hypothesis holds, then the differences in mean F-measures between the

English obituary announcements and Spanish crime reports seen in Table 8.1 have no statistical significance. Although a more extensive experiment using documents from several more languages and domains is recommended before arriving at a more conclusive language and domain independence tests, this experiment indicates that this assumption may be viable.



nova: Single Factor

SUMMARY

Groups	Count	Sum	Average	Variance
EnglishObituary	10	1.500542	0.150054	0.003754
Spanish Crime Report	9	1.04572	0.116191	0.00167

ANOVA

Source of Variation	SS	df	MS	F	P-value	F crit
Between Groups	0.005432	1	0.005432	1.958743	0.179631	4.451322
Within Groups	0.047143	17	0.002773			
Total	0.052574	18				

Table 8 2: Anova Test Results

The average F-measure for the English obituaries and Spanish crime reports used for this experiment are 1.50 and 1.05 respectively. According to the test result F=1.96. With a critical value of 0.05, the critical F=4.51. Therefore, since the F statistic is less than the critical value, the null hypothesis is accepted. In other words, the NER system behaved equivalently across the two languages and domains tested.

Figure 8 35 shows the distribution of the F-measure values for both document categories. Further analysis would be required in order to explain why the values skew to different side of their respective means.

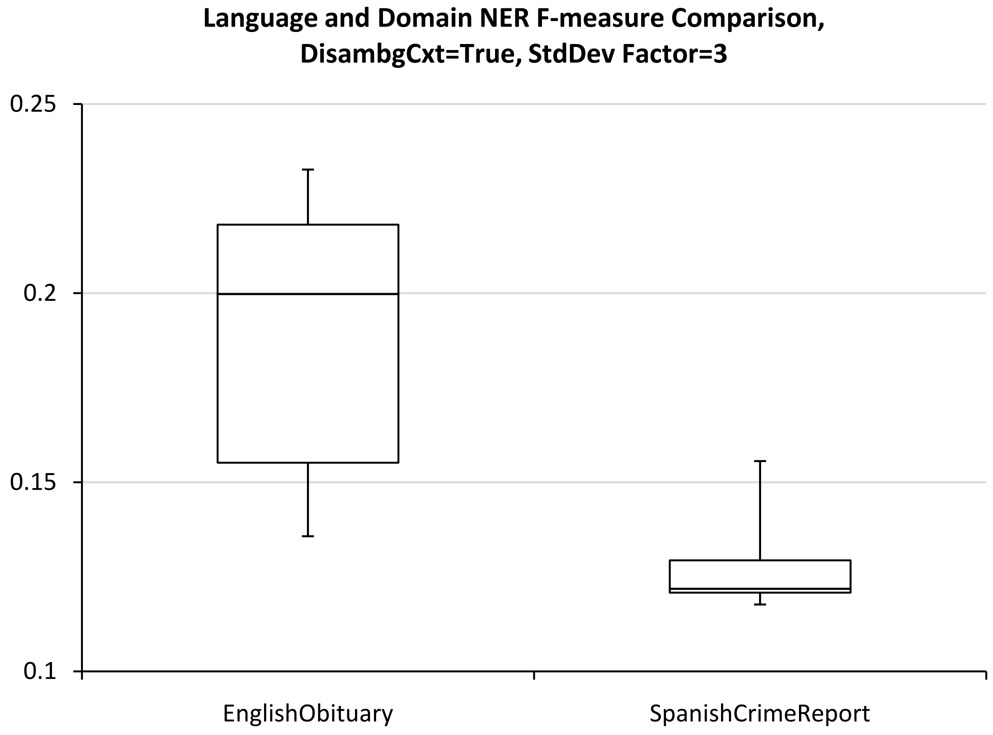


Figure 8 35: Average F-measure value distribution for the English obituaries and Spanish crime reports

8.5 High Performance Computer Acknowledgement

This work was supported in part by the National Science Foundation under Grant CRI CNS-0855248, Grant EPS-0701890, Grant MRI CNS-0619069, and OISE-0729792.

Chapter 9

Discussions, Future Work and Conclusion

9.1 Discussions

9.1.1 Applications

This data-intensive approach to NER has several possible applications. Its feature of flexibly assigning semantic roles to recognized entities within different domains and languages makes it deployable in different organizational environments. The transformation of the information contained in UTI into a structured format such that it can be understood in a transactional manner e.g. the wedding announcement depicted in Figure 4.1 can be stored in a relational table (wedding) with the different entities representing the table columns enables users to apply the already well developed (within many organizations) analytical skills and tools to the information.

Current research involves the application of this NER tool within a law enforcement environment for the purpose of identifying critical information about crimes from public sources such as blogs and blog comments, social media as well as news reports. For example, a single incident described by two or more different internet users each possibly having different vital information about the incident. Since the text from the different users will very likely not be verbatim, effectively comparing the information and reconciling/resolving them where necessary without human reading of the text can be done by using this NER tool to recognize the entities of interest in each text. The recognized entities can then be structured (e.g. using relational database tables) and then resolved (entity resolution). This will enable law enforcement organizations to more efficiently use crowd knowledge for sourcing information on crime fighting , especially in situations where witnesses may feel more protected from retributions by anonymously posting information online than by speaking directly to law enforcement agencies.

Similar to the law enforcement use described above, organizations may also use this NER tool on other types of unstructured text like contracts, e-mails etc to reduce the amount of human reading necessary in order to obtain information about entities and their interactions from them. Other uses of the NER tool include creating resource description framework (RDF) for semantic web and multi-level indexing of the information for tasks such as information retrieval and analytics.

Although some newly developed websites comply with semantic web specifications by having the conceptual description of the content described in embedded RDF documents. This RDF information helps search engines to provide a richer user experience by identifying websites containing the semantic information in the search query. For example, a RDF document may indicate that the word “Washington” on a webpage indicates a person and not a geographical location. Millions of existing websites and other online resources however do not currently comply with this specification in that they do not contain RDF information. This NER tool can be used to automatically create RDF for such websites and online resources, thereby ensuring that the information in such systems can be retrieved with more certainty. An advantage of using this data-intensive NER approach is that the RDF documents for new web resources can be created using existing similar RDF information. For example, RDF information of a set of website can be used to annotate and create RDFs for content of other web pages of other similar websites. Other than the automatic creation of RDF documents, it also enables web developers to automatically categorize their websites with those websites contained in the KB.

Another possible application of this research is the development of semantic search engines that are able to retrieve information using multi-level semantic indexing. Such an application would involve the representation of unstructured website content in relational database tables to be indexed for semantically named entities of interest. This indexed information can then be used to provide options that users may use to make more search queries more specific. Corporations such as Endeca and Radar Networks are currently using proprietary software to deliver similar solutions. Endeca announced the launch of its McKinley release of the Endeca Information Access Platform called newssift in March, 2009 (Endeca Technologies, Inc, 2009) and Radar Networks announced the features of its T2, Twine Semantic Search engine in September, 2009 (Schonfeld, 2009). Both search engines provide users with the experience of “faceted search”. Their search engines suggest categories users can use to filter their search until the scope is narrow enough to only return a few hits. Using this NER tool provides users with the option of choosing/entering attribute values that can significantly narrow down the amount of information retrieved and increase the likelihood of relevant documents being returned.

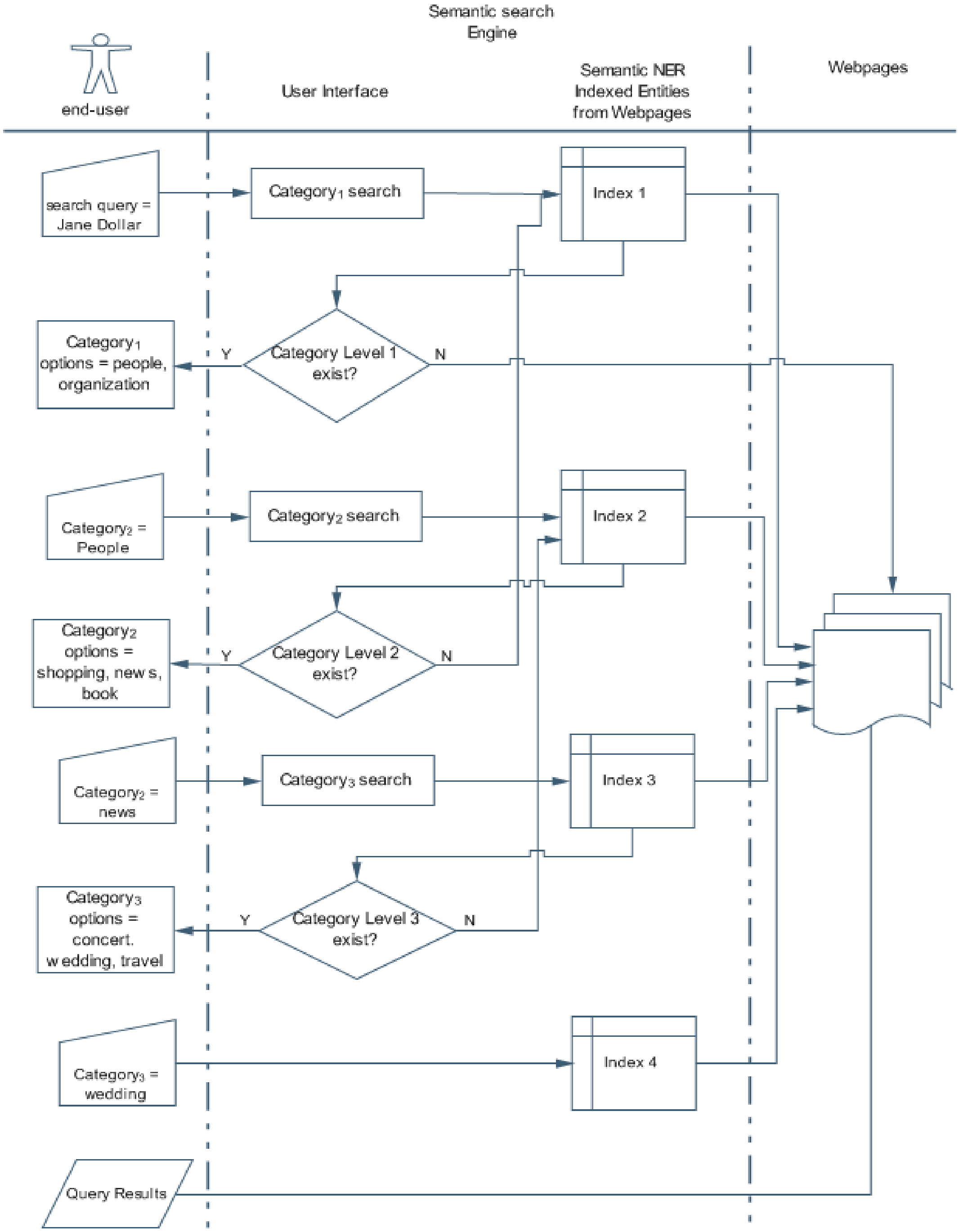


Figure 9.1: Example of a Semantic Search Engine Workflow

9.1.2 Opportunities

Because this NER approach is still in its infancy stage, it is early to fully enumerate its opportunities (and/or limitations). This opportunities (and/or limitations) would be discovered to a fuller extent with further research.

In addition to the opportunities previously mentioned, this data-intensive approach to NER provides the opportunity to increase collaboration among diverse actors in the human language technology community. Because the candidate location heuristics and selection are language and domain independent, NER researchers from different language and domain background can more easily share information on improving the system. An industry specialized in the compilation of KBs for different domains and languages may also emerge; making it possible for organizations looking to apply this NER approach to buy an off the shelf KB or outsource the compilation of a robust KB to corporations in such an industry.

From an information quality perspective, the possibility of an organization’s unstructured textual information being made available for automatic routine analysis hence more inclusion in business intelligence and decision support increases the value of the organizational data asset and its information quality.

9.1.3 Limitations

Some of the immediately visible drawbacks of this NER approach include the manual annotation of the example documents. Human errors such as inconsistent labeling or mislabeling of entities may reduce the performance of the system. On the other hand semantically labeled example documents can be easily retagged as other more general entity types (as long as the metadata for the tags are available). E.g. In a wedding announcement, the entities bride, groom and bridesmaid can be regrouped as people for a different task. This re-use opportunity (not backward compatible) enables the cost of KB creation to be shared across different applications.

This NER approach is not suitable for unstructured text with few characters e.g. tweets, mobile phone text messages and chat messages because there often isn’t enough contexts from which the semantic meaning of entities can be inferred. In order to more efficiently navigate through and increase the opportunities from such UTI, visualization techniques that aggregates and displays the contents of UTI in relation to information such as geographical and time to enable users to more easily access the content were explored (Osesina, et al., 2010) (Osesina, et al., 2010).

Finally, unstructured text with significant number of unusual abbreviations or slangs might also affect the NER performance unless the abbreviations and slangs are used consistently a large number of times.

9.2 Future Work

Some of the future works on this NER research include exploring more methods of candidate scoring. For example, the inclusion of the candidate scores of entity tags used in locating other candidates (when using disambiguated contexts) might help in reducing the error that might be propagated by a misidentified candidate.

The consideration of additional KB characteristics in the candidate location heuristics may lead to a higher likelihood of extracting correct candidates. For example, the matching of two consecutive fragments simultaneously may help in increasing performance since the candidate location conditions would be relatively stricter.

Furthermore, more experiments using test documents from different languages and domains would be conducted. These experiments would provide more empirical evidence of the language and domain independence of the proposed NER approach.

9.3 Conclusion

A novel data-intensive approach to NER has been introduced in this dissertation. This approach unlike previous NER approaches does not require hand written grammar rules, external lexical resources, or statistical algorithms for extracting named entities from unstructured text. Furthermore the presented NER approach assigns semantic roles to the extracted entities thereby making it possible to represent UTI as structured transactional information such that an array of data analytic tools can be applied to it for further analysis. In addition, a new framework for describing the properties of entities in the KB i.e. contextual and intrinsic properties was introduced. These properties provide new avenues for assessing and comparing KBs in different languages and domains.

The implementation method of the NER system depending on the amount of available resources was discussed. The purist implementation requires a large volume of annotated example documents as well as substantial high performance computing resources and can afford to search for only exact context matches and characteristics comparison. An economic implementation on the other hand can perform efficiently with considerably less amount of annotated examples and computing resources but uses approximate context matches and character comparison, hence more prone to errors in identifying correct entities.

Experiments using for four candidate scoring methods (Bayesian, dispersion, Dempster-Shafer belief theory and probabilistic classifier) were performed using the regular and disambiguated contexts. The results of the different scoring methods were analyzed and compared for different languages and domains. Generally, the precision of the system is relatively higher than its recall (0.9 and 0.62 respectively for highest F-measure performance). Unfortunately, the unavailability of publicly shared semantically annotated corpus makes the direct comparison of the results of the discussed experiments with those of other systems very difficult. These results are promising especially given that this is a NER approach at its infancy.

Visualization techniques for unstructured text when the proposed NER technique cannot be used due to entities not surrounded by enough contexts to enable an unambiguous candidate extraction or the sporadic use of unusual abbreviations and slangs was referenced.

In the concluding part of this dissertation, opportunities such as automatically creating RDF documents and semantics search engines presented by this NER approach were described. Also mentioned were the opportunities such as rapid system improvement due to contribution by several NER actors from different language and domain backgrounds. One limitations of this system is the manual annotation of example document.

Future work includes exploring more candidate scoring methods as well as simultaneously using multiple fragments in the entity location heuristics. The better handing of context would also be researched.

Appendix

Appendix A.

Year	HDD $/GB[1]	Internet IP Traffic (TG per month)[2]	Non-Internet IP Traffic (TB per month)[2]
1980	213000		
1981	318333.333		
1982	260000		
1983	195142.857		
1984	175684.211		
1985	71000		
1987	65000		
1988	29200		
1989	33666.6667		
1990	9000		
1991	7000		
1992	4000		
1993	2000		
1994	950		
1995	870.7		
1996	239.4		
1997	112.515		
1998	65.3682927		
1999	25.6515152		
2000	12.4454902		
2001	6.0725		
2002	2.96818182		
2003	1.71875		
2004	1.36090909		
2005	0.67	2342040	619814
2006	0.53	3199476	1032438
2007	0.42	4675176	1965863
2008	0.27	6734408	3619219
2009	0.07	9221807	5565951
2010		12440474	8160182
2011		17338055	11170878

Table A-1: Data for Figure 3.3: Global IP Traffic and Hard Drive Cost per Gigabyte since 1980.

[1] (Nova Scotia), [2] (Cisco Systems, 2007)

Year	Average Transistor Price (US$)[3]	Microprocessor Clock Speed (GHz)[3]
1968	1	
1969	0.85	
1970	0.6	
1971	0.3	
1972	0.15	
1973	0.1	
1974	0.07	
1975	0.028	
1976	0.015	1.35
1977	0.008	2.06
1978	0.005	2.14
1979	0.002	2.29
1980	0.0013	1.94
1981	0.00082	2.41
1982	0.0004	2.63
1983	0.00032	4.07
1984	0.00032	5.19
1985	0.00015	5.89
1986	0.00009	7.21
1987	0.000081	9.43
1988	0.00006	12.66
1989	0.000035	15.63
1990	0.00002	19.44
1991	0.000017	21.18
1992	0.00001	29.03
1993	0.000009	34.15
1994	0.000008	53.38
1995	0.000007	78.04
1996	0.000005	140.5
1997	0.000003	184.28
1998	0.0000014	337
1999	0.00000095	413.68
2000	0.0000008	413.68
2001	0.00000035	1684
2002	0.00000026	2317
2003		3088
2004		3990
2005		5173
2006		5631
2007		6739
2010		11511
2013		19348
2016		28751

Table A-2: Data for Figure 3.4: Cost of Microprocessor compared with its speed 1968-2010.

[3] (Kurzweil)

Appendix B.

Funeral for Levoyd J. Hughes, 78, of Dectaur will be Wednesday, Feb. 7, 2000, at 1 p.m. at Weck Funeral Home with the Rev. Randy Lee and the Rev. Kenny Smith officiating. Burial will be in Glowingtree Memorial Gardens. Visitation will be tonight from 6 to 9 at the funeral home. Mr. Hughes, who died Monday, Feb. 5, 2007, at his residence, was born Jan. 3, 1929, in Morgan County to Irving C. Hughes and Arrena M. Cobbs Hughes. He was a member of Seventh Street United Methodist Church and a member of Wheeler Basin Good Sam Camping Club. His brother, Jason Hughes, preceded him in death. Survivors include his wife, Sarah Hughes; three sons, Michael Levoyd Hughes and his wife, Pat, of Vicksburg, Miss., Williams Buel Hughes and his wife, Danette, and Paul Lamar Hughes and his wife, Becky, all of Decatur; two daughters, Beverly Astor and Karen Dilard and her husband, Sam, all of Dectaur; three brothers, Carrol Hughes of Center Springs, Hulond Hughes of Wedowee and Billy Hughes of Eva; two sisters, Sarah Eddleman of Fairview and Yvonne Hughes of Pinson; 10 grand-children; and four greatgrandchildren.

Figure B-1: Example of an English language obituary announcement

Asesinan a padre e hijo de 4 años en San Isidro 18 Agosto 2009 Actualizado: 08:54 PM hora de Cd. Juárez staff ?El Diario Un hombre y un niño fueron asesinados y una mujer resultó gravemente lesionada, tras un atentado perpetrado en la carretera Juárez-Porvenir la noche del lunes. La agresión ocurrió justo a 500 metros donde el pasado viernes otra familia, residentes de Las Cruces, Nuevo México, fue acribillada entre los límites de Loma Blanca y San Isidro y a un kilómetro del punto donde están instaladas siete cámaras de video de la Secretaría de Seguridad Pública Municipal. De acuerdo al protocolo de comunicación emitido por la Subprocuraduría de Justicia en la zona norte las víctimas fueron identificadas como Armando Pulido Mota, de 45 años, residente del poblado de San Agustín y el niño Iván Christopher Salgado, de 4 años. En el mismo evento una mujer, cuya identidad no fue revelada, se encuentra hospitalizada y su estado de salud fue reportado como grave. El documento oficial refiere que los hechos fueron reportados a la Policía Ministerial a las 23:15 horas en el kilómetro 24 de la carretera Juárez-Porvenir, a la altura del poblado de San Isidro. Inicialmente, el reporte mencionaba el hallazgo de un cuerpo sin vida, el cual correspondía a Pulido Mota, quien fue localizado en el interior de una camioneta Chevrolet Suburban, color blanco, modelo 1995, sobre el asiento del conductor.

Figure B-2: Example of a Spanish language crime report

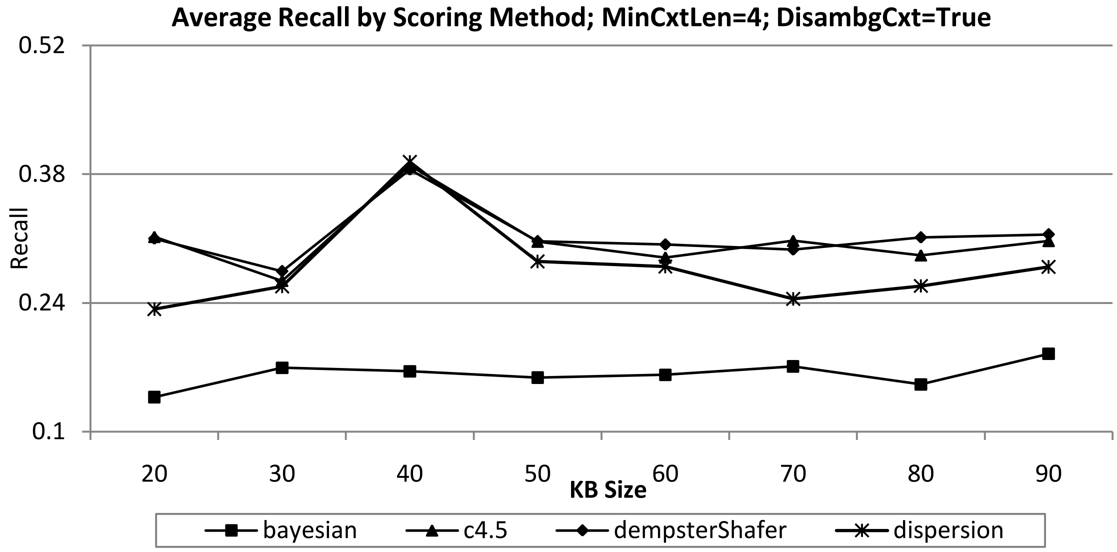


Figure B-3: Relationship between recall and KB size (English Obituary)

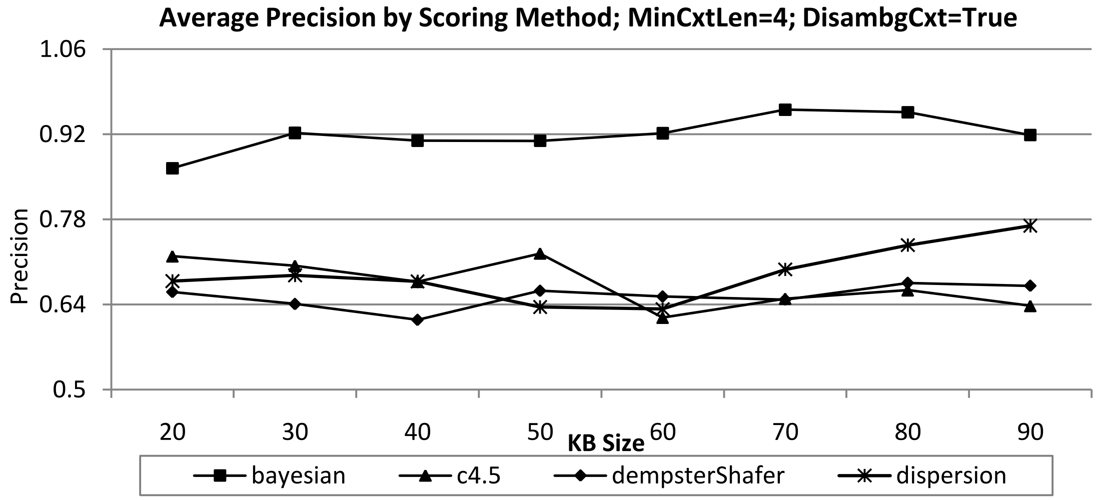


Figure B-4: Relationship between precision and KB size (English Obituary)

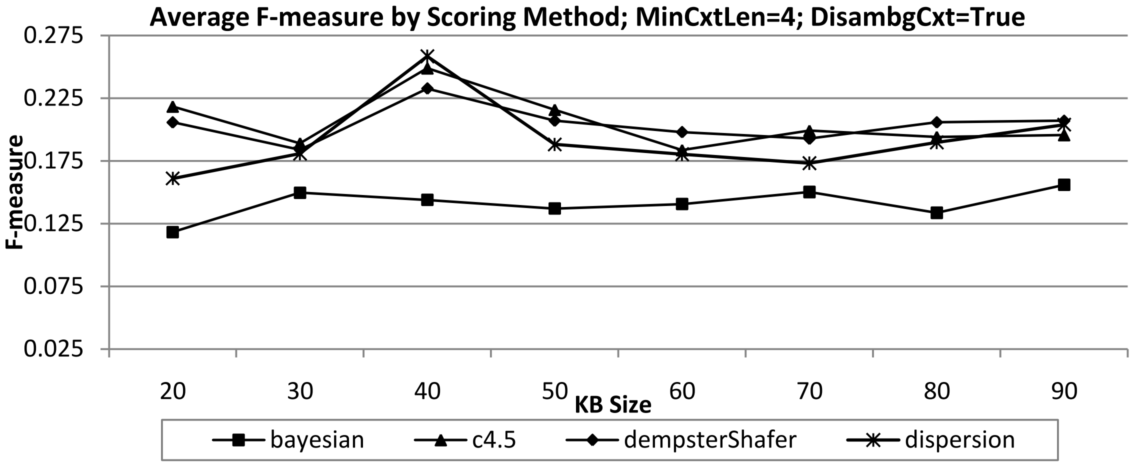


Figure B-5: Relationship between F-measure and KB size (English Obituary)

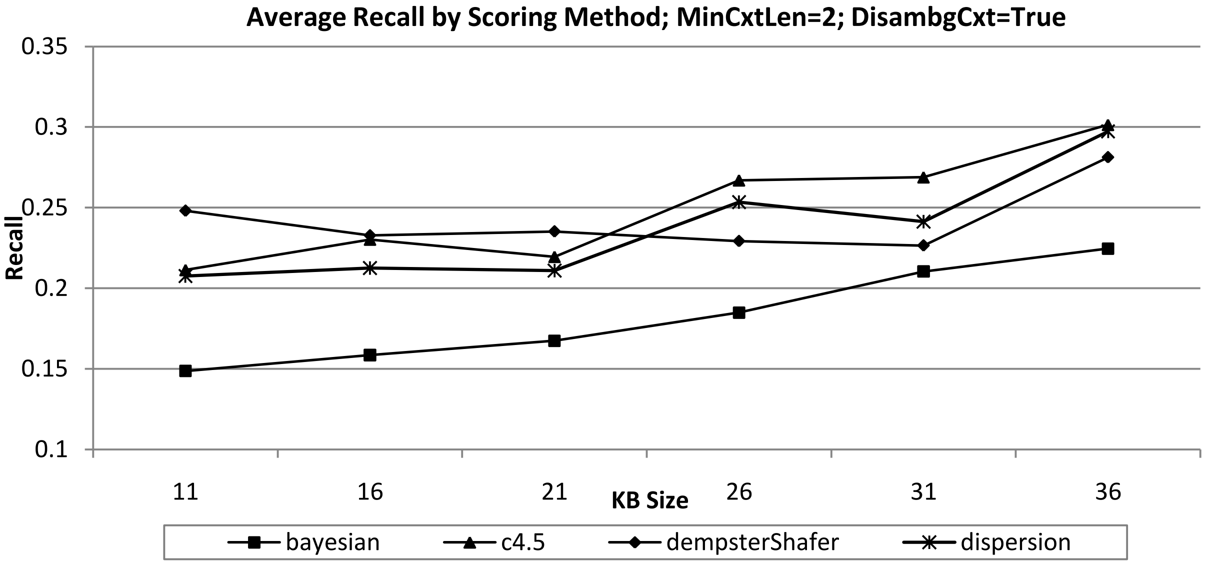


Figure B- 6: Relationship between recall and KB size (Spanish Crime Report)

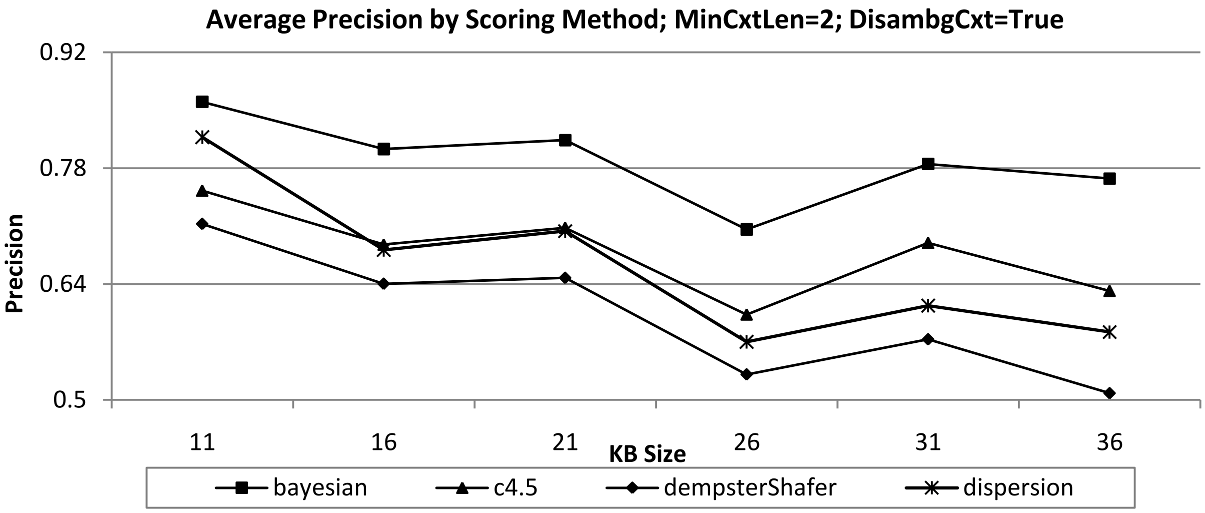


Figure B- 7: Relationship between precision and KB size (Spanish Crime Report)

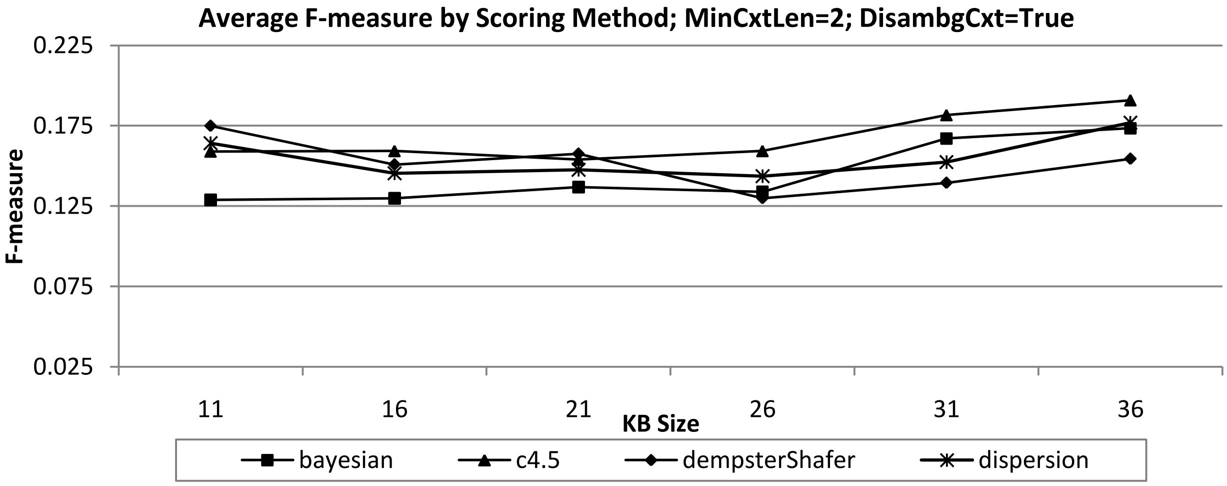
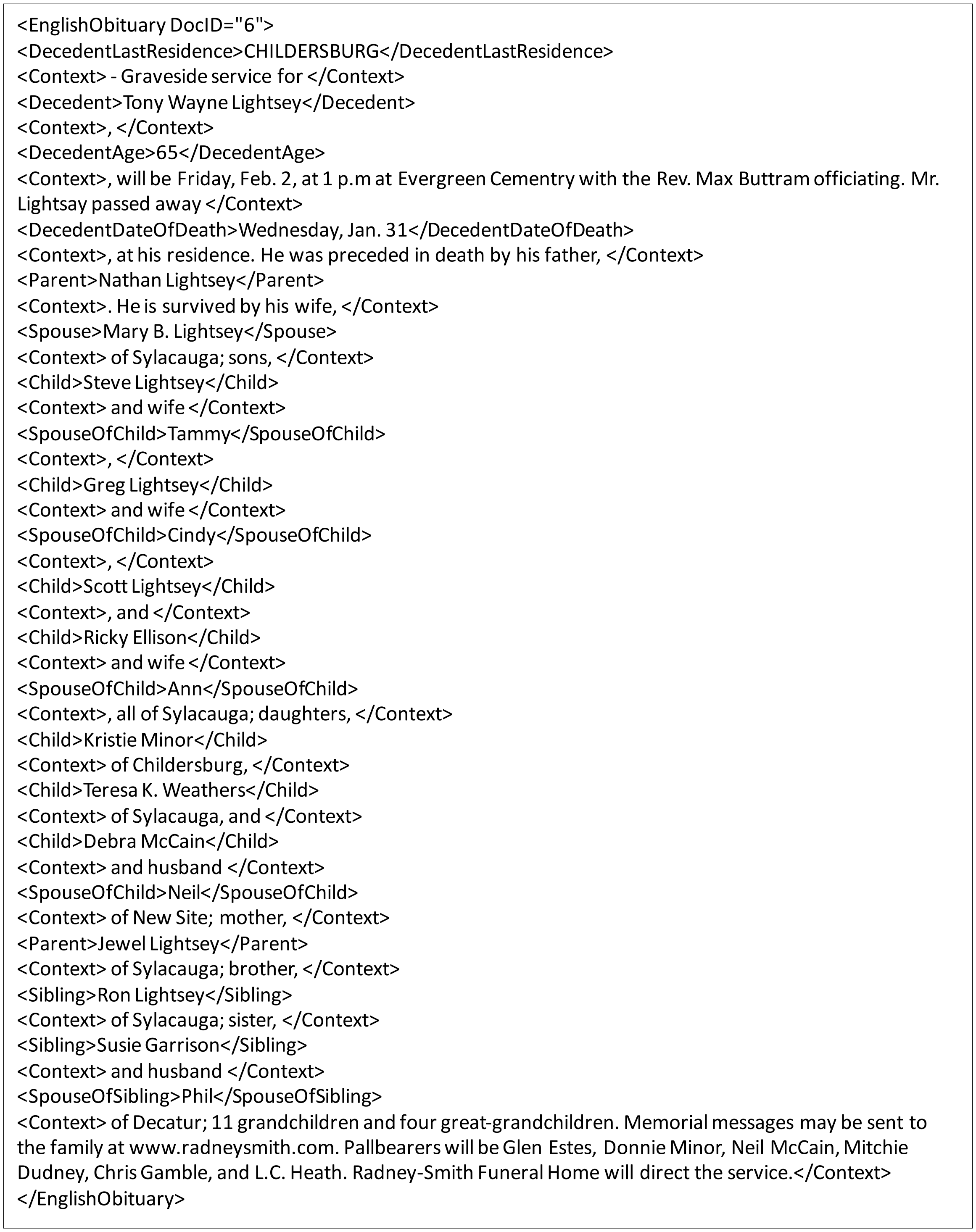
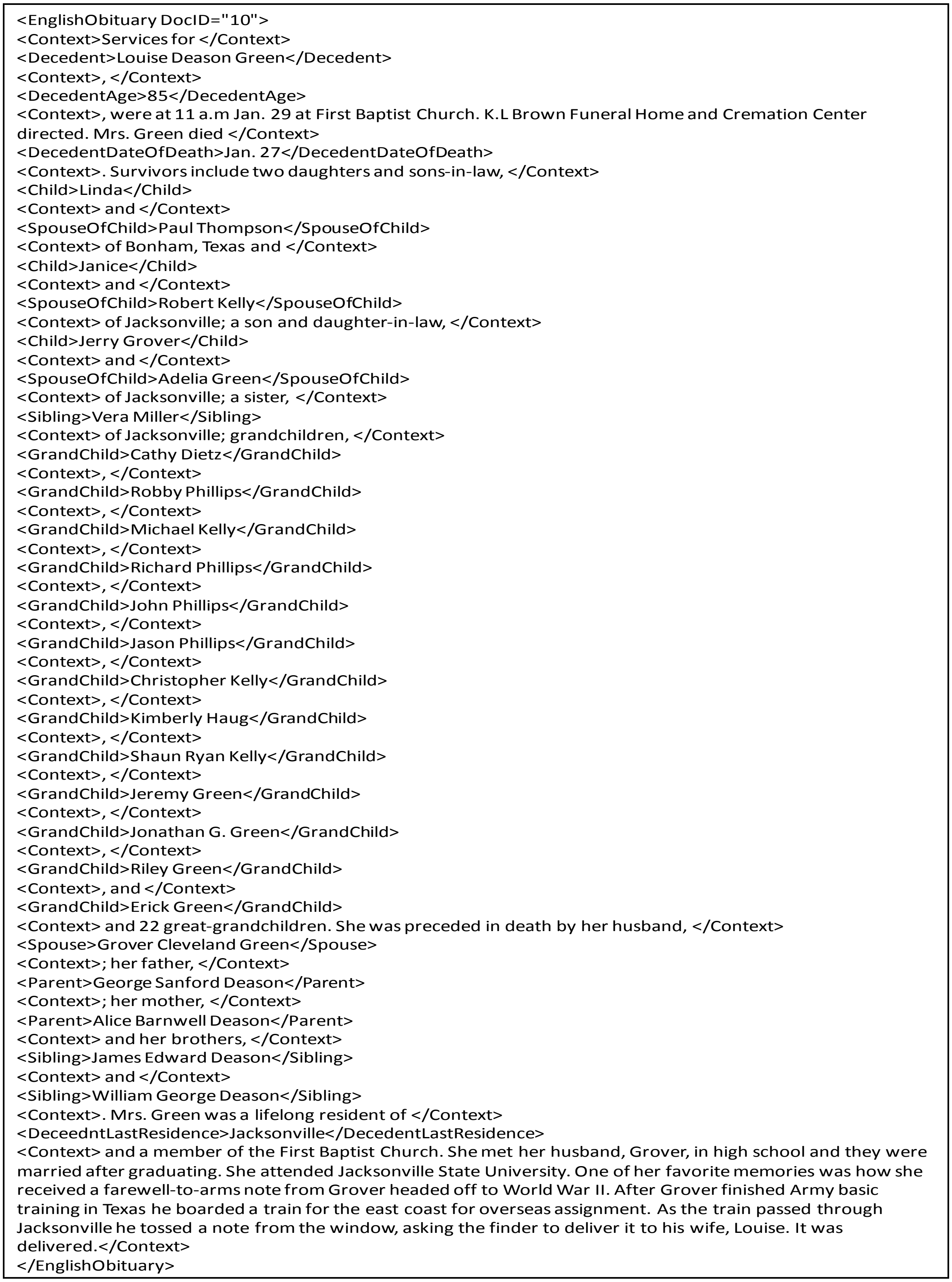


Figure B- 8: Relationship between F-measure and KB size (Spanish Crime Report)



```
<EnglishObituary DocID="6">
<DecedentLastResidence>CHILDERSBURG</DecedentLastResidence>
<Context> - Graveside service for </Context>
<Decedent>Tony Wayne Lightsey</Decedent>
<Context>, </Context>
<DecedentAge>65</DecedentAge>
<Context>, will be Friday, Feb. 2, at 1 p.m at Evergreen Cementry with the Rev. Max Buttram officiating. Mr. Lightsay passed away </Context>
<DecedentDateOfDeath>Wednesday, Jan. 31</DecedentDateOfDeath>
<Context>, at his residence. He was preceded in death by his father, </Context>
<Parent>Nathan Lightsey</Parent>
<Context>. He is survived by his wife, </Context>
<Spouse>Mary B. Lightsey</Spouse>
<Context> of Sylacauga; sons, </Context>
<Child>Steve Lightsey</Child>
<Context> and wife </Context>
<SpouseOfChild>Tammy</SpouseOfChild>
<Context>, </Context>
<Child>Greg Lightsey</Child>
<Context> and wife </Context>
<SpouseOfChild>Cindy</SpouseOfChild>
<Context>, </Context>
<Child>Scott Lightsey</Child>
<Context>, and </Context>
<Child>Ricky Ellison</Child>
<Context> and wife </Context>
<SpouseOfChild>Ann</SpouseOfChild>
<Context>, all of Sylacauga; daughters, </Context>
<Child>Kristie Minor</Child>
<Context> of Childersburg, </Context>
<Child>Teresa K. Weathers</Child>
<Context> of Sylacauga, and </Context>
<Child>Debra McCain</Child>
<Context> and husband </Context>
<SpouseOfChild>Neil</SpouseOfChild>
<Context> of New Site; mother, </Context>
<Parent>Jewel Lightsey</Parent>
<Context> of Sylacauga; brother, </Context>
<Sibling>Ron Lightsey</Sibling>
<Context> of Sylacauga; sister, </Context>
<Sibling>Susie Garrison</Sibling>
<Context> and husband </Context>
<SpouseOfSibling>Phil</SpouseOfSibling>
<Context> of Decatur; 11 grandchildren and four great-grandchildren. Memorial messages may be sent to the family at www.radneysmith.com. Pallbearers will be Glen Estes, Donnie Minor, Neil McCain, Mitchie Dudney, Chris Gamble, and L.C. Heath. Radney-Smith Funeral Home will direct the service.</Context>
</EnglishObituary>
```

Figure B-9: Annotated Obituary Document 6



```
<EnglishObituary DocID="10">
<Context>Services for </Context>
<Decedent>Louise Deason Green</Decedent>
<Context>, </Context>
<DecedentAge>85</DecedentAge>
<Context>, were at 11 a.m Jan. 29 at First Baptist Church. K.L Brown Funeral Home and Cremation Center
directed. Mrs. Green died </Context>
<DecedentDateOfDeath>Jan. 27</DecedentDateOfDeath>
<Context>. Survivors include two daughters and sons-in-law, </Context>
<Child>Linda</Child>
<Context> and </Context>
<SpouseOfChild>Paul Thompson</SpouseOfChild>
<Context> of Bonham, Texas and </Context>
<Child>Janice</Child>
<Context> and </Context>
<SpouseOfChild>Robert Kelly</SpouseOfChild>
<Context> of Jacksonville; a son and daughter-in-law, </Context>
<Child>Jerry Grover</Child>
<Context> and </Context>
<SpouseOfChild>Adelia Green</SpouseOfChild>
<Context> of Jacksonville; a sister, </Context>
<Sibling>Vera Miller</Sibling>
<Context> of Jacksonville; grandchildren, </Context>
<GrandChild>Cathy Dietz</GrandChild>
<Context>, </Context>
<GrandChild>Robby Phillips</GrandChild>
<Context>, </Context>
<GrandChild>Michael Kelly</GrandChild>
<Context>, </Context>
<GrandChild>Richard Phillips</GrandChild>
<Context>, </Context>
<GrandChild>John Phillips</GrandChild>
<Context>, </Context>
<GrandChild>Jason Phillips</GrandChild>
<Context>, </Context>
<GrandChild>Christopher Kelly</GrandChild>
<Context>, </Context>
<GrandChild>Kimberly Haug</GrandChild>
<Context>, </Context>
<GrandChild>Shaun Ryan Kelly</GrandChild>
<Context>, </Context>
<GrandChild>Jeremy Green</GrandChild>
<Context>, </Context>
<GrandChild>Jonathan G. Green</GrandChild>
<Context>, </Context>
<GrandChild>Riley Green</GrandChild>
<Context>, and </Context>
<GrandChild>Erick Green</GrandChild>
<Context> and 22 great-grandchildren. She was preceded in death by her husband, </Context>
<Spouse>Grover Cleveland Green</Spouse>
<Context>; her father, </Context>
<Parent>George Sanford Deason</Parent>
<Context>; her mother, </Context>
<Parent>Alice Barnwell Deason</Parent>
<Context> and her brothers, </Context>
<Sibling>James Edward Deason</Sibling>
<Context> and </Context>
<Sibling>William George Deason</Sibling>
<Context>. Mrs. Green was a lifelong resident of </Context>
<DeceedntLastResidence>Jacksonville</DecedentLastResidence>
<Context> and a member of the First Baptist Church. She met her husband, Grover, in high school and they were
married after graduating. She attended Jacksonville State University. One of her favorite memories was how she
received a farewell-to-arms note from Grover headed off to World War II. After Grover finished Army basic
training in Texas he boarded a train for the east coast for overseas assignment. As the train passed through
Jacksonville he tossed a note from the window, asking the finder to deliver it to his wife, Louise. It was
delivered.</Context>
</EnglishObituary>
```

Figure B-10: Annotated Obituary Document 10

Bibliography

Abney Steven P. Rapid Incremental parsing with repair [Conference] // Proceedings of the 6th. New OED Conference: Electronic Text Research. - 1990. - pp. 1-9.

Abney Steven Partial parsing via finite-state cascades [Journal] // Natural Language Engineering. - December 1996. - 4 : Vol. 2. - pp. 337–344.

Abney Steven The SCOL Manual - Version 0.1b [Report]. - Tuebingen : [s.n.], 1997.

Ahern S. [et al.] World Explorer: Visualizing Aggregate Data from Unstructured Text in Geo-Referenced Collections. [Conference] // Proceedings of the Seventh ACM/IEEE-CS Joint Conference on Digital Libraries. - Vancouver : ACM, 2007. - pp. 1 - 10.

Amazon.com Amazon Mechanical Turk [Online] // Amazon.com. - Amazon.com. - October 14, 2010. - https://www.mturk.com/mturk/welcome.

Anderson Chris The End of Theory: The Data Deluge Makes the Scientific Method Obsolete [Online] // Wired Magazine. - June 23, 2008. - April 03, 2010. - http://www.wired.com/science/discoveries/magazine/16-07/pb_theory.

Anderson Tom H. C. Reward for Being the Top Market Researcher on Twitter [Online] // Tom H. C. Anderson - Next Gen Market Research. - Next Gen Market Researcher, February 3, 2010 . - February 4, 2010. - http://www.tomhcanderson.com/2010/02/03/reward-of-being-the-top-market-researcher-on-twitter/.

Appelt Douglas E. [et al.] FASTUS: A finite-state processor for information extraction from real-world text [Conference] // Proceedings of the 13th International Joint Conference on Artificial Intelligence. - Chambery, France : [s.n.], 1993. - pp. 1172-1178.

Baeza-Yates Ricardo and Navarro Gonzalo XQL and proximal nodes [Journal] // Journal of the American Society for Information Science and Technology. - [s.l.] : John Wiley & Sons, Inc. , May 2002. - 6 : Vol. 53. - pp. 504-514.

Bahl L.R. and Mercer R.L. Part of speech assignment by a statistical decision algorithm [Conference] // nternational Symposium on Information Theory. - Ronneby, Sweden : [s.n.], 1976.

Bangalore Srinivas Complexity of Lexical Descriptions and its Relevance to Partial Parsing // Institute for Research in Cognitive Science. - [s.l.] : University of Pennsylvania Philadelphia, PA, USA, June 1997.

Banko Michele and Brill Eric Scaling to Very Very Large Corpora for Natural Language Disambiguation [Conference] // Proceedings of the 39th Annual Meeting on Association for Computational Linguistics. - Toulouse, France : Association for Computational Linguistics Morristown, NJ, USA, 2001. - pp. 26-33.

Beale Andrew David Lexicon and grammar in probabilistic tagging of written English [Conference] // 26th annual meeting on Association for Computational Linguistics. - Buffalo, New York : Association for Computational Linguistics Morristown, NJ, USA , 1988. - pp. 211-216.

Beeferman Doug Lexical Discovery with an Enriched Semantic Network [Conference] // In Proceedings of the ACL/COLING Workshop on Applications of WordNet in Natural Language Processing Systems. - 1998. - pp. 358-364.

Bertino Elisa. and Ferrari Elena XML and data integration [Journal] // IEEE Internet Computing. - [s.l.] : IEEE, Nov-Dec 2001. - 6 : Vol. 5. - pp. 75-76.

Bharati Akshar [et al.] Two stage constraint based hybrid approach to free word order language dependency parsing [Conference] // Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09). - Paris, France : Association for Computational Linguistics, 2009. - pp. 77-80.

Bikel Daniel M. [et al.] Nymble: a High-Performance Learning Name-finder [Conference] // In Proceedings of the Fifth Conference on Applied Natural Language Processing. - 1997. - pp. 194-201.

Bitton Dina One platform for mining structured and unstructured data dream or reality? [Journal] // ACM. - [s.l.] : VLDB Endowment, September 2006. - pp. 1261-1262.

Black Ezra [et al.] Towards History-based Grammars Using Richer Models for Probabilistic Parsing [Conference] // Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics. - Columbus, Ohio : Association for Computational Linguistics Morristown, NJ, USA, 1993. - pp. 31-37.

Black William J, Rinaldi Fabio and Mowatt David Facile: Description Of The NE System Used For MUC-7 [Conference] // 7th Message Understanding Conference. - 1998.

Black William J. and Vasilakopoulos Argyrios proceedings of the 6th conference on Natural language learning [Conference] // Proceedings of the 6th conference on Natural language learning. - [s.l.] : Association for Computational Linguistics Morristown, NJ, USA, 2002. - Vol. 20. - pp. 1-4.

Bontcheva Kalina [et al.] Shallow Methods for Named Entity Coreference Resolution [Conference] // Traitement Automatique des Langues Naturelles (TALN). - Nancy, France : [s.n.], 2002.

Borthwick Andrew [et al.] NYU: Description of the MENE Named Entity System as Used in MUC-7 [Conference] // In Proceedings of the Seventh Message Understanding Conference (MUC-7). - 1998.

Boulton Clint Twitter Use Tapering Off in the U.S., but Rising Overseas - Web Services, Web 2.0 & SOA [Online] // eWeek.com. - January 15, 2010. - February 28, 2010. - http://www.eweek.com/c/a/Web-Services-Web-20-and-SOA/Twitter-Use-Declining-in-the-US-But-Rising-Overseas-592323/.

Box George E. P. and Draper Norman R. Empirical Model-Building and Response Surfaces [Book]. - [s.l.] : Wiley, 1987. - 0471810339.

Brants Thorsten [et al.] Large Language Models in Machine Translation [Conference] // Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). - Prague, Czech Republic : Association for Computational Linguistics Morristown, NJ, USA, 2007. - pp. 858-867.

Brill Eric Transformation-based error-driven learning and natural language processing: a case study in part of speech tagging [Journal] // Computational Linguistics. - 1995. - Vol. 21. - pp. 543-565.

Briscoe Ted and Carroll John Robust accurate statistical annotation of general text [Conference] // Proceedings of the Third International Conference on Language Resources and Evaluation. - Las Palmas de Gran Canaria : [s.n.], 2002. - pp. 1499-1504.

Brown Peter F. [et al.] Class-Based N-Gram Models of Natural Language [Journal] // Computational Linguistics. - December 17, 1990. - 4 : Vol. 18. - pp. 467-479.

Che Wanxiang [et al.] Multilingual Dependency-based Syntactic and Semantic Parsing [Conference] // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. - Boulder, Colorado : Association for Computational Linguistics Morristown, NJ, USA, 2009. - pp. 49-54.

Chiang Chia-Chu [et al.] A case study in partial parsing unstructured text [Conference] // Fifth International Conference on Information Technology: New Generations (itng 2008),. - Vegas, NV : IEEE Press, 2008. - pp. 447-452.

Chieu Hai Leong and Ng Hwee Tou Name Entity Recognition: a maximum entropy approach using global information [Conference] // In Proceedings of COLING-02. - 2002. - pp. 190-196.

Chu-Carroll Jennifer and Prager John An experimental study of the impact of information extraction accuracy on semantic search performance [Conference] // Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. - Lisbon, Portugal : [s.n.], 2007. - pp. 505-514.

Church Kenneth W. A stochastic parts program and noun phrase parser for unrestricted text [Conference] // International Conference on Acoustics, Speech, and Signal Processing. - Glasgow, Scotland : [s.n.], 1989. - Vol. 2. - pp. 695-698.

Church Kenneth W. On Memory Limitations in Natural Language Processing [Report]. - [s.l.] : Massachusetts Institute of Technology Cambridge, MA, USA, 1980.

Cisco Systems Global IP Traffic Forecast and Methodology // Global IP Traffic Forecast and Methodology, 2006–2011. - 2007.

Cohen Sara [et al.] XSEarch: A semantic search engine for XML [Conference] // Proceedings of the 29th international conference on Very large data bases. - Berlin, Germany : VLDB Endowment, 2003. - Vol. 29. - pp. 45-56.

Cunningham Hamish [et al.] GATE: an Architecture for Development of Robust HLT Applications [Conference] // Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. - Philadelphia, Pennsylvania : Association for Computational Linguistics Morristown, NJ, USA, 2002. - pp. 168 - 175.

Curran James R. and Clark Stephen Language independent NER using a maximum entropy tagger [Conference] // Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003. - Edmonton, Canada : Association for Computational Linguistics Morristown, NJ, USA, 2003. - Vol. 4. - pp. 164-167.

Daelemans Walter, Bosch Antal Van Den and Weijters Ton IGTree: Using Trees for Compression and Classification in Lazy Learning Algorithms [Journal] // Artificial Intelligence Review. - [s.l.] : Springer Netherlands, February 1997. - 1-5 : Vol. 11. - pp. 407-723.

De Meulder Fien and Daelemans Walter Memory-based named entity recognition using unannotated data [Conference] // Proceedings of the seventh conference on Natural language learning. - Edmonton, Canada : Association for Computational Linguistics, Morristown, NJ, USA, 2003. - pp. 208-211.

Doan AnHai [et al.] Information extraction challenges in managing unstructured data [Journal] // ACM. - [s.l.] : VLDB Endowment , September 2007. - pp. 14-20.

Dong X. L. and Naumann F. Data Fusion - Resolving Data Conflicts for Integration. [Conference] // Proceedings of the VLDB Endowment. - Lyon : ACM, 2009. - pp. 1654 - 1655.

Economist, The Data, data everywhere - A special report on managing information [Report]. - [s.l.] : Compulink Management Center, Inc., 27th Feb., 2010.

Efron M. Generative Model-Based MetaSearch for Data Fusion in Information Retrieval. [Conference] // Proceedings of the Ninth ACM/IEEE-CS Joint Conference on Digital Libraries. - Austin : ACM, 2009. - pp. 153 - 162.

Elson J., Howell J. and Douceur J. R. MapCruncher: Integrating the World's Geographic Information. [Journal] // SIGOPS Operating Systems Review. - 2007. - pp. 50 - 59.

Emblen Donald Lewis Mark Roget: The Word and the Man [Book]. - [s.l.] : Longman Group, London, UK, 1970.

Endeca Technologies, Inc Endeca Unveils McKinley Release of the Information Access Platform, Allowing for Faster and Eas [Online] // Enterprise Search, Information Access, and Guided Navigation. - Endeca, March 23, 2009. - October 1, 2009. - http://www.endeca.com/12249d97-8db7-4b4c-9503-a74aa5290676/news-and-events-press-releases-2009.htm.

Fellbaum Christiane WordNet An Electronic Lexical Database [Book]. - Cambridge, MA; London : The MIT Press, 1998. - 978-0-262-06197-1.

Ferrucci David and Lally Adam Accelerating corporate research in the development, application and deployment of human language technologies [Conference] // Proceedings of the HLT-NAACL 2003 workshop on Software engineering and architecture of language technology systems. - [s.l.] : Association for Computational Linguistics Morristown, NJ, USA , 2003. - Vol. 8. - pp. 67-74.

Ferrucci David and Lally Adam Building an example application with the unstructured information management architecture [Journal] // IBM Systems Journal. - [s.l.] : IBM Corp. Riverton, NJ, USA , July 2004. - 3 : Vol. 43 . - pp. 455-475.

Fisher Craig [et al.] Introduction to Information Quality [Book]. - Boston : MIT Publications, 2008. - 4.

Fisher David [et al.] Description of the UMass system as used for MUC-6 [Conference] // Proceedings of the 6th conference on Message understanding. - Columbia, Maryland : Association for Computational Linguistics Morristown, NJ, USA, 1995. - pp. 127-140.

Fisher David [et al.] Description of the UMass system as used for MUC-6 [Conference] // Proceedings of the 6th conference on Message understanding. - Columbia, Maryland : Association for Computational Linguistics Morristown, NJ, USA, 1995. - pp. 127-140.

Frank J. About Us: MetaCarta Inc. [Online] // MetaCarta Inc. Web site. - January 1, 2001. - 02 01, 2010. - http://metacarta.com/.

Friedman Nir [et al.] Learning Probabilistic Relational Models [Conference] // Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI). - Stockholm, Sweden : [s.n.], 1999. - pp. 1300-1307.

Fuhr Norbert and Großjohann Kai XIRQL: A Query Language for Information Retrieval in XML Documents [Conference] // Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. - [s.l.] : ACM, 2001. - pp. 172-180.

GATE project team GATE.ac.uk - gate/doc/plugins.html [Online] // GATE.ac.uk. - October 18, 2010. - http://gate.ac.uk/gate/doc/plugins.html.

Goth Greg News: A Structure for Unstructured Data Search [Journal] // IEEE Distributed Systems Online. - January 2007. - 1 : Vol. 8. - pp. 1-4.

Grishman Ralph and Sundheim Beth Message Understanding Conference - 6: A Brief History [Conference] // Proceedings of the 16th conference on Computational linguistics. - Copenhagen, Denmark : Association for Computational Linguistics Morristown, NJ, USA, 1996. - Vol. 1. - pp. 466-471.

Grishman Ralph Information Extraction: Techniques and Challenges [Book Section] // Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology, International Summer School / book auth. SCIE / ed. Pazienza Maria Teresa. - Frascati/Rome : Springer, 1997.

Gruber Thomas R. A translation approach to portable ontology specifications [Journal] // Knowledge Acquisition. - [s.l.] : Academic Press Ltd. London, UK, June 1993. - 2 : Vol. 5. - pp. 199--220.

Hajic Jan [et al.] The CoNLL-2009 shared task: syntactic and semantic dependencies in multiple languages [Conference] // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. - Boulder, Colorado : Association for Computational Linguistics Morristown, NJ, USA, 2009. - pp. 1-18.

Harris Zellig Harris. String Analysis of Language Structure [Journal]. - [s.l.] : Mouton and Co., The. Hague, Netherlands, 1962.

Harris Zellig String Analysis of Language Structure [Journal]. - [s.l.] : Mouton and Co., The. Hague, Netherlands, 1962.

Hashemi Ray [et al.] Building Semantic-Rich Patterns for Extracting Features from Online Announcements [Conference] // International Association for Development of Information Society (IADIS) International Conference on WWW/Internet. - Algarve, Portugal : [s.n.], 2003.

Hashemi Ray R. [et al.] Extraction of Features with Unstructured Representation [Conference] // Proceedings of the IADIS International Conference on WWW/ Internet. - Lisbon, Portugal : International Association for Development of Information Society, 2002. - pp. 47-53.

Havenstein Heather LA Fire Department all 'aTwitter' over Web 2.0 [Online] // PCWorld. - August 3, 2007. - February 15, 2010. - http://www.pcworld.com/article/135518/la_fire_department_all_atwitter_over_web_20.html.

Hendrickx Iris and van den Bosch Antal Memory-based one-step named-entity recognition: effects of seed list features, classifier stacking, and unannotated data [Conference] // Proceedings of the seventh conference on Natural language learning. - Edmonton, Canada : Association for Computational Linguistics, Morristown, NJ, USA. - pp. 176-179.

Hobbs Jerry R. [et al.] Background The TACITUS System: The MUC-3 Experience [Report]. - Menlo Park, CA : Artificial Intelligence Center SRI International, 1991.

Hobbs Jerry R. [et al.] FASTUS: A System for Extracting Information from Text [Conference] // Proceedings of the workshop on Human Language Technology. - Princeton, New Jersey : Association for Computational Linguistics Morristown, NJ, USA, 1993. - pp. 133-137.

Holt Alex Monitter [Online] // Soyrex . - March 15, 2010. - http://monitter.com/.

Hong Gumwon [et al.] A Hybrid Approach to English-Korean Name Transliteration [Conference] // Proceedings of the 2009 Named Entities Workshop: Shared Task on

Transliteration (NEWS 2009)},. - Suntec, Singapore : Association for Computational Linguistics, 2009. - pp. 108-111.

Hsu Tzu-Wei [et al.] MonkEllipse: Visualizing the History of Information Visualization [Conference] // IEEE Symposium on Information Visualization. - [s.l.] : IEEE, 2004. - pp. r9-r9. - 10.1109/INFVIS.2004.48.

Inmon William H. and Nesavich Anthony Taping in Unstructured Data [Book]. - [s.l.] : Prentice Hall, 2008. - p. 241. - 978-0-13-236029-6.

Intel Corporation Intel Microprocessor Quick Reference Guide [Online] // Intel Corp.. - October 21, 2010. - http://www.intel.com/pressroom/kits/quickrefyr.htm.

Jelinek Fred Self-organized language modeling for speech recognition [Book Section] // Readings in Speech Recognition. - San Francisco, CA, USA : Morgan Kaufmann Publishers Inc. , 1990. - 1-55860-124-4.

Jelinek Frederick Self-organized language modeling for speech recognition [Book Section] // Readings in Speech Recognition. - San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1990. - 1-55860-124-4.

Jiang Wei, Guan Yi and Wang Xiao-Long Improving Feature Extraction in Named Entity Recognition Based on Maximum Entropy Model [Conference] // International Conference on Machine Learning and Cybernetics. - Dalian, China : IEEE Publications, 2006. - pp. 2630 - 2635.

Jing Wang, Dequan Zheng and Tiejun Zhao Research on Improved TBL Based Japanese NER Post-Processing [Conference] // Advanced Language Processing and Web Information Technology. - Dalian Liaoning, China : IEEE Publications, 2008. - pp. 145-149.

Joachims Thorsten Text categorization with Support Vector Machines: Learning with many relevant features [Conference] // Proceedings of ECML-98, 10th European Conference on Machine Learning. - [s.l.] : Springer, 1997. - pp. 137-142.

Joshi Aravind K. and Hopely Phil A parser from antiquity [Journal] // Natural Language Engineering. - [s.l.] : Cambridge University Press New York, NY, USA, 1996. - 4 : Vol. 2. - pp. 291-294.

Kiesler Max Twitter StreamGraph [Online] // Neoformix . - March 15, 2010. - http://www.maxkiesler.com/2009/07/20/twitter-streamgraph-visualization/.

Kimoto Haruo and Iwadera Toshiaki Construction of a dynamic Thesaurus and its use for associated information retrieval [Conference] // Proceedings of the 13th annual international conference on Research and development in information retrieval. - Brussels, Belgium : ACM New York, NY, USA, 1989. - pp. 227-240.

Klein Sheldon and Simmons Robert F. A Computational Approach to Grammatical Coding of English Words [Journal] // Journal of the ACM. - [s.l.] : ACM New York, NY, USA, July 1963. - 3 : Vol. 10. - pp. 334-347.

Kuflik Tsvi, Boger Zvi and Shoval Peretz Filtering search results using an optimal set of terms identified by an artificial neural network [Journal] // Information Processing and Management: an International Journal. - [s.l.] : Pergamon Press, Inc. Tarrytown, NY, USA, March 2006. - 2 : Vol. 42. - pp. 469-483.

Kurzweil Ray Singularity is Near - SIN Graph [Online] // The Singularity is Near: When Humans Transcend Biology. - Viking press. - October 21, 2010. - http://www.singularity.com/charts/. - 0670033847.

Levas Anthony [et al.] The Semantic Analysis Workbench (SAW): Towards a Framework for Knowledge Gathering and Synthesis [Conference] // Proceedings of the International Conference on Intelligence Analysis. - McClean, VA : IBM Research Report, 2005.

Levenshtein Vladimir Binary Codes Capable of Correcting Deletions, Insertions and Reversals [Journal] // Soviet Physics Doklady. - 1966. - Vol. 10. - pp. 707-710.

Li Yaoyong, Bontcheva Kalina and Cunningham Hamish Adapting svm for data sparseness and imbalance: A case study in information extraction [Journal] // Natural Language Engineering . - [s.l.] : Cambridge University Press New York, NY, USA, April 2009. - 2 : Vol. 15. - pp. 241-271.

Lima Manuel Social Networks - TweetWheel [Online] // Visual Complexity. - March 15, 2010. - http://www.visualcomplexity.com/vc/project.cfm?id=587.

Lin Dekang Dependency-based evaluation of MINIPAR [Conference] // Proceedings of the Workshop on the Evaluation of Parsing Systems. - Granada, Spain : [s.n.], 1998. - pp. 28-30.

Mansouri Alireza, Affendey Lilly Suriani and Mamat Ali Named Entity Recognition Approaches [Journal]. - 2008. - 2 : Vol. 8.

Marcken Carl G. de Parsing the LOB corpus [Conference] // 28th annual meeting on Association for Computational Linguistics. - Pittsburgh, Pennsylvania : Association for Computational Linguistics Morristown, NJ, USA, 1990. - pp. 243-251.

Maskey Sameer R. [et al.] Class-based named entity translation in a speech to speech translation system [Conference] // Spoken Language Technology Workshop, 2008. - Goa : IEEE, 2008. - pp. 253-256. - 978-1-4244-3471-8.

Mayfield James, McNamee Paul and Piatko Christine Named entity recognition using hundreds of thousands of features [Conference] // Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003. - Edmonton, Canada : Association for Computational Linguistics , 2003. - Vol. 4. - pp. 184-187.

McCallum Andrew Information Extraction: Distilling Structured Data from Unstructured Text [Journal] // ACM Queue. - [s.l.] : ACM, September 2005. - pp. 49-57.

Mccallum Andrew, Freitag Dayne and Pereira Fernando Maximum Entropy Markov Models for Information Extraction and Segmentation [Conference] // Proceedings of the Seventeenth International Conference on Machine Learning. - [s.l.] : Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2000. - pp. 591-598. - ISBN:1-55860-707-2 .

McCallum Andrew, Freitag Dayne and Pereira Fernando Maximum Entropy Markov Models for Information Extraction and Segmentation [Conference] // Proceedings of the Seventeenth International Conference on Machine Learning. - [s.l.] : Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2000. - pp. 591-598. - ISBN:1-55860-707-2 .

McDonald Scott Target word selection as proximity in semantic space [Conference] // Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics . - Montreal, Quebec, Canada : Association for Computational Linguistics Morristown, NJ, USA , 1998. - Vol. 2. - pp. 1496-1498.

Merialdo Bernard Tagging English text with a probabilistic model [Journal] // Computational Linguistics. - [s.l.] : MIT Press Cambridge, MA, USA, June 1994. - 2 : Vol. 20. - pp. 155-171.

Merrill Lynch & Co. Enterprise Information Portals // Server and Enterprise Software - indepth Report / ed. Shilakes Christopher C. and Tylman Julie. - [s.l.] : Merrill Lynch & Co., November 16, 1998.

Mihalcea Rada and Moldovan Dan I. Document Indexing using Named Entities [Book Section] // Studies in Informatics and Control. - 2001.

Mikheev Andrei, Grover Claire and Moens Marc DESCRIPTION OF THE LTG SYSTEM USED FOR MUC-7 [Conference] // Proceedings of the 7<sup>th</sup> Message Understanding Conference (MUC-7). - Fairfax, Virginia : [s.n.], 1998.

Mikheev Andrei, Moens Marc and Grover Claire Named Entity Recognition without Gazetteers [Conference] // Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics. - Bergen, Norway : Association for Computational Linguistics Morristown, NJ, USA , 1999. - pp. 1-8.

Miller Scott [et al.] BBN: Description Of The SIFT System As Used For MUC-7 [Conference] // 7th Message Understanding Conference. - 1998.

Moore Robert J. New Data on Twitter’s Users and Engagement [Online] // The Metric System. - January 26, 2010. - February 26, 2010. - http://themetricsystem.rjmetrics.com/2010/01/26/new-data-on-twitters-users-and-engagement/.

Morozov Evgeny Iran Elections: A Twitter Revolution? [Online] // The Washington Post. - June 17, 2009. - February 9, 2010. - http://www.washingtonpost.com/wp-dyn/content/discussion/2009/06/17/DI2009061702232.html.

Narayanan Srinivas and Harabagiu Sanda M. Answering Questions Using Advanced Semantics And Probabilistic Inference [Conference] // In Proceedings of the Workshop on pragmatics of question-answering. - 2004. - pp. 10-16.

NASA World Wind [Online] // World Wind. - February 9, 2010. - http://ti.arc.nasa.gov/projects/worldwind/index.php.

Nova Scotia Cost of Hard Drive Storage Space [Online] // Nova Scotia's Electric Gleaner. - October 21, 2010. - http://ns1758.ca/winch/winchest.html.

Ohbuchi Ryutarou and Kobayashi Jun Unsupervised learning from a corpus for shape-based 3D model retrieval [Conference] // Proceedings of the 8th ACM international workshop on Multimedia information retrieval. - Santa Barbara, California, USA : ACM New York, NY, USA , 2006. - pp. 163-172.

Osesina O. Isaac and Talburt John R. Towards a Data-Intensive Approach to Named Entity Recognition [Conference] // International Conference on Information Quality. - Little Rock, Arkansas : [s.n.], 2010.

Osesina O. Isaac, Bartley Cecilia and Tudoreanu M. Eduard Improving information quality of textual data by geographical reference [Conference] // 2010 ALAR Conference on Applied Research in Information Technology. - Conway, Arkansas : [s.n.], 2010.

Osesina O. Isaac, Bartley Cecilia and Tudoreanu M. Eduard Mapping realities: The co-visualization of geographic and non-spatial textual information [Conference] // 2010 International Conference on Modeling, Simulation, and Visualization Methods. - Las Vegas, Nevada : [s.n.], 2010. - pp. 10-16.

Ostrow Adam Twitter’s Massive 2008: 752 Percent Growth [Online] // Mashable The Social Media Guide. - February 9, 2010. - http://mashable.com/2009/01/09/twitter-growth-2008/.

Paliouras Georgios [et al.] Learning Decision Trees for Named-Entity Recognition and Classification [Conference] // ECAI Workshop on Machine Learning for Information Extraction. - 2000.

Paliouras Georgios [et al.] Learning Decision Trees for Named-Entity Recognition and Classification [Conference] // ECAI Workshop on Machine Learning for Information Extraction. - 2000.

Quinlan Ross J. Learning, C4.5: Programs for Machine [Book]. - [s.l.] : Morgan Kaufmann, 1992.

Raghuveer Aravindan [et al.] Towards efficient search on unstructured data an intelligent-storage approach [Journal] // ACM. - [s.l.] : ACM, September 2007. - pp. 951-954.

Salton Gerard and McGill Michael J. Introduction to Modern Information Retrieval [Book]. - New York : McGraw-Hill Companies, 1983. - p. 448.

Sanderson Mark and Croft Bruce Deriving concept hierarchies from text [Conference] // Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval . - Berkeley, California, United States : ACM New York, NY, USA , 1999. - pp. 206-213.

Schonfeld Erick Sneak Peek At T2, Twine’s Semantic Search Engine [Online] // TechCrunch. - September 18, 2009. - October 1, 2009. - http://www.techcrunch.com/2009/09/18/sneak-peak-at-t2-twines-semantic-search-engine/.

Schütze Hinrich Dimensions of meaning [Conference] // IEEE Supercomputing Proceedings. - Minneapolis, MN : [s.n.], 1992. - pp. 787-796.

Shafer Glenn A Mathematical Theory of Evidence [Book]. - [s.l.] : Princeton University Press, 1976.

Shannon Claude E. A Mathematical Theory of Communication. - [s.l.] : CSLI Publications, 1948.

Shilakes Christopher C. and Tylman Julie Enterprise Information Portals // Server and Enterprise Software - indepth Report. - [s.l.] : Merrill Lynch & Co., November 16, 1998.

Shou X. M. and Sanderson M. Experiments on Data Fusion Using Headline Information. [Conference] // Proceedings of the Twenty-fifth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. - Tampere : ACM, 2002. - pp. 413 - 414.

Srihari Rohini, Niu Cheng and Li Wei A hybrid approach for named entity and sub-type tagging [Conference] // Proceedings of the sixth conference on Applied natural language processing. - Seattle, Washington : Association for Computational Linguistics, Morristown, NJ, USA, 2000. - pp. 247-254.

Srihari Rohini, Niu Cheng and Li Wei A hybrid approach for named entity and sub-type tagging [Conference] // Proceedings of the sixth conference on Applied natural language processing. - Seattle, Washington : Association for Computational Linguistics Morristown, NJ, USA, 2000. - pp. 247-254.

Srinivasan Savitha H Features for unsupervised document classification [Conference] // Proceedings of the 6th conference on Natural language learning. - [s.l.] : Association for Computational Linguistics Morristown, NJ, USA, 2002. - Vol. 20. - pp. 1-7.

Strong Diane M., Lee Yang W. and Wang Richard Y. Data Quality in Context [Journal] // COMMUNICATIONS OF THE ACM. - [s.l.] : ACM Publication, May 1997. - 5 : Vol. 40. - p. 8.

Systems Stateless Trendsmap [Online] // Trendsmap Real-Time Local Twitter Trends. - March 15, 2010. - http://trendsmap.com/.

Tabor Damon LAFD's One-Man Geek Squad Brings Web 2.0 to Firefighting [Online] // WIRED MAGAZINE. - October 20, 2008. - February 15, 2010. - http://www.wired.com/entertainment/theweb/magazine/16-11/st_firefight.

Takeuchi Koichi and Collier Nigel Use of support vector machines in extended named entity recognition [Conference] // Proceedings of the 6th conference on Natural language learning. - [s.l.] : Association for Computational Linguistics Morristown, NJ, USA, 2002. - Vol. 20. - pp. 1-7.

Talburt John and Bell Mark A Bayesian Approach To The Identification Of Postal Address Lines Utilizing Word Frequencies Derived From Expert Coded Corpora [Conference] // Third International Symposium on Soft Computing for Industry. - Maui, Hawaii : World Automation Congress, 2000. - p. 6.

Talburt John R. [et al.] Entity Identification in Documents Expressing Shared Relationships [Conference] // 11th WSEAS International Conference on SYSTEMS. - Agios Nikolaos, Crete Island, Greece : [s.n.], July, 2007. - pp. 224-229.

Talburt John R. [et al.] Entity Identification Using Indexed Entity Catalogs [Conference] // Proceedings of the 2007 International Conference on Information & Knowledge Engineering (IKE). - Las Vegas, Nevada, USA : [s.n.], 2007. - pp. 338-342.

Talburt John R. A New View of Information Quality [Conference] // Database Grand Conference. - Seoul, South Korea : [s.n.], 2009. - pp. 241-251.

Taskar Ben, Abbeel Pieter and Koller Daphne Discriminative probabilistic models for relational data [Conference] // Proceedings of Uncertainty in Artificial Intelligence . - 2002. - pp. 485-492.

The University of Waikato Weka 3 - Data Mining with Open Source Machine Learning Software in Java [Online] // The University of Waikato. - September 27, 2010. - http://www.cs.waikato.ac.nz/ml/weka/.

Thorp Jer blprnt.blg [Online] // Just-landed: Processing, Twitter, MetaCarta & Hidden Data. - May 11, 2009. - March 15, 2010. - http://blog.blprnt.com/blog/blprnt/just-landed-processing-twitter-metacarta-hidden-data.

Tjong Kim Sang Erik F. Memory-based named entity recognition [Conference] // Proceedings of the 6th conference on Natural language learning. - [s.l.] : Association for Computational Linguistics, Morristown, NJ, USA, 2002. - pp. 1-4.

Twitter Inc. Twitter Search API [Online] // Twitter. - February 28, 2010. - http://search.twitter.com/api/.

Twitter Inc. Twitter Support: Frequently Asked Questions [Online] // Twitter. - November 4, 2008. - February 29, 2010. - http://help.twitter.com/forums/10711/entries/13920-frequently-asked-questions.

Twitter Update and API Limits [Online] // Twitter. - November 30, 2008. - February 29, 2010. - http://twitter.zendesk.com/forums/10711/entries/15364.

Van Rijsbergen Cornelis J Information Retrieval [Book]. - [s.l.] : Butterworth-Heinemann Newton, MA, USA, 1979. - 0408709294.

Ward Wayne and Issar Sunil A class based language model for speech recognition [Conference] // Acoustics, Speech, and Signal Processing. - Atlanta, GA : IEEE, 1996. - Vol. 1. - pp. 416-418.

WBI-ICC (Tech^Edge) WBI - ICC Home Page [Online] // Wright Brothers Institute - Innovation and Collaboration center. - March 1, 2010. - http://www.wbi-icc.com/news/.

Weglarz Geoffrey Two Worlds of Data – Unstructured and Structured [Online] // Information Management Magazine. - September 2004. - October 13, 2009. - http://www.information-management.com/issues/20040901/1009161-1.html.

Wu Ningning [et al.] A method for entity identification in open source documents with partially redacted attributes [Journal] // Journal of Computing Sciences in Colleges. - 2007 . - 5 : Vol. 22. - pp. 138-144.

Wu S. and Crestani F. Data Fusion with Estimated Weights [Conference] // Proceedings of the Eleventh International Conference on Information and Knowledge Management. - McLean : ACM, 2002. - pp. 648 - 651.

Wu Yu-Chieh [et al.] Extracting Named Entities Using Support Vector Machine [Journal] // Knowledge Discovery in Life Science Literature. - [s.l.] : SpringerLink, 2006. - Vol. 3886. - pp. 91-103. - 10.1007/11683568_8.

Yang Hung-chih [et al.] Map-reduce-merge: simplified relational data processing on large clusters [Conference] // Proceedings of the 2007 ACM SIGMOD international conference on Management of data. - Beijing, China : ACM New York, NY, USA, 2007. - pp. 1029-1040.

Yarowsky David Word-sense disambiguation using statistical models of Roget's categories trained on large corpora [Conference] // Proceedings of the 14th conference on Computational linguistics. - Nantes, France : [s.n.], 1992. - Vol. 2. - pp. 454-460.

Zhang Xiaoli [et al.] Investigator name recognition from medical journal articles: a comparative study of SVM and structural SVM [Conference] // Proceedings of the 9th IAPR International Workshop on Document Analysis Systems. - Boston, Massachusetts : ACM New York, NY, USA, 2010. - pp. 121-128.

Zhang Yi, Wang Rui and Oepen Stephan Hybrid Multilingual Parsing with HPSG for SRL [Conference] // Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task. - Boulder, Colorado : Association for Computational Linguistics, 2009. - pp. 31-36.

Zhou GuoDong and Su Jian Named Entity Recognition using an HMM-based Chunk Tagger [Conference] // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL). - Philadelphia, USA : [s.n.], 2002. - pp. 473-480.