TOWARDS A DATA-INTENSIVE APPROACH TO NAMED ENTITY RECOGNITION

(Research-in-Progress)
IQ of Unstructured and Extracted Data

O. Isaac Osesina
oiosesina@ualr.edu

John R. Talburt
jrtalburt@ualr.edu

University of Arkansas at Little Rock, Arkansas, USA

Abstract: Many difficult natural language and machine learning problems are now yielding to data-intensive solutions made possible by the advent of economical, high-performance computing. In data-intensive computing, complex rules systems and statistical models can often be replaced by the ability to scan large volumes of data to find exact or similar instances in which the solution is known. This paper discusses the problem of named entity recognition in unstructured textual information, a discussion of how data-intensive computing methods have evolved to address this problem, and a comparison of results obtained between different data-intensive methods.

Key Words: Named Entity Recognition (NER), Unstructured Textual Information, Information Extraction, Data-Intensive Computing, Entity Resolution, Entity Identification, Text mining

INTRODUCTION

As vast amounts of data are being collected and stored, high-performance computing (HPC) is also becoming more affordable and accessible. As a result many difficult problems that previously required complex algorithmic or statistical models are now yielding to more data-intensive computing solutions that learn from past experiences stored in large corpora of relevant data [1]. Organizations such as Google, Yahoo!, and Bing holding massive amounts of data are able to extract at a relatively fast rate the needed knowledge from the same or similar situations as it has occurred in the past. For example Google translates languages by finding instances of past translations of the same words and phrases without actually "knowing" any of the language rules [1]. They are on the leading edge of a trend towards data-intensive computing, an approach to solving problems by using huge volumes of data and simple applied mathematics [1]. For example, Google spell checker uses past user entries to suggest spellings [2 p. 9]. Data-intensive computing solutions place more emphasis on searching through a wealth of examples than on developing complex rule or statistical models based on a relatively small sample of the data. This perhaps can be summarized by Peter Norvig, Google's Research Director, in his statement that, "All models are wrong, and increasingly you can succeed without them" [1] a twist on the often quoted statement by George Box that "All models are wrong, but some are useful" [3 p. 424].

This paper explores the evolution of research on the problem of named entity recognition (NER) in unstructured textual information (UTI) undertaken at the University of Arkansas at Little Rock (UALR). In particular it discusses how the research has evolved from complex algorithmic to a much more data-intensive approach, and a description of the recently developed methods and results in this area.

NAMED ENTITY RECOGNITION

Entity Recognition is the process of locating a word or a phrase in an unstructured document that references a particular entity such as a person, organization, place, or event. When the entity recognition process also determines (or names) the role of the entity in the structure of the document, the process is called Named Entity Recognition (NER) [4]. For example in the text "XYZ Inc announced today that it has acquired controlling interest in ABC Corp," both XYZ Inc and ABC Corp are references to organizations. However from the standpoint of NER the organizations have different roles, XYZ being the acquiring organization and ABC the acquired organization. The roles and relationships among entities that occur in a family of documents comprise the document ontology. NER is an important aspect of intelligent information extraction and management research [5]. Unstructured data can be defined as data without a conceptual or data type definition [6] and it exists in two broad categories namely bitmap data (e.g. image, audio, and video) and free text data (e.g. e-mail, contract). This research focuses on free text also referred to here as unstructured textual information (UTI). Given that a vast majority (up to 80%) of available data is in unstructured form [7] and that the popularly available data manipulations and analytical tools are inefficient on unstructured data inefficient [8] [9], NER and information extraction in general are important research fields.

The natural language processing (NLP) approach to NER has made very significant progress over the years. Research approaches in NLP range from manual programming of linguistic rules [10][11] to corpus-based learning approaches [12][13][14]. Different applications use linguistic information from a variety of sources to locate entities in unstructured text with a high degree of accuracy. Some of these include lexical resources such as WordNet [15], thesauri such as Roget's [16], and entity catalogs/gazetteer lists such as USPS delivery points, NASA's ADL Gazetteer, and the U.S. people directory, zabasearch.com. Some of the drawbacks with these approaches are the high cost of creating and maintaining these resources, their language specificity and their often domain specificity. Furthermore, they are not robust enough to capture the richness and flexibility exhibited in UTI (e.g. slangs, tweets). On the other hand a NER system that can be applied across different languages and domains (i.e. language and domain independent) would have a relatively lower cost of creation and maintenance (cost can be shared across different languages and domains). Moreover, the entity extraction knowledge is directly transferred across domains, languages and experts specialized in these different fields.

The application of NER techniques can make the information in UTI available for other uses such as creating resource description framework (RDF) for semantic web, populating a relational database table and multi-level indexing. This increased accessibility to information makes this research fall within the scope of Information Quality (IQ) as defined in terms of maximizing the value of organizational data assets [17 p. 148-150].

RELATED WORK

Early research in part-of-speech automation involved the manual programming (which require explicit knowledge of the document language and domain) of language rules for NER tasks [11][10]. Later researchers have introduced the use of probabilistic approaches as a way to cope with the increasing size of the corpora [18][14][19].

Rule-based and decision tree models approach NER with an array of statistical algorithms and manually crafted rules. Paliouras et al. [20] approached the task of reducing manual customization of named-entity recognition and classification (NERC) system to specific domains by using decision trees to automatically acquire NERC "grammar" from text data. Brill [21] introduced the transformation-based error-driven learning, in which learned ordered transformation list from manually annotated corpus is applied to

parsed text obtained from an initial-state annotator such that it results in a simple rule-based approach to learning of linguistic knowledge. Brill's approach is able to offer transformation list even when equivalent decision trees do not exist for a fixed set of primitive queries. Jing et al. proposed an improved transformation based learning based post-processing approach for NER that uses error-driven learning to obtain tuning rules which can be used to improve the results of Japanese NER to different degrees based on the given threshold conditions [22].

Various models with varying degree of complexity have also been introduced to NER tasks. Markov-based models were among the first to be introduced; hidden Markov model (HMM) has been used to stochastically determine optimal linguistic patterns for NER tasks [12][23][24]. Maximum entropy model, unlike HMM, can define a conditional probability of state sequences of observed arbitrary overlapping features [25][22]. Class based models predict the probability of word as a specific named entity based on the n-gram tokens generated from previously named words [26] [27]. Memory-based learning has also been used to develop rules from training instances instead of creating generalized rules for the entire training instances [28][29][30].

Several information extraction systems have been presented at Message Understanding Conferences. Fisher et al. [31] 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. [32] developed FACILE a rule-based system that 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. [33] introduced SIFT; a fully trained information extraction system capable of performing NER tasks using statistical language models trained on annotated data alone.

Several of the existing information extraction systems that apply linguistic tools (e.g. thesauri) often assign tags to entities based on the similarity estimates of its context to other corpora; however the definition of context varies. Some consider context at the document level [34][35] while others consider it sententially. Sentential context extractors have varying linguistic complexity. Sliding window methods define the context of the target word in terms of the neighboring words within a limited distance called a window [36] [37] [38][39]. Shallow methods (e.g. CASS, ANNIE, SEXTANT, FASTUS) identify entities in general categories without attaching semantic meaning or structure to them [40] [41] while deep methods (e.g. MINIPAR RASP) identify entities in a structured manner and attach semantic meaning to them [42][43][44].

Furthermore, systems such as IBM's Unstructured Information Management Architecture [9] and the University of Sheffield's General Architecture for Text Engineering [45] provide an environment whereby the different information extraction components can be shared in a collaborative fashion. In contrast to the above described systems, a truly data-intensive approach must infer any language specific knowledge entirely from access to a large set of annotated examples.

MOTIVATION FOR NER RESEARCH

The research into the use of open source documents for resolving, identifying, disambiguating and/or updating entities attributes in entity catalogs or proprietary repositories has led to much of the research described in this paper[46][47] [48][49][50][51]. In order for organizations to effectively take advantage of the abundant information available online using their current data and analytical tools, the unstructured information must be transformed in to a structure in which the entities of interests within the text can be represented using relational database systems. An example of information that can be frequently updated

using public sources is the different researched techniques for NER, the system requirements, required resources as well as their strengths and weaknesses.

Much of the NER research has been conducted using obituary announcements mainly because of their free and online availability. An example of the application of their research is the timely identification of deceased people in public databases such as the zabasearch.com catalog by using information extracted from online obituary announcements. Although there are a large number of publicly available obituary announcements online, the drawback to using this source is that the information exists in unstructured/free text form, hence it is not directly comparable to structured information. Furthermore, the volume of the available obituaries quickly made it evident that in order to take advantage of this valuable information, an automated method is needed for the purpose of extracting information of interest. In order to be effective, the capability of the method must include associating semantic meanings to identified entities e.g. a name should be identified as belonging to the decedent, decedent's children, parents, siblings, etc. The information about the decedent's relatives contained in the obituaries can also be used in resolving and/or updating other entities in the database.

Talburt et al. [50] in their formal problem formulation described an entity identification process of determining the "best match" of a single identity fragment[1] extracted from unstructured documents from among a set of possible candidates in entity catalogs. Two of the limitations of this method are the catalog may be replete with many similar identity choices and the catalog may be incomplete, hence the correct identity is not among the choices. Therefore, the best-match algorithm is often in the form of a belief function. They discussed two methods of using entity catalogs for entity identification namely single-reference, attribute matching and multiple reference, shared relationship. Single-reference, attribute matching identification is the process by which the attributes of a single entity is compared with those of a set of possible candidates in order to identify the candidate referenced in the document. When the more than one entity fragments can be extracted from the source and a relationship can be asserted among them, multiple references, shared relationship technique can exploit the attribute (dis)similarities among them for the identification of the identity fragments.

TWO APPROACHES TO NER

The research into NER methods has evolved in both the scope of the types of named entities extracted and the methods applied. The manner in which the knowledge base is composed and applied varies between the different NER techniques researched. The general trend has been towards the extraction of more named entities types and reduced dependence on language rules that must be hardcoded. The approaches to NER are classified into algorithmically complex or data-intensive depending on the use of explicit language or grammatical knowledge and/or the structure of the extraction program.

Algorithmically Complex

Algorithmically Complex NER techniques which rely on linguistic and/or language resources, any knowledge of the grammatical structure of the text which may be hardcoded in the extraction program or techniques that use language or linguistic resources as algorithmically complex approach.

Hashemi et al. [47] introduced a NER technique for extracting Names, Titles, and their associations. The paradigm introduced used the knowledge about distances among names and titles as well as character patterns observed from manually extracted names and titles stored in a knowledge base. The words in the text are represented as token, therefore an entity is composed of a token (T) or a block (collection of

[1] Identity fragment is an entity reference with insufficient attributes to identify a particular entity

tokens separated by at least one space). This approach is considered algorithmically complex mainly because of the use of the grammatical knowledge of conjunctions and prepositions (O) during the processing of the tokens and the some of the very formal rules to which a token must conform e.g. a token must start with a capital letter, a word cannot be the first token in a page or in a sentence, unless it is followed by another token. An example of the token processing would represent the string "Marketing Director of ACME" is represented as "TTOT". The KB is composed of entries of one, two, or three consecutive tokens (called clumps) obtained from a name and address corpus. The likelihood scores for the clumps are used to determine the likelihood of strings from the target document being a name, title or their associations.

Chiang et al. [49] used an extractor (FASTUS) to identify named entities and relationships from the text at the sentential level. Since the extractor depends on explicit language knowledge [44] we classified this NER approach also as algorithmically complex. The indentified named entities are given a semantic meaning based on the pattern of the text surrounding it. 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. It requires the several often domain specific keywords and patterns to determine the relationship(s) between entities. This technique was applied to 31 obituary announcements (Nov. 2006) and yielded a precision and recall rates of 37.2% and 20.1% respectively. After further language specific modifications were made to the program, the precision and recall rates increased to 71.4% and 41.3% respectively.

Data-Intensive

When a NER technique does not require any external language or linguistic resource or explicit grammatical knowledge of the document but obtains all its information for from the annotated example corpus (a.k.a. knowledge base), we classify it as data-intensive.

The data-intensive NER approaches currently under research are motivated by the work of Talburt and Bell [52] on a Bayesian identification of so called "floating address lines" using only information from an annotated knowledge base. They implemented an automatic method of identifying standard lines in a US postal address by function (e.g. individual name, business name, street address, etc.). The method classifies each line of a target address into one of 7 functions without any semantic knowledge of the words on each line. Instead it calculates estimates of conditional probability distribution of words in the lines of the target address based on a large corpus of addresses where each line was expert-coded as one of 7 functional types. The expert-coded corpus is preprocessed and summarized into a very-large table containing the frequency of occurrence of every one, two, and three word phrase by line function and by relative position in the line. The function of a line in the target address is inferred by looking up each phrase in the line in the frequency table 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 results in a type assignment for each line.

Using a corpus compiled from 100,000 expert-coded addresses tested against 23 million addresses, the method yielded an overall increase of 7% identification accuracy compared to the original, rule-based production system which used only a few small generic name tables. Although the initial prototype was less accurate in identifying names recorded in last-name-first order, and city-state lines that did not have zip codes compared to the original system, the problem was easily fixed by including these types of records in the corpus and recompiling the frequency table. Because the rectification of identification errors is data driven and does not require the modification of program code, the likelihood that a modification to correct the identification of one record leading to the misidentification of other records is greatly reduced.

NER by Example Approach

The approach described here extends the work of Talburt and Bell because it attempts to extract entities from entirely unstructured text. Although they used an expert-code corpus of addresses to build a statistical model, the line identification scenario assumed that each address 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, any phrase could potentially represent an entity reference meaning that the extraction and identification problems are intertwined making it a more complex problem.

In our current NER approach we sought to avoid the drawbacks in the algorithmically complex approach such as the difficult and, time-consuming task of programming every language rule and possible exceptions and the constraints of linguistic and lexical resources in capturing all the richness and flexibility of natural language.

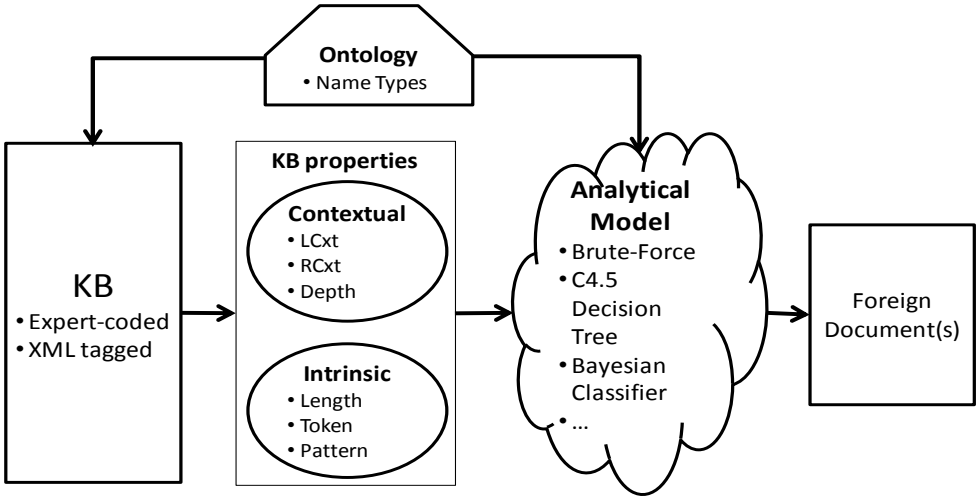

Figure 1: NER Exploiting Context and Intrinsic Properties

Instead of fixing the linguistic and grammatical rules of the language into programming code, we provide the system with several examples of the occurrence of entities of interest in an annotated text, define the necessary attributes and parameters, and then leave it up to the system to infer the extraction model. Furthermore, since no external language resources or catalog is used, the named entity values are used as a glossary to which candidates can be compared to estimate a belief of the resemblance of the candidate value to that of known entities of the same type. Our approach helps in decoupling the language and domain knowledge acquisition part of the system from the analytical extraction model part. This has two very significant advantages namely: (i) the program design for the extraction model does not necessarily need to be modified when a new/rare entity is to be handled and (ii) the extraction model can be applied to different languages and ontology.

The KB is composed of several examples of documents from which named entities of interest are to be extracted. These examples are expert-coded and XML tags are used to annotate named entities of interest as well as the surrounding text (context). For example, the obituary announcement

"Memorial service for William T. Doe, 95, of Boston will be Saturday, Feb. 10, 2007, at 10 a.m."

is expert-coded as

<Context>Memorial service for </Context><Decedent>William T. Doe Jr.</Decedent><Context>, </Context><DecedentAge>95</DecedentAge><Context>, of </Context> <DecedentLastResidence>Boston</DecedentLastResidence><Context> will be Saturday, Feb. 10, 2007, at 10 a.m.</Context>

The properties of the named entities which we categorized as contextual and intrinsic are used to develop the logic (model) for extracting and validating candidate entities (possible entities extracted from the KB). Contextual property refers to the attributes of the surroundings of the named entities in the text and is measured by the parameters:

- Left context (LCxt) – closest sequence of characters preceding the labeled entity
- Right context (RCxt) – closest sequence of characters following a labeled entity
- Depth (Dep) – index position of the entity within the document (unstructured text) expressed as a proportion of the number of characters in the document

Intrinsic property refers to the attributes of the named entity value itself and is measured by the parameters:

- Length (Len) – count of the characters of the entity string
- Token (Tok) – number of word(s) contained in the entity (i.e. number of spaces plus one)
- Pattern (Patrn) – classifying the characters of the entity string as numeric or alphabets

Since this NER approach does not assume any prior language and domain information, it leverages the KB contexts as its "dictionary" and the KB entity values as its "glossary". This implies that even though the extraction logic is independent of language, domain and ontology, the KB is not. Hence a KB can be expected to be effective in extracting entities only from target documents categories for which it contains example(s).

Candidate Extraction

In order to extract/locate a candidate entity from within a target document, the contextual property is used. Starting with either the LCxt or RCxt, the character sequence of the context beginning from the side closest to the annotated named entity is searched for in the target text. If a user-defined pre-determined minimum number of character sequences or all the entire context characters are matched, one character space is skipped and then in a manner similar to the previous context, a second character sequence match using the other context is performed. If the string in the middle of the matched contexts characters is within a specified range of the depth, it is designated as a candidate entity and assigned the same entity type as that surrounded by the used context pair in the KB. For instance given,

Funeral service for Jane S. Doe, 95 of...
← →

Using the expert-coded KB in the above example, part of the LCxt (" service for ") and the RCxt (", ") would be matched in order to extract the string "Jane S. Doe" as a candidate for the entity named decedent. This procedure is then repeated exhaustively for each context pair throughout the text. In starting the context match on the side closest to the annotated entity, we subscribe to the research of Schütze [37] in which he assigns to a context the set of words that occur in proximity. Hence, the hypothesis is that in general, the closer a word is to the annotated entity the more contribution it has to its role and semantic meaning in the text. This procedure coupled with the fact that using an entire context match would require the target document to be almost identical in composition to the expert-coded examples (which is not realistic) are the reasons why partial context match is adopted.

Another method of extracting candidates for consideration is building a decision tree using the contextual property of the KB. Experiments for this were done using the C4.5 classifier. The J48 class (an implementation of C4.5) in Weka was included in our NER software to perform this function. The contextual properties decision tree (Figure 2) built using the contextual properties parameter values, was traversed in order to determine the contexts and depth values combination needed to extract a candidate; and the value of the leaf node is assigned as its entity type.

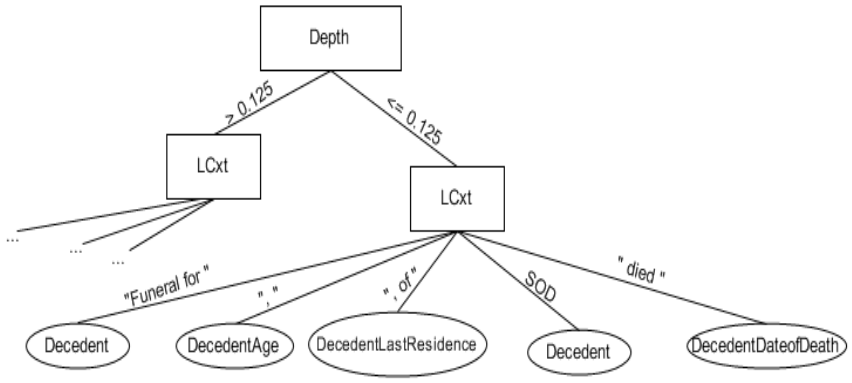

Figure 2: Example of Decision Tree built by J48 algorithm using the contextual attributes. For instance, if Depth is <= 0.125 and LCxt = "Funeral for " or SOD – Start of Document, then the following string is classified as decedent. Another part of the tree is used to determine the terminating point of the string.

Candidate Scoring and Disambiguation

In contrast to many other NER approaches, our current approach considers words simply as a character sequence. No knowledge of the semantic meaning of any of the words surrounding the entities is used or assumed during the candidate extraction and evaluation processes. The only external knowledge used is that of general distinction between alphabetic, numeric and punctuation characters (i.e. a-Z: alphabet, 0-9: numeric, everything else is considered as punctuation characters). A far as we know these are not language dependent, hence it can still be maintained that our approach to building the model is language independent.

As can be expected, several plausible and implausible candidates would be extracted due to the variations in the target document and annotated examples. It is therefore important to devise a belief function that can help in choosing the most likely correct candidates. For the comparison of the candidates, we introduced a term called "candidate strength". The candidate strength is a single valued belief function that unifies the evidence that supports or opposes the candidate. The general premise is that given the context with which the candidate is extracted (contextual properties) and the value of the attributes of the candidate value (intrinsic properties) what is the likelihood that the candidate is an entity reference of that type? Several methods for calculating candidate strength have been attempted; each with its own strengths and weaknesses. Some are briefly described below.

Dispersion Statistics Method

The dispersion statistics approach assigns a score to a candidate based on the product of standard deviations dependent measures and an ambiguity measure. The means and standard deviations of the previously described attributes (Len, Tok, Char (Patrn) and Dep) for each of the different entity types annotated in the KB are calculated and used as benchmarks for assessing candidate values. The z-score of the candidate attribute is used to determine its probability distribution value. The closer a value is to the mean, the higher the probability of its correctness.

The score of the candidate's contexts attribute (LCxt and RCxt) is determined by dividing the number of its occurrence where it indicates the same entity type as the candidate by the total number of occurrence in the KB. This measure of the level of the contexts ambiguity, gives an indication of the candidate context support based on the information in the KB.

$$\text{Candidate Stength}_{\text{Dispersion Statistics}} = \text{f}(\sigma_{\text{Len}}) * \text{f}(\sigma_{\text{Tok}}) * \text{f}(\sigma_{\text{Patrn}}) * \text{f}(\sigma_{\text{Dep}}) * (\text{context ambiguity})$$

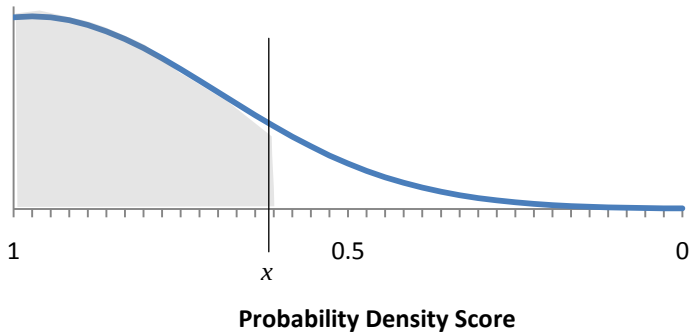

Figure 3: Probability Density Function Chart. The closer x is to the mean the higher the probability of its correctness

This approach was very effective in identifying candidate outliers; however, the impact of a single low attribute score on the final candidate strength can be very severe. For example, a zero score on the length score reduces the final candidate score to zero.

Probabilistic Classifier Method

The classifier method uses the KB attributes to generate probabilistic models that can be used to extract and score candidates. It creates two decision trees; the first is the contextual properties decision tree (Figure 2) and the second is the intrinsic properties decision tree built using the intrinsic properties parameter values. Once the candidate is extracted, the contextual score is calculated as the product of the probability (weight) of the leaf node and the overall ability of the tree in indentifying the particular entity type (entity type f-measure). The second decision tree is traversed using the candidate's intrinsic properties values. If no leaf node can be reached or the value of the leaf node is not the same as that of the contextual properties tree, the candidate is discarded. Otherwise, the intrinsic score is calculated as the product of the probability of the leaf node and the overall ability of the tree in indentifying the particular entity type. Finally, the candidate strength within this method is calculated by multiplying the contextual and intrinsic properties scores.

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

This method is very efficient in handling a very-large KB as the decision tree eliminates redundancies. Furthermore, it gives very robust probability values for different attribute value combinations. On the other hand; the classifier may not include rare attributes in the decision tree, therefore making it difficult to locate candidates with such attributes.

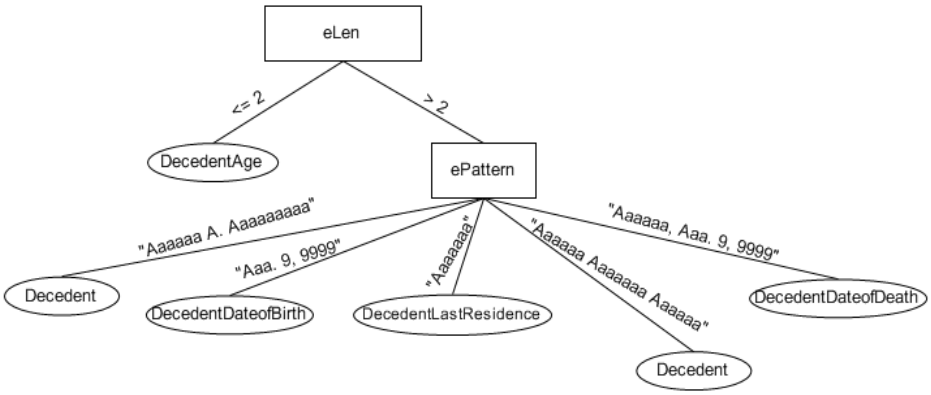

Figure 4: Example of Decision Tree built by J48 algorithm using the intrinsic attributes. For example, if the length of the candidate value is less than or equal to 2, then the candidate is classified as decedent age.

Bayesian Method

The Bayesian method calculates the candidate strength using the conditional probability of the occurrence of all the observed candidate attributes in the KB. In other words, the candidate strength is the probability of the candidate being a type X given that it has $LCxt_i$, $RCxt_i$, Tok_i, Len_i etc. Assume that the intersection of the sets of the entities drawn from the KB using the candidate attributes is called A. The number of correct entities from A is divided by the number of items in A.

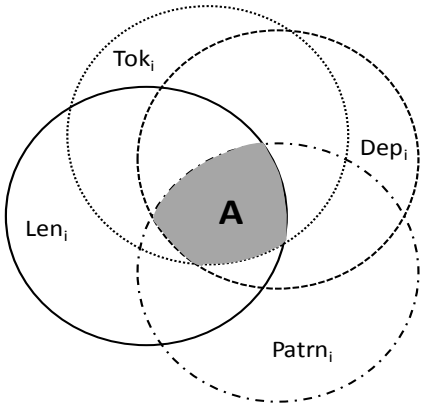

Figure 5: Intersection of entities drawn from the KB using the intrinsic candidate characteristics. A is the intersection of all the four sets of entities i.e. A = $\mathbf{Len_i \cap Tok_i \cap Dep_i \cap Patrn_i}$

This method is implemented by using the attributes of a candidate to iteratively extract entities from the KB. Intersecting candidates are then checked for "correct" entities. Due to the relatively small size of the KB, the conditional probability is given some elbow room in order to improve the chances of finding hits. For example, even though a candidate has length Len_i, a string with a length between Len_i-ε_{Len} and Len_i+ε_{Len} is considered among the set extracted from the KB when using Len_i; where ε is a relatively small number. For our experiments, we calculated ε as three standard deviations of the attribute (except contexts) values of each entity types in the KB.

$$\text{Candidate Stength}_{\text{Bayesian}} = (\#\text{correct entities in A})/(\text{size of set A})$$

A drawback to this approach is that the limited number of examples in the KB may mean that A is empty making the candidate strength equal to zero. Similarly, if the set of entities drawn from the KB using one of the attributes is null, the candidate strength is reduced to zero possibly discarding a correct candidate.

Belief Theory Method

This method is still under investigation and is not discussed in details in this paper. A belief theory such as the Dempster-Shafer theory enables us to increase or decrease our confidence in the correctness of a candidate based on new evidence provided by successive attributes. This method has the advantage that the candidate strength is not nullified due to a single piece of null evidence from any of the observed attributes.

RESULTS DISCUSSION

The source of unstructured text used for our experiment is obituary announcements. Obituary announcements are used as a source of unstructured data because of their free availability in large volume on the internet. In this experiment 50 random obituaries are tested. During each of the 10 runs of the experiment, an obituary announcement from the KB is used as the target document while the remaining 49 are used in the KB. The results described below show the average values of these 10 experiment runs for the dispersion statistics, C4.5 classifier and Bayesian candidate strength calculation methods.

The effectiveness of each method is measured using precision, recall and F-measure; where [49],

$$\text{Precision} = \frac{\text{number of correct answers the system outputs}}{\text{number of answers the system outputs}}$$

$$\text{Recall} = \frac{\text{number of correct answers the system outputs}}{\text{number of possible correct answers}}$$

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

Where β indicates the relative importance of recall and precision.

$$\beta \rightarrow \begin{cases} < 1 & \textit{precision more important} \\ = \mathbf{1} & \textbf{equal importance} \\ > 1 & \textit{recall more important} \end{cases}$$

Table 1 below shows the average values of the precision, recall and f-measure for the 10 runs. We choose the f-measure (β=1) as a simple way to compare the effectiveness of the different measures because it is the harmonic mean of the precision and recall. As illustrated in Table 1, the Bayesian method is the most effective while the C4.5 classifier method is the least effective. The better performance of the Bayesian method compared to the dispersion method can be attributed to the relatively fewer number of returned candidates (due to the different conditions that must be met simultaneously) and high precision. The dispersion statistics, C4.5 classifier and Bayesian methods returned an average of 36.7, 28.2 and 28.2 candidates respectively out of which a respective average of 8.1, 5, 8.6 candidates are correct.

Method	Precision	Recall	F-Measure
Dispersion Statistics Method	0.550	0.206	**0.308**
C4.5 Classifier Method	0.317	0.185	**0.221**
Bayesian Method	0.536	0.305	**0.385**

Table 1: Comparison of average candidate strength values for the three methods

Figure 6 compares the f-measure of the different candidate strength calculation methods by entity types. The vertical axis depicts the average f-measure of each entity type for the 10 experiment runs. It also shows that in general the NER approaches work most effectively for decedent age entity type (f-measure average: 0.62) This high effectiveness specific to this entity type can be attributed to its strong and nearly homogenous intrinsic properties values (most of the decedent ages in the KB are numeric, 2 characters long and a single token) making it relatively easy to effectively filter out wrong values from passing as the age. The entity type "decedent" also enjoys a high f-measure average which is a result of its very strong left context, the place of birth. The spouse, child and parent entities types have the lowest f-measures mainly because they have relatively weak contexts and they also share many intrinsic properties with one another and other entity types e.g. decedent and sibling which makes them difficult to disambiguate. Although the decedent and sibling entity types also have similar attributes to these "weak entities", they benefit mainly from their strong right and left contexts.

We also realized that context provides the strongest support for the candidates. For example, modifications of the dispersion scoring method to include only the context ambiguity score:

$$\text{Candidate Strength}_{\text{Dispersion Statistics}} \cong \text{context ambiguity}$$

Yields an average f-measure of 0.298 (a difference of 0.01 or 3.25%), indicating that the other attributes

only improve the method efficiency negligibly. Furthermore, due to the wide variation in the size of documents (from 78 to 1,457 characters) the depth attribute can be especially detrimental for short documents since the entity offsets are all relatively very close to one another.

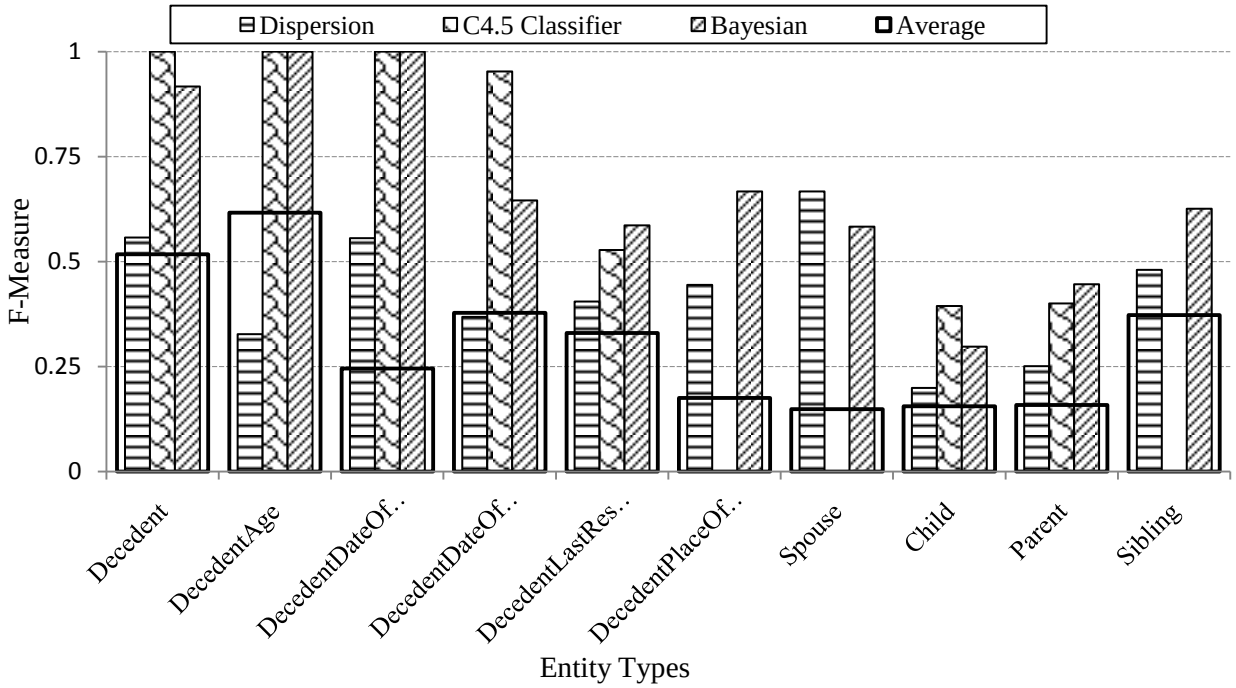

Figure 6: Comparison of the candidate strength calculation methods across entity types. The average is the average of the 3 f-measures under comparison.

As can be observed in Figure 6, the C4.5 classifier method has zero f-measure values for the decedent place of birth, spouse, and sibling. These are two reasons for this. First, the intrinsic properties decision tree generated by the classifier has an overall decedent place of birth f-measure value equal to zero; therefore the candidate strength also equals to zero, hence candidate is discarded. The second reason is that all the candidates extracted as spouse and siblings are all wrong. The effectiveness of the different methods relative to each other and also across entity types provides an opportunity to develop a system that exploits the strengths of the different methods and systems that can be targeted towards the extraction of particular entity types.

CONCLUSION

NER has drawn increased attention over the years. The increased focus on utilizing unstructured data is driving the need for more research into automated processing of textual data. We have discussed our NER research from the perspective of our journey from algorithmically complex towards data-intensive solutions and also presented the techniques and methods used in our current research. Of the three candidate strength calculation methods (dispersion statistics, C4.5 classifier and Bayesian) described, the Bayesian method has the highest efficiency with a relatively moderate number of returned candidates and it high precision. Our examination of the performance of the methods across entity types revealed that age due to its nearly homogeneous intrinsic characteristics effectively filters out wrong candidates from being accepted as the same entity type.

FUTURE WORK

In the continuation of this research, we plan to develop hybrid NER systems based on a combination of the methods described in this paper. The hybrid system would enable us to leverage the strengths of each method based on the entity types for which it performs best. We plan to consider substituting entity transition scoring for depth scoring i.e. with the knowledge of a preceding entity, $entity_{i-1}$ we estimate the likelihood of $entity_i$ following. We also plan to develop an implementation of the Dempster-Shafer belief theory approach for the calculation of the candidate strength. Efforts will also be made to increase the size and scope of the KB in terms of the number of documents as well as the inclusion of documents from other languages and domains. This expansion of the KB should allow us the opportunity to more rigorously compare the performance of our NER approaches within and across languages as well as domains.

ACKNOWLEDGEMENTS

The work described in this paper was supported in part by the U.S. Air Force Research Laboratory at Wright Patterson Air Force Base and UALR

REFERENCE

[1]. Anderson, Chris. The End of Theory: The Data Deluge Makes the Scientific Method Obsolete.;Wired Magazine, 23/6/08.;http://www.wired.com/science/discoveries/magazine/16-07/pb_theory. Accessed: 03/4/10.

[2] ECONOMIST, THE. *Data, data everywhere - A special report on managing information*. 27th Feb., 2010.

[3] Box, George E. P. and Draper, Norman R. *Empirical Model-Building and Response Surfaces*. Wiley, 1987.

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

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

[6]. Weglarz, Geoffrey. Two Worlds of Data – Unstructured and Structured.;Information Management Magazine, Sept 2004.;http://www.information-management.com/issues/20040901/1009161-1.html. Accessed: 13/10/09.

[7] MERRILL LYNCH & CO. *Enterprise Information Portals*. 1998.

[8] Raghuveer, Aravindan, Jindal, Meera, Mokbel, Mohamed F., Debnath, Biplob, and Du, David. Towards efficient search on unstructured data an intelligent-storage approach. *ACM* (September 2007), 951-954.

[9] Ferrucci, David and Lally, Adam. Building an example application with the unstructured information management architecture. *IBM Systems Journal*, 43 , 3 (July 2004), 455-475.

[10] Klein, Sheldon and Simmons, Robert F. A Computational Approach to Grammatical Coding of English Words. *Journal of the ACM*, 10, 3 (July 1963), 334-347.

[11] Harris, Zellig. Harris. String Analysis of Language Structure (1962).

[12] Jelinek, Fred. Self-organized language modeling for speech recognition. In *Readings in Speech Recognition*. Morgan Kaufmann Publishers Inc. , San Francisco, CA, USA, 1990.

[13] Church, Kenneth W. A stochastic parts program and noun phrase parser for unrestricted text. In *International Conference on Acoustics, Speech, and Signal Processing* (Glasgow, Scotland 1989), 695-698.

[14] Bahl, L.R. and Mercer, R.L. Part of speech assignment by a statistical decision algorithm. In *nternational Symposium on Information Theory* (Ronneby, Sweden 1976).

[15] Fellbaum, Christiane. *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA; London,

1998.

[16] Emblen, Donald Lewis. *Mark Roget: The Word and the Man*. Longman Group, London, UK, 1970.

[17] Fisher, Craig, Lauría, Eitel, Chengalur-Smith, Shobha, and Wang, Richard. *Introduction to Information Quality*. MIT Publications, Boston, 2008.

[18] Beale, Andrew David. Lexicon and grammar in probabilistic tagging of written English. In *26th annual meeting on Association for Computational Linguistics* (Buffalo, New York 1988), Association for Computational Linguistics Morristown, NJ, USA , 211-216.

[19] Marcken, Carl G. de. Parsing the LOB corpus. In *28th annual meeting on Association for Computational Linguistics* (Pittsburgh, Pennsylvania 1990), Association for Computational Linguistics Morristown, NJ, USA, 243-251.

[20] Paliouras, Georgios, Karkaletsis, Vangelis, Petasis, Georgios, and Spyropoulos, Constantine D. Learning Decision Trees for Named-Entity Recognition and Classification. In *ECAI Workshop on Machine Learning for Information Extraction* (2000).

[21] Brill, Eric. Transformation-based error-driven learning and natural language processing: a case study in part of speech tagging. *Computational Linguistics*, 21 (1995), 543-565.

[22] Jing, Wang, Dequan, Zheng, and Tiejun, Zhao. Research on Improved TBL Based Japanese NER Post-Processing. In *Advanced Language Processing and Web Information Technology* (Dalian Liaoning, China 2008), IEEE Publications, 145-149.

[23] Merialdo, Bernard. Tagging English text with a probabilistic model. *Computational Linguistics*, 20, 2 (June 1994), 155-171.

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

[25] Mccallum, Andrew, Freitag, Dayne, and Pereira, Fernando. Maximum Entropy Markov Models for Information Extraction and Segmentation. In *Proceedings of the Seventeenth International Conference on Machine Learning* (2000), Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 591-598.

[26] Brown, Peter F., Pietra, Vincent J. Della, deSouza, Peter V., Lai, Jenifer C., and Mercer, Robert L. Class-Based N-Gram Models of Natural Language. *Computational Linguistics*, 18, 4 (December 17, 1990), 467-479.

[27] Ward, Wayne and Issar, Sunil. A class based language model for speech recognition. In *Acoustics, Speech, and Signal Processing* (Atlanta, GA 1996), IEEE, 416-418.

[28] Tjong Kim Sang, Erik F. Memory-based named entity recognition. In *Proceedings of the 6th conference on Natural language learning* (2002), Association for Computational Linguistics, Morristown, NJ, USA, 1-4.

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

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

[31] Fisher, David, Soderlan, Stephen, Feng, Fangfang, and Lehnert, Wendy. Description of the UMass system as used for MUC-6. In *Proceedings of the 6th conference on Message understanding* (Columbia, Maryland 1995), Association for Computational Linguistics Morristown, NJ, USA, 127-140.

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

[33] Miller, Scott, Crystal, Michael, Fox, Heidi et al. BBN: Description Of The SIFT System As Used For MUC-7. In *7th Message Understanding Conference* (1998).

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

[35] Sanderson, Mark and Croft, Bruce. Deriving concept hierarchies from text. In *Proceedings of the 22nd annual*

international ACM SIGIR conference on Research and development in information retrieval (Berkeley, California, United States 1999), ACM New York, NY, USA , 206-213.

[36] Yarowsky, David. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of the 14th conference on Computational linguistics* (Nantes, France 1992), 454-460.

[37] Schütze, Hinrich. Dimensions of meaning. In *IEEE Supercomputing Proceedings* (Minneapolis, MN 1992), 787-796.

[38] Daelemans, Walter, Bosch, Antal Van Den, and Weijters, Ton. IGTree: Using Trees for Compression and Classification in Lazy Learning Algorithms. *Artificial Intelligence Review*, 11, 1-5 (February 1997), 407-723.

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

[40] Abney, Steven. *The SCOL Manual - Version 0.1b*. Tuebingen, 1997.

[41] Bontcheva, Kalina, Dimitrov, Marin, Maynard, Diana, Tablan, Valentin, and Cunningham, Hamish. Shallow Methods for Named Entity Coreference Resolution. In *Traitement Automatique des Langues Naturelles (TALN)* (Nancy, France 2002).

[42] Lin, Dekang. Dependency-based evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems* (Granada, Spain 1998), 28-30.

[43] Briscoe, Ted and Carroll, John. Robust accurate statistical annotation of general text. In *Proceedings of the Third International Conference on Language Resources and Evaluation* (Las Palmas de Gran Canaria 2002), 1499-1504.

[44] Hobbs, Jerry R., Appelt, Douglas, Bear, John, Israel, David, Kameyama, Megumi, and Tyson, Mabry. FASTUS: A System for Extracting Information from Text. In *Proceedings of the workshop on Human Language Technology* (Princeton, New Jersey 1993), Association for Computational Linguistics Morristown, NJ, USA, 133-137.

[45] Cunningham, Hamish, Maynard, Diana, Bontcheva, Kalina, and Tablan, Valentin. GATE: an Architecture for Development of Robust HLT Applications. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (Philadelphia, Pennsylvania 2002), Association for Computational Linguistics Morristown, NJ, USA, 168 - 175.

[46] Hashemi, Ray R., Ford, Charles W., Vamprooyen, Tim, and Talburt, John R. Extraction of Features with Unstructured Representation. In *Proceedings of the IADIS International Conference on WWW/ Internet* (Lisbon, Portugal 2002), International Association for Development of Information Society, 47-53.

[47] Hashemi, Ray, Ford, Charles W., Bansal, A., Sieloff, S., and Talburt, John R. Building Semantic-Rich Patterns for Extracting Features from Online Announcements. In *International Association for Development of Information Society (IADIS) International Conference on WWW/Internet* (Algarve, Portugal 2003).

[48] Wu, Ningning, Talburt, John, Heien, Chris et al. A method for entity identification in open source documents with partially redacted attributes. *Journal of Computing Sciences in Colleges*, 22, 5 (2007), 138-144.

[49] Chiang, Chia-Chu, Talburt, John, Wu, Ningning, Pierce, Elizaberth, Heien, Chris, Gulley, Ebony, and Moore, JaMia. A case study in partial parsing unstructured text. In *Fifth International Conference on Information Technology: New Generations (itng 2008)*, (Vegas, NV 2008), IEEE Press, 447-452.

[50] Talburt, John R., Wu, Ningning, Pierce, Elizabeth, Chiang, Chia-Chu, Heien, Chris, Gulley, Ebony, and Moore, Jamia. Entity Identification in Documents Expressing Shared Relationships. In *11th WSEAS International Conference on SYSTEMS* (Agios Nikolaos, Crete Island, Greece July, 2007), 224-229.

[51] Talburt, John R., Wu, Ningning, Pierce, Elizabeth M., and Hashemi, Ray R. Entity Identification Using Indexed Entity Catalogs. In *Proceedings of the 2007 International Conference on Information & Knowledge Engineering (IKE)* (Las Vegas, Nevada, USA 2007), 338-342.

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