Neural Comput & Applic (2014) 25:533–548
DOI 10.1007/s00521-013-1516-6

REVIEW

Review of information extraction technologies and applications

Sharon Gower Small · Larry Medsker

Received: 16 March 2013 / Accepted: 6 November 2013 / Published online: 1 December 2013

Abstract Information extraction (IE) is an important and growing field, in part because of the development of ubiquitous social media networking millions of people and producing huge collections of textual information. Mined information is being used in a wide array of application areas from targeted marketing of products to intelligence gathering for military and security needs. IE has its roots in artificial intelligence fields including machine learning, logic and search algorithms, computational linguistics, and pattern recognition. This review summarizes the history of IE, surveys the various uses of IE, identifies current technological accomplishments and challenges, and explores the role that neural and adaptive computing might play in future research. A goal for this review is also to encourage practitioners of neural and adaptive computing to look for interesting applications in the important emerging area of IE.

Keywords Information extraction · Information retrieval · Computational linguistics · Machine learning · Neural computing · Adaptive computing · Web information analysis · Web IE · Big Data

S. G. Small · L. Medsker (✉)
Department of Computer Science and Siena College Institute for Artificial Intelligence, Siena College, Loudonville, NY, USA
e-mail: lmedsker@siena.edu

S. G. Small
e-mail: ssmall@siena.edu

L. Medsker
Department of Physics and Astronomy, Siena College, Loudonville, NY, USA

1 Introduction and background on the information extraction field

The journal *Neural Computing and Applications* has for 21 years encouraged work on the basic workings of neural and adaptive computing as well as the search for more ways to apply the fundamentals of this field to important practical applications. This review focuses on the established field of information extraction (IE), which has already lead to many research issues involving artificial intelligence (AI) techniques in established areas of computational linguistics and more recently machine learning. Neural and adaptive computing, while definitely already in use in IE, has the potential for much more improvement of IE applications. The importance of finding more efficient and effective interpretations of massive amounts of data is evident in the rapid increase in social media networking and the emergence of the field of Big Data.

To be able to manipulate vast amounts of unstructured data effectively, automated systems require efficient and accurate methods to derive information structures directly from text. The purpose of adding structure to otherwise flat text is to generate a partial representation of content in a form that can be effectively manipulated by the computer. Specifically, most applications typically require representation that captures key events reported and the attributes of these events, including their role in analyzing a corpus. Information extraction is the field that primarily deals with text structuring.

IE work automates the recognition of interesting information related to pre-specified types of events, entities, or relationships in text from sources such as newswire articles and the Web. IE is of great importance in many text-understanding applications such as in Web intelligence and search engines. IE systems automatically extract structured

Table 1 Terms and concepts in the field of information extraction

Term or concept	Definition and explanation
Information Retrieval	The process of retrieving a ranked set of relevant documents from a corpus based on a user's stated information need
Annotation	The process of marking text relative to a specific information need; e.g., identify/mark all suicide attack mentions in newspaper articles
Pattern/ template	The structure that an information need of interest will commonly appear as, e.g., <terrorist_group> bombed <location>
Entity	An object of interest such as person, location, and organization
Clustering	The process of grouping text based on common attributes, e.g., all text related to corporate takeovers would form in a single cluster
Event	An activity of interest such as a lawsuit or a terrorist action
Precision	A measure of performance in IE: Precision = Number of items found/number of items correct; where items can be entities, relations, events, etc.
Recall	A measure of performance in IE: Recall = Number of items found/total number of items that should have been found; where items can be entities, relations, events, etc.
Relations	Semantic relationship between two named entities; e.g., Employee_Of

information from unstructured and semi-structured machine-readable documents. To extract information from text documents, most IE systems rely on a set of extraction patterns. Extraction patterns are defined based on the syntactic and semantic constraints on the positions of desired entities within natural language sentences. Table 1 contains a set of terms and concepts commonly used in the IE field.

1.1 Evolution of the information extraction field

Historically, the Message Understanding Conference (MUC), which ran from 1987 to 1997, was a forum for researchers to have their IE systems evaluated [1] and was instrumental in encouraging research in this field. MUC designed formal evaluations for detecting named entities, relations (e.g., location_of, employee_of), and events (management succession, terrorist events, etc.). This early research quickly realized the importance of named entities. Many of the items of interest to be extracted from text were either simply the named entities themselves or events and relations, which are even more complex to detect. These events and relations typically were centered around one or more named entities, e.g., "*location* was bombed" and "*person* was hired." Named entity recognizers began by classifying a small set of proper nouns and numerical information into classes such as persons, organizations, and dates [2]. Current state of the art named entity recognizers can spot hundreds of entities of interest from the traditional to more domain-specific entities such as disease, law, and scientific results. These named entity recognizers are over 90 % accurate, very near to human performance on this task.

Research in IE has explored a variety of machine learning techniques. Initially, many IE systems relied on manually built patterns and rules, e.g., SRI's FASTUS system. FASTUS used patterns encoded as a cascade of finite state machines [3] to capture both syntactic and semantic structures in text. For example, the following pattern was defined for the terrorist incidents domain of MUC-4:

<PERP> attacked <Human Target>'s <Physical Target> in <Location> <Date> with <Device>

SRI prepared 95 such patterns with 253 trigger words for their participation in MUC-4.

This work evolved into supervised learning methods, where a model is learned from a set of training examples that have been manually annotated. The model can then be used to extract information from new documents. This process was expected to be simpler than manually creating patterns and rules by hand and therefore faster and more accurate than the pattern matching technique. BBN's statistical language model used this approach for their MUC system which performed extremely well [4] at MUC and is currently at the core of their leading IE system (Identifinder). However, the effort to annotate training data for each new domain was found to be more challenging than expected. Just for MUC-7, the system was trained on $\sim$500,000 words hand annotated from the New York Times newswire text covering the domains of air disasters and space technology (MUC-7 domains). Researchers began looking for ways to ease the annotation burden. This led to semi-supervised machine learning approaches where much less training data are required as well as unsupervised learning approaches, which try to discover patterns and relations using just the texts themselves.

Semi-supervised learning methods also use sets of annotated data, but they are much smaller than what is used in supervised methods, and they are augmented with large amounts of un-annotated data. The typical semi-supervised learning process is that annotations of some examples, named entities, events, and relations can be used to find more examples and thus more patterns from un-annotated text. For example, in named entity recognition, given the manual annotation of the company_name General Electric, we observe that this is still a company in our un-annotated documents. Therefore, we can automatically extract new patterns for company_name where we see General Electric used in different contexts. The BaLIE, Baseline Information

Extraction, system learned to accurately spot 100 entities from just a dozen examples of each entity type [5]. This contrasts with the previously required annotations of thousands of documents or handmade rule patterns. This approach has also been applied to event extraction. Yangarber et al. [6] demonstrated a procedure for extracting events beginning with several seed patterns, where no manual annotation was needed. They built their work on research by Riloff [7] who claimed that patterns could be extracted from a corpus of documents could be separated into two sets, a set that contained the event and those that did not. Riloff argued this approach would work because documents that contained an event of interest in one pattern form would likely contain the event in other relevant pattern forms. Yangarber used the seed patterns to recognize documents that should be in the relevant set. Other patterns were automatically extracted from the relevant set and added to the seed patterns, and the process was repeated. This semi-supervised learning approach is often called bootstrapping.

Unsupervised learning methods attempt to glean information automatically from the texts themselves. Research has primarily focused on discovering relations in a narrower domain, typically from newspaper articles. Early research by Hasegawa et al. [8] first tagged documents for named entities. Then, they extracted *relations* between all pairs of entities, e.g., ORGANIZATION-LOCATION and PERSON–PERSON, that occur together in a small span of text (within a 7-word window). These pairs of entities are saved with their candidate relations: the sequence of words between the entities. These triples are then grouped into sets of the same types of entity pairs, and clustering techniques are applied in order to group triples into the same semantic relation. Other groups have attempted to utilize structured information to automatically extract information in an unsupervised manner. Dalvi et al. [9] explored the possibility of utilizing tables found on the Web. Unsupervised machine learning for IE is still an active area of research, and approaches differ significantly.

For a more detailed description of evaluation in IE as well as non-English IE techniques, the reader may refer to Piskoraski and Yangarber [10].

2 Supporting technologies for information extraction

Many supporting technologies are utilized in the field of information extraction ranging from "simple" information retrieval to very complex machine learning algorithms.

2.1 Information retrieval

The first step in most IE systems is document retrieval. IE is typically applied against large document collections, and some of the required sub-steps (part of speech tagging, parsing, etc.) are process intensive. Information retrieval (IR) is used to reduce the set of initial documents that IE would be applied to, thereby realizing a system with a reasonable response time. Traditional IR systems start by building an index of the terms in a document collection, typically recording the term location and frequency. For a given query, the IR system will then return a ranked list of documents by computing the similarity between the query and the documents, using the index and a similarity function. Some of the best-known IR methods are based upon the vector space model used in the Smart system [11] and the probabilistic model used in the InQuery system [12]. Systems have also been built that index and retrieve text at the passage level [13].

2.2 Part of speech taggers and parsers

Part of speech (POS) taggers are very commonly utilized in information extraction systems. They may be used to assist in pattern/template matching or as an input attribute for machine learning models. POS taggers accept text as input and tag all of the words as to their part of speech, e.g., noun, verb, and adjective. One of the best performing and most commonly used taggers is the Stanford POS tagger [14]. Parsers are used in similar ways, to assist in pattern/template matching and also as attributes for machine learning models. They perform a deeper semantic processing, accepting text as input and outputting the grammatical structure, e.g., subject, object, and predicate. Parsers have the potential to improve the overall performance of an IE system versus POS taggers, but it is a significantly slower process to run on text. Stanford has also developed and provided a free version of a parser that has been heavily used in IE research [15].

2.3 Wordnet

Wordnet is a large lexical database of nouns, verbs, adverbs, and adjectives developed at Princeton University [16]. Words are grouped into synonym sets, synsets, that can be utilized effectively in IE systems. Wordnet currently has 155,287 words/phrases and 117,659 synsets. In Fig. 1, one can see how it would be a useful resource for the IE system that would like to be able to automatically determine whether rifle is a synonym for machine gun, where they both may fill the role in a pattern looking for a weapon.

2.4 Coreference resolution

Coreference resolution (CR) is the task of resolving different mentions of the same entities. This is typically

```
rifle
    => firearm, piece, small-arm
      => gun
        => weapon, arm, weapon system
          => weaponry, arms…
machine gun
    => firearm, piece, small-arm
      => gun
        => weapon, arm, weapon system
          => weaponry, arms…
```

Fig. 1 A portion of Wordnet hypernym trees for "rifle" and "machine gun"

across multiple sentences and may even be across documents or even the "simple" case within the same sentence, e.g., "The *company* fired 50 % of their employees so *they* did not have to declare bankruptcy." CR on the MUC-7 collection has not been able to achieve higher than ~60 % precision and ~60 % recall. Supervised learning is the common approach for CR machine learning, specifically the mention-pair model [17]. This model identifies pairs of entity mentions utilizing a set of annotated examples, classifies them as either coreferent pairs or disreferent pairs, and then clusters them based on these classifications. Bird et al. [18] demonstrated a twofold increase in the performance of relation extraction when also applying coreference resolution.

2.5 Annotation tools

Annotation tools have been built for a variety of purposes. The CSLU Toolkit [19] is a suite of tools used for annotating spoken language. Similarly, the EMU System [20] is a speech database management system that supports multilevel annotations. Systems have been created that allow users to also readily build their own annotation tools such as AGTK [21]. The multimodal tool DAT [19] was developed to assist testing of the DAMSL annotation scheme. With DAT, annotators were able to listen to the actual dialogs as well as view the transcripts. While these tools are all highly effective for their respective tasks, DAT is unique in its synchronized view of both event action and chat utterances. Small et al. [22] developed RAT, the Relational Annotation Tool, to assist in the creation of multimodal action-communication corpora from online massively multi-player games (MMGs). MMGs typically involve groups of players (5–30) who control their avatars, perform various activities (questing, competing, fighting, etc.), and communicate via chat or speech using assumed screen names.

Annotation tools for information extraction are typically very task specific, and most often a simple tool is developed to meet the needs of a particular project. Their basic functionality is to speed up the annotation process, which can be very time-consuming and error-prone. The tool gives the annotator a GUI that allows them to quickly highlight words and/or phrases and tag their role. For example, the tool created for the Siena Environmental Review Project (SERP) [23] allowed the annotator to tag eight different attributes for a water quality testing (WQT) event in a database of public comments related to hydrofracking regulations. While a comment is displayed, annotators were able to quickly highlight words and/or phrases that they identify as one of the eight WQT attributes. There were eight buttons that then allowed the annotator to easily mark which attribute is currently highlighted. The output of the tool was saved in XML format for future automatic processing. The following is an example WQT annotated sentence:

> All <*impacted*>homeowner</*impacted*> <*geofeature*>water wells</*geofeature*> within a distance of <*geography*>2,000 feet</*geography*> beyond the outer boundary of wellhead horizontal projection arrays should be <*requirement*>sampled</*requirement*> prior to drilling and tested to establish <*time frame*> baseline</*time frame*> water quality according to accepted <*regulator*>EPA</*regulator*> methods.

2.6 Machine learning (ML)

ML is an essential subfield of AI that models and enhances the ability of human intelligent behavior to increase knowledge and to solve problems not previously encountered in a majority of IE systems. Most current ML systems are characterized by the ability to identify known properties of data and are typically supervised learning approaches. Current research and development in machine learning can be classified broadly into symbol-based, statistical learning (including Bayesian and Markov methods), and connectionist approaches. Symbolic methods include knowledge representation and search algorithms on tree structures, and various practical systems have been developed using rule-based, evolutionary, and fuzzy logic techniques. While neural computing and support vector machine (SVM) techniques are already used in IE, they have good potential for additional new uses in IE research and applications. Various ML tools match the different types of IE work, which is classified in three modes: ones that involve supervised, semi-supervised, and unsupervised exploration of corpora.

2.6.1 Support vector machines and neural computing

Support vector machines (SVM), given a set of inputs, produce outputs in one of two classes [24, 25]. Training

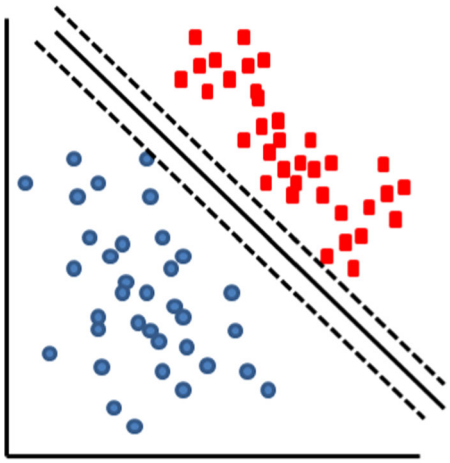

Fig. 2 The *solid line* is the maximum margin separator between the two classes of output

examples are modeled as data points, where x is represented from a linear function over the features of the input and y is the binary output produced for those combinations of features. The SVM finds the maximum margin separator between the two types of outputs. This can be visualized in the solid line (Fig. 2 below), which represents a boundary between all of the input examples with the longest possible distance to example points. This model can then be used to determine the output for new sets of inputs. The original model has been extended to multiclass SVM systems to classify data into more than two classes [26].

Figure 2 is a two-dimensional example where a linear decision boundary exists. It is important to note that in some cases, a linear decision boundary does not exist. SVMs are able to handle nonlinear decision boundaries by transforming the data points to a higher-dimensional space (feature space) where a linear boundary can be found [27].

SVMs are a popular supervised machine learning approach in the field of information extraction. The GATE project [28–30] used SVMs for named entity recognition. Named entities can span several words, and some GATE projects have trained two SVMs for each entity type—one to detect the start of the named entity and one to detect the end of the named entity. Each word in a document was tagged as being part of a named entity phrase or not part of a named entity phrase. Each word then had an associated set of input features, e.g., part of speech and semantic class, that could be used with this judgment as the SVM training data. Their system achieved an *F* Score of 89.7 % on the CoNNL-2003 shared task data collection [30]. The CoNNL (Conference on Natural Language Learning) has been meeting since 1997 and has included a shared task since 2003 [31]. This shared task supplies the researcher with testing and training sets of data allowing for systems and results to be comparable to those working on the same task. In 2003, the shared task had researchers working on language-independent (English and German) named entity recognizers.

Mayfield et al. [32] also used SVMs for named entity recognition. They had a much larger set of feature attributes. They not only represented the features of each individual word in the training, but they include the features of the words surrounding it—3 words before and 3 words after for a window size of seven. They also tested their system on the CoNNL data collection and achieved an *F* Score of 90.85 %, very good results but not a significant difference from the GATE work.

2.6.2 *Neural computing*

The various adaptive capabilities in neural computing give researchers several ways to try to improve the performance of IE systems. Artificial neural networks (ANN) use external text data to optimize the weights and structure of different collections of nodes that store and then use information for analyzing new samples of words and phrases. The methods that can potentially be effective in IE range from supervised feedforward multilayer networks to unsupervised and semi-supervised learning modes.

For IE tasks, supervised systems can map labeled text data for a given task (e.g., event extraction, POS tagging, and Named Entity Recognizers (NER)) to categories and patterns, based on selected external training data with known input and output values, and then use the trained ANN to analyze a new corpus of text for related tasks. Unsupervised systems such as those based on self-organizing feature mapping can be used to determine patterns that characterize sample text and use the resulting network to analyze new text. Associative techniques, derived from the ideas underlying Hopfield nets and the Boltzmann machine, can optimize sets of weights to represent and characterize features of words that make up textual passages. Deep learning algorithms are making progress toward text understanding that is closer to human capabilities.

Experiments using ANNs were completed by Hammerton [33] in the CoNLL-2003 shared task. He used a recurrent neural network, which he refers to as Long Short-Term Memory (LSTM), for NER. For each sentence, the two-pass network first gathers information that is resolved during the second pass to produce an output. A self-organizing map is used to analyze sequences and generate representations for the lexical items presented to the LSTM, and orthogonal representations are used to represent the parts of speech and chunk tags. This system achieved very good performance on the English corpus, with a precision of 76 % and recall of 66 %.

Turian [34] evaluated the performance of NERs and chunkers when word representations learned in an unsupervised process are added as extra word features to the NER and chunker systems (Chunkers are a faster alternative to fully parsing a sentence, typically just identifying noun groups and verb groups). They compared different techniques for inducing the word representations including ANNs. Both ANNs that were evaluated improved performance over the baseline for both the NER and chunker. The best performing neural model was based on Collobert and Weston's [35] system, which uses a deep neural network with multiple layers that models distinct features of the system. Their system processed the input sentence by several layers of feature extraction, i.e., the first layer extracted word features and the second layer extracted sentence features. The experiments based on this system achieved an F score of 75.51 % on the MUC-7 data collection for NER, which was a significant improvement over their baseline of 67.48 %. This deep neural network was later improved on with the SENNA system [36], where they achieved an F Score of 89.31 % for NER.

"SENNA" (Semantic/syntactic Extraction using a Neural Network Architecture) is a standalone version of their architecture, written in the C language. SENNA uses unlabeled data sets to discover internal representations that prove useful for all the tasks of interest and a fast and efficient "all purpose" NL tagger. It uses neural computing techniques to build a tagging system with low computational costs. Their architecture and learning method address tasks including POS tagging, chunking, NER, and semantic role labeling (Semantic Role Labeling aims at giving a semantic role to a syntactic constituent of a sentence). A goal was to avoid manual input requiring specific domain knowledge that needs to be optimized for each application, and instead, they learn internal representations from large amounts of unlabeled data.

Unlike fully supervised approaches, they preprocess features as little as possible and then use a multilayer neural network architecture that is trained in an end-to-end fashion. The architecture takes the input sentence and learns several layers of feature extraction that process the inputs. The features computed by the deep layers of the network are automatically trained by backpropagation to be relevant to the task. Key to their architecture is the ability to perform well using mostly raw words. The ability of their method to learn good word representations is crucial to the approach. They use extremely large unlabeled data sets (i.e., 631 million words from English Wikipedia) to train language models that compute scores describing the acceptability of a piece of text. These language models are again large neural networks.

Another body of research has been done using unsupervised neural networks for natural language processing. The PhD thesis by Timo Honkela [37] contains a thorough review of that body of work, along with his study of self-organizing maps (SOM) in NLP. His focus is on developing categories for words from a natural language corpus. Each word is initially represented by a unique random vector, which is then modified through an analysis of all occurrences of the word in a corpus. Thus, each word vector is given further information—e.g., about the types of words preceding and succeeding the word—and thereby given a contextual fingerprint. Once a set of words is defined in this manner, the SOM is used to find categories or clusters of words that have similar characteristics. These word category maps can then be used to determine the context of expressions and gain insight into semantic aspects for information extraction. A subsequent work by Honkela et al. [38] in their Media Map project applies this SOM technique to develop a user interface to an information system for projects and publications.

3 Research related to information extraction

3.1 Knowledge discovery from databases (KDD)

KDD is an overall process for extracting information and knowledge from large databases. Several current data-oriented technologies are related to IE, and some have similar goals. Given that they operate on data that are highly structured, the KDD techniques are quite different from IE approaches. Applications for KDD may learn from experience and use unsupervised methods to find properties of databases that were previously unknown. Knowledge discovery can also be described as the non-trivial extraction of implicit, previously unknown, and potentially useful information from data [39].

One step in the process is the analysis of a specified database using one or more of a variety of data mining tools and methods. The KDD process comprises the following:

1. Analyze requirements: understand the user goals; the domain and prior knowledge
2. Identify and/or create the corpus from which knowledge is to be discovered
3. Preprocess the data to characterize the data, ensure correctness, and reduce redundancy and extraneous data
4. Optimize the data through goal-related ways involving representation, transformation, and projection
5. Determine the type of data mining to be done (clustering, classification, etc.)
6. Select the best data mining method and tool to match the task and goals

7. Perform data mining to produce potentially interesting results
8. Analyze the results, determine usefulness, and integrate into the body of prior knowledge.

Overall, KDD tends to be used to create new, and difficult to anticipate, knowledge from large collections of data. The complexity of the data makes human discernment of knowledge difficult, even to the point of not being able to specify the goals for data mining of new information and knowledge.

3.2 Data mining

As a step in the KDD process, data mining uses specific tools and systems to carry out the analysis of large data collections. Data mining tasks can be done automatically or semi-automatically to extract previously unrecognized patterns that may prove to be useful. Data mining includes text mining, image mining, web mining, predictive analytics, and much of the techniques used in dealing with massive data sets. Data mining applications automate for very large data collections what humans can do or conceive only for small data sets. This application of machine learning addresses the large and growing demand for finding useful information from large, complex data collections. Many societal and business requirements are based on mining data when specific targets can be described to guide the analysis of very large and dynamic bodies of data.

Types of applications that can benefit from data mining are:

1. pattern recognition for determining categories of data in a particular collection
2. classification of new data into predefined categories
3. searches for anomalies and exceptions in data
4. identification of associations between data items, such as products purchased by consumers
5. prediction and forecasting based on models created from data on past behaviors and events.

4 Research areas in information extraction

4.1 Research programs

The number of researchers working in the field of IE is quite large. Table 2 gives examples of important programs but is not meant to be inclusive of all contributors to the field.

Table 2 Key research programs in information extraction

Research group	Lead personnel	Research topics and systems
Artificial Intelligence Center SRI International	Douglas Appelt, Andrew Kehler	Information extraction from free text, FASTUS system
GATE Information Extraction University of Sheffield	Hamish Cunningham	GATE Built-in IE component ANNIE
Information Extraction and Synthesis Laboratory UMass Amherst	Andrew McCallum	Mining from unstructured text Factorie and Mallet toolkits
Information Extraction from Text BBN Technologies	Ralph Weischedel	Named Entity Recognizer: Identifinder
Information Sciences Institute University of Southern California	Jerry Hobbs	FASTUS system, temporal representation, commonsense
Language Technologies Institute Carnegie Mellon University	William Cohen, Jamie Callan	Machine learning, distributed IR, adaptive information filtering
Next Generation CiteSeer Project Penn State University	C. Lee Giles	Automated Digital Library AckSeer search engine
Open Information Extraction University of Washington	Oren Etzioni and Dan Weld	Open Information Extraction ReVerb OIE software, Big Data
Retrieval Group NIST Information Technology Laboratory	Ellen Voorhees	TREC workshops, crowdsourcing, medical record retrieval
Tetherless World Constellation Rensselaer Polytechnic Institute	James Hendler	Semantic Web, distributed information technology, Big Data

4.2 New research areas

The massive, and growing, amount of data and information resulting from the advent of the Web has created large challenges and opportunities for information management and application development. This has given a new incentive and challenge for information extraction research

groups, both from the sheer volume of data and from the need for efficient translation to knowledge bases that are readily accessible to the public and the broader research communities.

4.2.1 Web-based IE research

"Big Data" has emerged as the name for a new research area to address challenges of managing the large amounts of data in the modern world, and information extraction has a key role to play [40, 41]. Big Data is a consequence of several factors including the ubiquitous digitalization of data and use of sensors, the reduction in data storage costs and increase in bandwidth, and the ongoing advances in AI technologies such as machine learning. Other terms include analytics and data science. The goal is to find new and useful knowledge in data. An emerging aspect of this research is construction of new knowledge through innovative ways of extracting information from the interactions of massive numbers of people, such as in Semantic Web research at RPI's Tetherless World Constellation [40].

Web IE systems focus specifically on the problem of the data and information explosion: the dynamic and unstructured nature of today's information sources and the rising cost of extracting and managing information. Traditional methods for linguistic analysis do not exploit the tags and layout formats implicit in online text. Current approaches to Web IE use less of the linguistically rigorous techniques using wrapper features, especially using automatic induction of rules. Still, adaptive methods are needed to handle the variety of types of text—well-structured pages to nearly free text and hybrid types of textual data in the Web's collection of unstructured documents. This means transforming Web data into forms that are amenable to automated processing—relational forms and marking the data with which reasoning can be done. Thus, a Web IE system might analyze natural language text into a database of extracted information features that can be combined with contextual knowledge for reasoning from a classification model. Linked Data refers to publishing and connecting structured data on the Web and is used by a number of data providers to create a global Web of Data.

The accumulation and use of our new bodies of knowledge requires efficient and effective IE technologies. The Open IE project at the University of Washington [41] is an example of new research and technology projects addressing the increasing bottleneck in the use of information and knowledge with more efficient ways to query arbitrary text from various domains on the Web. For example, their research on IE software aims at automatically identifying and extracting binary relationships from English sentences for Web-scale applications where target relations cannot be specified in advance and speed is important.

4.2.2 Information extraction from research papers

The increasing use of IE systems for tasks such as literature search and information searches for hiring decisions demands high levels of accuracy. The following are examples of IE research that could contribute to the goal of satisfying these needs.

Research at Carnegie Mellon University, exploring Hidden Markov Model structures for IE models, is applying statistical machine learning techniques that have been useful in other areas of AI [42]. They specifically focus on how to learn model structure from labeled and unlabeled data. They explore strategies for learning the model structure automatically from data. They have demonstrated that distantly labeled data (a method for automatic labeling by relaxing strict rules for relations) can be used to set model parameters to improve information extraction accuracy. A demonstration example has been the task of extracting important fields from the headers of computer science research papers.

Research at the University of Massachusetts Amherst looks at conditional random field (CRF) approaches to the task of extracting various common fields from the headers and citation of research papers [43]. While the theory of CRFs has been developing, challenges to applied research remain when confronted with the types of dynamic large-scale data in the practical environment of today. The CRF research explores several factors, including variations on Gaussian, exponential, and hyperbolic-L1 priors for improved regularization. They are achieving new state-of-the-art performance in comparison with the previous best SVM and HMM results.

4.2.3 Automatic generation of metadata

Machine learning techniques have been applied to the extraction of information from digital libraries and collections, and a prime example is CiteSeer, a digital library and search engine for scientific literature [44–47]. The use of support vector machine (SVM) technology can give better results than using standard machine learning tools. An example is a system developed for extraction of information from the header parts of research papers [45]. Header lines are put into one or more classes followed by an iterative convergence process using predicted class labels for previous neighbor lines. Searches for best chunk boundaries of each line can further improve the metadata results. Recent work [46] produced AckSeer for automatically extracting acknowledgments in scientific papers. Current research on systems for automatic generation of metadata use techniques that aim to improve usability and provide scalability [47].

4.2.4 SERP: Siena's environmental review project: IE meets environmental science and online comments

Online comments are a relatively new type of data source IE researchers are exploring. Online comments are quite unique from electronic forms of text (e.g., newswire) that are typically used in computational linguistics (CL) research. Many comments are quite short (a single word), while others are very long (11,135 words). Many are made up of just questions, while others are fully made up of uppercase words with many exclamation points. The SERP project looks to explore how simple IE techniques would work on this type of document collections.

Natural gas extraction using hydraulic fracturing (hydrofracking) is a very controversial activity. Tens of thousands of public comments have been generated discussing updating regulatory oversight of this process. Government agencies must respond to the public concerns, but it is a significant challenge if not an impossible task for environmental scientists to manually analyze all of the comments in a timely manner. SERP project researchers use CL techniques to analyze a collection of 27,085 online public comments to automatically spot the mention of specific environmental impacts of interest [23].

Specifically, SERP analyzes the public comments that were submitted to the New York State Department of Environmental Conservation (DEC) in reference to its 2011 revised Draft Supplemental Generic Environmental Impact Statement (rdSGEIS), which proposes high-volume hydraulic fracturing (HVHF) regulations. Electronic copies of these public comments were obtained by sending a Freedom of information law (FOIL) request to DEC. The comment collection relayed the public's thoughts on environmental and economic implications that HVHF may impose. Typically, a comment contains an expressed concern or rejection made by the commenter in relation to a particular environmental or economic effect. Three actual public comments are as follows:

> "What right does a corporation have in devastating our waterways for personal gain? This creek is stocked every year with trout. Where are the studies to determine the impact not only on the water levels, but the fish?"
>
> "It's not worth it!!"
>
> "I am basically for the establishment of high-volume hydraulic fracturing in the State of New York if the process is closely monitored by state authorities and under specific environmental considerations… I would like to see a reparation fund set up in case there are environmental problems. Perhaps a well-head fee would work out best."

SERP experiments with the topic of environmental impact on water quality. Two experts in the field of Environmental Science manually identified attributes that consistently reoccurred in comments discussing WQT (water quality testing). It is important to note that attributes were deliberately of a high level in order to eventually scale this to a broader range of environmental testing events. They identified a total of eight attributes, which were typical to a WQT comment:

1. *Requirement*: What the comment is calling for; the action that is being proposed.
2. *Regulator*: The party who is carrying out the *Requirement*.
3. *Impacted*: Third parties who the event will have an effect on. The testing *Requirement* will likely benefit and be for these people.
4. *Operator*: Those directly involved in carrying out the event (HVHF) process.
5. *Geofeature*: The subject of testing to which the comment is referring (what they want to be tested).
6. *Time Frame*: When the suggested *Requirement* should be carried out.
7. *Fiscal*: Who is financially liable for the testing.
8. *Geography*: Specific whereabouts for the testing to occur.

The experts annotated a set of comments looking to identify mentions of WQT. The following is an example WQT annotated sentence:

> All <*impacted*>homeowner</*impacted*> <*geofeature*>water wells</*geofeature*> within a distance of <*geography*>2,000 feet</*geography*> beyond the outer boundary of wellhead horizontal projection arrays should be <*requirement*>sampled</*requirement*> prior to drilling and tested to establish <*time frame*>baseline</*time frame*> water quality according to accepted <*regulator*>EPA</*regulator*> methods.

Approximately 450 comments have been annotated in which 175 WQT mentions were tagged. These annotations were then automatically processed to extract the keywords and phrases that associate with each of the eight attributes. Table 3 shows a subset of the words and phrases that were tagged by the annotator for these attributes and how many sentences contained that attribute.

Experiments were run against a 30-comment test collection, which contained a little over 300 sentences. Two different rules were tested based on high attribute occurrence counts:

1. Requirement and Geofeature
2. Requirement, Geofeature, and Time Frame

Table 3 Number of sentences that each attribute was tagged by our annotator and some example words/phrases for each attribute

Attribute	# of Sentences	Common words/phrases
Requirement	157	Well testing, assessments, sampling, monitoring…
Regulator	29	Local Health, DEC, EPA, Legislature, DOH…
Impacted	34	Landowners, lease holder, local citizens…
Operator	49	Drilling operator, gas drillers, energy companies…
Geofeature	98	Groundwater, water wells, aquifers, watersheds…
Time Frame	99	Pre-drill, weekly, regular basis, prior…
Fiscal	32	Pay, hire, paid, funded by, fiscal, financed…
Geography	40	500 ft, mile, near well, 1,000 meters, 2,000 feet…

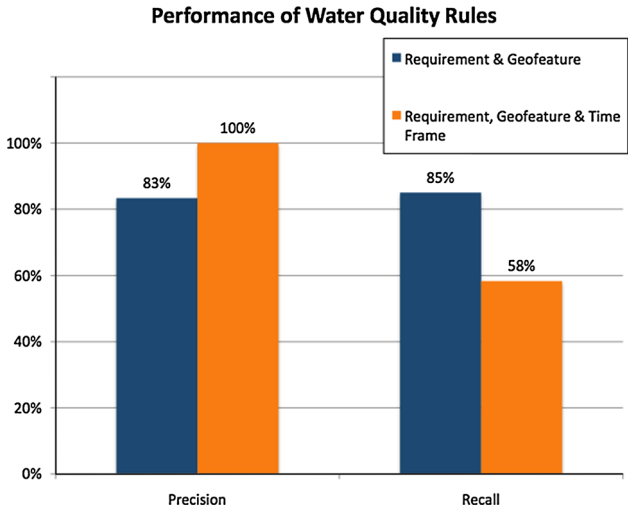

Fig. 3 Results of SERP's WQT spotter

Experiments were scored on traditional recall and precision criteria:

Recall = the number of WQTs we found/the total number of WQTs
Precision = # of correct WQTs we found/the total # of WQTs we found

Precision and recall results were quite different for the two rules. Looking for the existence of the Requirement and Geofeature attributes had 85 % recall and 83 % precision. SERP's Requirement, Geofeature, and TimeFrame rule had a perfect precision but lower recall, finding only 58 % of our WQT sentences (Fig. 3).

Given that an application goal would be to find WQT mentions for Environment Scientist manual review, it was better to have a higher recall at the sake of a slightly lower precision. Therefore, error analysis was performed on the Requirement and Geofeature rule, specifically looking at the WQT mentions the system failed to find. In most all of the cases, the Geofeature was actually not mentioned in the sentence, e.g.:

> Combine this with the fact that we cannot test for contamination since we do not (1) know all of the chemicals that are in the hydraulic fluids and (2) know little of the toxicological effects of these reagents (or the LD50).

Finally, the WQT spotter was run on all 27,085 electronic comments. These comments contained 254,162 sentences. SERP discovered 1,591 WQT mentions. This results in a significantly smaller set for the Environmental Scientists and agencies to manually review. This work has shown that "simple" IE techniques can be used to process and "understand" the content of public comments.

5 Selected examples of tools for information extraction

The following sections focus on important current tools for developing IE applications. The selection of particular application examples focused on topics that may be interesting to researchers and application developers who work in the adaptive computing area.

5.1 Unstructured information management architecture (UIMA)

UIMA is industry standard for content analytics [48]. UIMA is a software architecture specification for multimodal analytics and designed for processing unstructured information, such as email, video files, and other media from human interactions. Some structure is provided in XML documents, but more work is required to transform information from the syntax and semantics understandable by people but not adequately encoded for machine use. The UIMA Software Developers Kit (SDK) for Java applications is available at http://www.ibm.com/developerworks/data/downloads/uima/.

5.2 Waikato environment for knowledge analysis (WEKA)

The WEKA [49] software collects a variety of tools for work in data mining and other areas of machine learning. The workbench tools include traditional learning algorithms for search trees, tools for regression, classification, and clustering, as well as specific IE-oriented software for tasks such as rule mining and attribute selection.

Preprocessing and initial data testing can be done via data visualization and statistical analysis.

5.3 General architecture for text engineering (GATE)

GATE is a Java suite of tools originally developed at the University of Sheffield beginning in 1995 and now used worldwide by a large community of scientists, companies, teachers, and students for all sorts of natural language processing tasks, including information extraction in many languages [28–30]. GATE includes an information extraction system called ANNIE (A Nearly New Information Extraction System), which is a set of modules for tokenizing, sentence splitting, tagging parts of speech, transducing name entities, and co-referencing. ANNIE can be used as is to provide basic information extraction functionality, or provide a starting point for more specific tasks. GATE-based applications often generate vast quantities of information including natural language text, semantic annotations, and ontological information.

The GATE process illustrated below shows the kind of steps involved in information extraction as well as giving a specific example of an IE application system.

5.3.1 The two-minute guide to helping people find stuff with GATE (Private communication from Dr. Hamish Cunningham)

1. Take one large pile of text (documents, emails, tweets, patents, papers, transcripts, blogs, comments, acts of parliament, and so on and so forth)—call this your corpus.
2. Pick a structured description of interesting things in the text (a telephone directory or chemical taxonomy or something from the Linked Data cloud)—call this your ontology.
3. Use GATE Teamware to mark up a gold standard example set of annotations of the corpus (1) relative to the ontology (2).
4. Use GATE Developer to build a semantic annotation pipeline to do the annotation job automatically and measure performance against the gold standard.
5. Take the pipeline from (4) and apply it to your text pile using GATE Cloud (or embed it in your own systems using GATE Embedded).
6. Use GATE Mimir to store the annotations relative to the ontology in a multi-paradigm index server. (For techies: this sits in the backroom as a RESTful web service.)
7. Use Ontotext KIM to add semantic search, knowledge facet search, ontology browsing, entity popularity graphing, time series graphing, annotation structure search, and (last but not least) boolean full-text search (more techy stuff: mash up these types of search with your existing UIs).

Hey presto, you have state-of-the-art information management applying your ontology to your corpus (and a sustainable process).

5.4 Crowdsourcing for information extraction

Crowdsourcing refers to the division of a complex task over a large number of people for the purpose of problem solving and decision making. In the context of IE, a network of people of varying resources and abilities can potentially pool small individual contributions from experience, knowledge, and distributed data gathering to address significant problems, possibly of societal significance. The rapid growth of Web-based social networks and online communities presents many opportunities to produce new information and knowledge.

Crowdsourcing systems can combine human collective intelligence with machine learning for hybrid systems for work on shared tasks and shared data. Crowdsourcing for IE tasks can address text, audio, and video sources of information and have as a goal the assessment of the relevance of items to stated tasks. Current projects focus on a range of levels, research on methods and human factors issues for using crowdsourcing effectively (see e.g., [50]), the use of IE tools for analyzing results, and commercial systems such as Amazon Mechanical Turk.

6 Future directions and opportunities

The extensive work in IE, illustrated through examples in previous sections, demonstrates great strides and also significant challenges. Further research is needed to meet the many emerging needs for applications to extract new information and knowledge.

6.1 Selected current challenges for IE

Event extraction, one of the most difficult of the IE tasks, is a very active area of research. The best reported performance on the MUC-7 data collection has not been able to achieve higher than 70 % precision, and this was at the cost of a significantly low recall, around 30 %. The highest recall systems, just below 50 % had lower precision, also just below 50 %. This reported performance has been in areas of supervised and semi-supervised learning approaches. As discussed earlier in this paper, unsupervised learning, attempting to automatically discover the events (and relations) that exist in document collections, is a much harder problem than the supervised and semi-supervised

approaches. Overall, for unsupervised approaches, more work has been done on relation extraction versus event extraction, as it is the easier of the two tasks, but both are still active research areas.

As described in Sect. 2.4, coreference resolution is important to IE because it is has been shown that if performance was to improve in this area, we would see a resulting performance improvement in other areas of IE (see, e.g., [18]). Event extraction that involves different mentions of the same entities (potentially spanning multiple sentences or documents) would be improved if we were able to resolve these mentions. Coreference resolution on the MUC-7 collection has not been able to achieve higher than $\sim$60 % precision and $\sim$60 % recall.

Generally, IE systems need to be more robust and accurate. Manual modification of domain-specific linguistic knowledge for new similar tasks is time-consuming and a source of errors. Future work on IE systems could benefit from adaptive methods that might improve the understanding of the semantics of text [51].

6.2 Potential new research opportunities in IE

The type and scale of problems that will be addressed with IE are well represented by the state and future of the Web. The growing volume and types of data from scientific work, education, business, government, and society will increasingly require automated systems for information extraction and knowledge discovery [52–56]. Innovations leading to new products and services for the society of the future will require advances in IE for a variety of areas such as bioinformatics, time series and forecasting, evolutionary computing, business intelligence, legal and medical systems, knowledge discovery in educational information systems, multimedia mining, and military intelligence. Many of the extensions of current techniques will require unsupervised methods that can discover useful information and knowledge from large amounts of unstructured data and text.

The many components of the WEKA workbench, described in Sect. 5.2 above, can provide guidance for where adaptive computing techniques—including SVM, supervised neural networks, unsupervised neural computing methods, genetic algorithms [57], and fuzzy systems—may be used as alternate tools or as part of hybrid systems for IE work [58, 59]. Future IE systems could potentially be improved by utilizing adaptive computing methods (neural nets, GA, etc.) in something similar to a pseudo-relevance feedback loop as used in IR systems. This process may be able to improve initial IE templates and achieve higher accuracy [51, 60]. The following examples of current research projects represent areas that potentially could make use of neural and adaptive computing approaches.

6.2.1 Medical information retrieval systems (MIRS)

The Siena College Institute for Artificial Intelligence (SCIAI) has several IE projects that look for potential uses of adaptive computing. SCIAI work on IE from medical data started with the 2012 Medical Records TREC competition [61], which had the task of processing lists of randomly selected queries against a large corpus of medical records. The objective was to simulate searches for patients who might meet the criteria for participating in particular clinical trials, and the data set had an average of 15 reports from each of approximately 100 K patient visits to a hospital. The records, which comprised one month of reports from multiple hospitals, came from the University of Pittsburgh NLP Repository and were de-identified in regard to specific patient names. The Medical Record track organizers from TREC also provided year 2011 judgment sets, produced by medical professionals at the Oregon Health Science University, that were then used for testing MIRS software at different states of development.

The information extraction process was done in two separate ways: one used an index created from open source IR system Lucene [62] and an alternate method was based on principles of neural computing. For each topic, the MIRS system was required to search the medical records corpus and return information as a ranked list of the top 10 relevant hospital "visits," which were proxies for specific patients. The first step was traditional keyword analysis with Lucene to index the medical records corpus. Then the English statements of the requirements for a patient to be a candidate for a clinical trial were translated into named entity templates that could be run against the corpus. This involved analyzing results to identify patients who were relevant and irrelevant to the requirement and also results that missed relevant records.

The exploration of potential uses of neural computing includes a process for translating indexed words and phrases into a numerical format suitable for analysis by neural computing software. As with the traditional IE process, the output from the Lucene indexing produces a table of keywords extracted from each topic statement. That table is then expanded so each keyword is mapped to a subset of equivalent medical keywords.

Each subset can be used to produce a table with entries of ones and zeros that produces a digital pattern that is an equivalent representation of the text of a particular requirement statement (See Fig. 4). This table becomes the data set that can be used for training a neural network. The trained net can then be used to look for patterns in the records for hospital visits, if the latter are similarly mapped

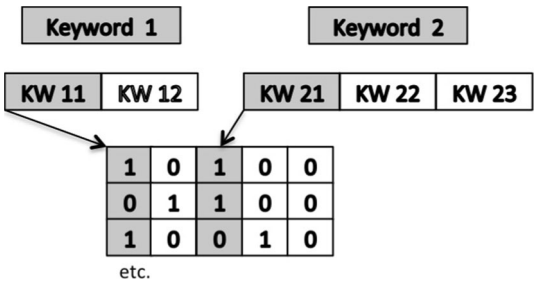

Fig. 4 Example of an expansion of initial keywords and translation to a format that is suitable for neural network analysis

to the ones and zeroes format, to identify patients who might qualify for a clinical trial. Grouping the digital results for all the records associated with one patient visit for a given topic statement gives an another picture of the patient and gives statistical information for assessing relevance of a patient to the topic statement.

The preliminary result of this work is a design for neural processing of requirement statements and medical records for efficient mapping of the important keywords from a task statement into a digital expanded set of medically equivalent words that are relevant for automated searches using neural computation. We have also created a hybrid model that may lead to effective integration of neural computing techniques with conventional text analysis capabilities.

6.2.2 *SCEPS: social-computational event prediction system*

SCEPS research is being done at the Siena College Institute for Artificial Intelligence to apply IE techniques for detecting temporal effects in unstructured text from social media. Societal events are often framed by changes in communication among the populations that these events are affecting. These changes may be detected in and extracted from online open source social media such as blogs and micro-blogs. The extracted communications may then be used to predict significant societal events, such as political crises, mass violence, riots, mass migrations, economic instability, resource shortages, and responses to natural disasters [53–56]. Elements of communications can be extracted and analyzed according to social event categories (SECs) as defined by subject matter experts (SMEs). SMEs define encompassing sets of keywords and phrases relevant to each SEC.

An Internet-scale array of machines and people builds on the everyday use of social media sites by a very large population of people. SCEPS discovery of knowledge (predicted social events), using a hybrid system of social media sites and people, has the potential to increase our understanding and use a knowledge base to predict important social events. Such a system can potentially learn and exhibit intelligence not present in people or computers alone. The research fields targeted by this project encompass IE and other aspects of artificial intelligence, along with applied research in the social science fields of economics and political science.

SCEPS is a hybrid approach, not only in the sense of multiple AI and non-AI technologies working together, each doing what they do best, but also the partnership between people and machine. The task is to blend and balance the constituent sub-systems producing text. The human element of the hybrid system approach is the group of SME's and designers, and developers who are part of the learning phase, using extracted information from social media sources. As the non-human part of the adaptive system becomes more intelligent, the human contribution decreases. At any point, the autonomous, operational system of course has to work on its own, but the accuracy will go from whatever initial level is achieved to a level that meets the goals of a task.

This project is investigating the use of standard IE methods along with adaptive computing techniques such as independent component analysis (ICA), which is particularly useful for extracting patterns from complex mixtures of source signals. ICA can produce patterns of variables that make up mixed-source signals so that correlated keywords and phrases can be identified quickly from complex source data. The FastICA algorithm [63, 64] is readily available as open-source code. Slices of data at selected regular time intervals can be used as input to the FastICA algorithm.

Our learning approach for a given societal events category (SEC) is to use intelligent operators on collections of data matrices comprising the number of occurrences of keywords and phrases derived from IE snapshots over a specified set of time intervals. The operators, using ICA, identify different keywords and phrases that together illuminate trends and patterns as potential events of interest. The process searches open source collections for the occurrences of specific keywords and phrases that are determined, in consultation with SEC-specific subject matter experts, as elemental indicators of potential societal events of interest. The ICA processing is intended to produce output for the number of times important text has been discovered in social media and indicate levels of situations as normal, warnings, or alerts. Through iterative processes of feedback from the autonomous operational system to human SMEs and designers, the system may be modified at the meta-level to reflect new rules about patterns in the body of knowledge that are evolving. A goal is to have subsequent versions of the system become more intelligent with more capability to learn from future data and produce more reliable alerts and warnings.

6.2.3 RPI Tetherless World project

The World Wide Web affects all aspects of our life, and new multi-disciplinary research is needed to realize the potential opportunities for revolutionary capabilities. Models of the Web need to elucidate the architectural principles and the basic social values involved in its ubiquitous impact on our lives. A new research agenda must target the Web and its use in a new and creative way. RPI's Tetherless World Constellation [40] addresses the emerging area of "Web Science," focusing on the World Wide Web and its future use, exploring the research and engineering principles that underlie the Web, and developing new technologies and languages that expand the capabilities of the Web. TWC work is within three themes Future Web, Xinformatics, and Semantic Foundations. TWC personnel work on the design of new techniques to explore social, scientific [e.g., 65], and legal impacts of the evolving technologies deployed on the Web. Areas of study include Semantic Web technology, tetherless and mobile Web access, social networking and collaboration technologies for the Web. Current research and applications for information extraction may have powerful roles in this work, along with scientific and mathematical techniques from many disciplines, to explore the modeling of the Web and making the next generation Web natural to use while being responsive to the growing variety of policy and social needs.

6.2.4 Big Data

The emerging research and development area called "Big Data" could lead to important new applications for innovations from improving sales and recommending products—to scientific research, personalizing medicine, and understanding and mitigating problems from climate change. Diverse projects [66–70] are investigating IE for the many information sources and forms of data on the Web, such as open domain event extraction from Twitter [68]. As another example, research by Lin et al. [69] aims at automatically creating very large knowledge bases of general facts using Entity Linking. They have developed techniques for coupling information extraction and entity linking over millions of high-precision textual extractions from a corpus of 500 million Web documents.

6.3 Summary

The need for improvement in automatic discovery of the events (and relations) that exist in document collections is a challenge for information extraction that may require unsupervised approaches. Relation extraction may be a more amenable problem for research using adaptive technology, and further work on the even more challenging need for event extraction could follow for researchers in the neural computing field.

Research and application development in the information extraction field need to draw upon the best advances in AI research and technology, including work on adaptive computing. The imperatives of today's Web-centric world, with exploding amounts and ubiquitous sources of data, and the societal impacts of Web interactivity present unprecedented opportunities for creating knowledge that will greatly impact our lives. New emerging areas of research and development such as the Semantic Web, Web Science, and Big Data, present exciting opportunities in which neural computing may play an important role.

References

1. Chinchor N, Lewis DD, Hirschman L (1993) Evaluating message understanding systems: an analysis of the third message understanding conference (MUC-3). Comput Linguist 19(3):409–449
2. Bikel DM, Schwartz R, Weischedel RM (1999) An algorithm that learns what's in a name. Mach Learn J Spec Issue Nat Lang Learn 34(1–3):211–231
3. Appelt, DE, Hobbs J, Bear J, Israel D, Tyson M (1993) FASTUS: a finite-state processor for information extraction from real-world text. In: Proceedings IJCAI-93, pp 1172–1178
4. Miller S, Crystal M, Fox H, Ramshaw L, Schwartz R, Stoner R, Weischedel R (1998) Algorithms that learn to extract information; BBN: description of the SIFT system as used for MUC-7. In: Proceedings of the seventh annual message understanding conference (MUC-7), 17 pp
5. Nadeau D (2007) PhD Thesis. Ottawa-Carleton Institute for Computer Science, School of Information Technology and Engineering, University of Ottawa
6. Yangarber R, Grishman Tapanainen RP, Huttunen S (2000) Unsupervised discovery of scenario-level patterns for information extraction. In: Proceedings of the applied natural language processing conference (ANLP 2000), pp 282–289
7. Riloff, E (1996) Automatically generating extraction patterns from untagged text. In: Proceedings of the thirteenth national conference on artificial intelligence (AAAI-96), pp 1044–1049
8. Hasegawa H, Satoshi S, Grishman R (2004) Discovering relations among named entities from large corpora. In: Proceeding of ACL-2004, 8 pp
9. Dalvi N, Kumar R, Soliman M (2011) Automatic wrappers for large scale web extraction. Proc VLDB Endowment 4(4):219–230
10. Piskoraski J, Yangarber R (2013) Information extraction: past, present and future, Chapter 2. In: Poibeau et al (eds) Multi-source, multilingual information extraction and summarization 11, theory and applications of natural language processing. doi:10.1007/978-3-642-28569-1_2, Springer
11. Buckley C (1985) Implementation of the Smart information retrieval system. Cornell University Department of Computer Science Technical Report, 37 pp
12. Callan JP, Croft WB, and Harding SM (1992) The INQUERY retrieval system. In: Proceedings of the third international conference on database and expert systems applications, pp 78–83
13. Cormack GV, Clarke CLA, Palmer CR, Samuel SL (2000) Passage-based query refinement (MultiText experiments for TREC-6). Inf Process Manage 36(1):133–153

14. Toutanova K, Klein D, Manning C, and Singer Y. (2003) Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of HLT-NAACL, pp 252–259
15. http://nlp.stanford.edu/software/tagger.shtml
16. Miller GA (1995) Wordnet: a lexical database for English. Commun ACM 38(11):39–41
17. Soon WM, Ng HT, Lim DCY (2001) A machine learning approach to coreference resolution of noun phrases. Comput Linguist 27(4):521–544
18. Bird S, Liberman M (2001) A formal framework for linguistic annotation. Speech Commun 33(1–2):23–60
19. Core MG and Allen JF (1997) Coding dialogs with the DAMSL annotation scheme. In: Working notes of AAAI fall symposium on communicative action in humans and machines. CSLU Toolkit: http://www.cslu.ogi.edu/toolkit
20. Cassidy S, Harrington J (2001) Multi-level annotation in the Emu speech database management system. Speech Commun 33:61–77
21. Maeda, K, Bird S, Ma X, Lee H (2002) Creating annotation tools with the annotation graph toolkit. In: Proceedings of the third international conference on language resources and evaluation, 8 pp
22. Small, SG, Strzalkowski T, Stommer-Galley J (2012) Multi-modal annotation of quest games in second life. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, pp 171–179
23. Small SG, Booker J (2013) Hydrofracking comments meets computational linguistics. Submitted to the 7th international AAAI conference on weblogs and social media
24. Joachims T (2002) Learning to classify text using support vector machines: methods, theory, and algorithms. Kluwer Academic, Dordrecht
25. Cortez C, Vapnik VN (1995) Support-vector networks. Mach Learn 20(3):273–297
26. Ji Y, Sun S (2013) Multitask multiclass support vector machines: model and experiments. Pattern Recogn 46(3):914–924
27. Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: Haussler D (ed) 5th annual ACM workshop on COLT, pp 144–152, Pittsburgh, PA. ACM Press
28. Li Y, Bontcheva K, Cunningham, H (2005) SVM based learning system for information extraction. In: Proceedings of the Sheffield machine learning workshop
29. Li Y, Bontcheva K, Cunningham H (2005) Using uneven margins SVM and perceptron for information extraction. In: Proceedings of the ninth conference on computational natural language learning CoNLL-2005, pp 72–79
30. Li Y, Bontcheva K, Cunningham H (2004) SVM based learning system for information extraction. Determ Stat Methods Mach Learn 3635:319–339
31. Sang EF, Kim T, Meulder FD (2003) Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In: Proceedings of CoNLL-2003, vol 4, pp 142–147
32. Mayfield J, McNamee P, Piatko C (2003). Named entity recognition using hundreds of thousands of features. In: Proceedings of CoNLL-2003, vol 4, pp 184–187
33. Hammerton J (2003) Named entity recognition with long short-term memory. In: Proceedings of the seventh conference on natural language learning at HLT-NAACL, vol 4, pp 172–175
34. Turian J, Ratinov L, and Bengio Y (2010) Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th annual meeting of the association for computational linguistics, pp 384–394
35. Collobert R, Weston J (2008) A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th international conference on machine learning, Finland
36. Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P (2011) Natural language processing (almost) from scratch. J Mach Learn Res 12:2493–2537
37. Honkela T (1997) Self-organizing maps in natural language processing, PhD Thesis. Helsinski University of Technology
38. Honkela T, Laaksonen J, Törrö H, Tenhunen J (2011) Media Map: a multilingual document map with a design interface, WSOM 2011, LNCS 6731, pp 247–256, Springer
39. Frawley WJ, Piatetsky-Shapiro G, Matheus CJ (1992) Knowledge discovery in databases: an overview. AI Mag 13(3):57
40. Hendler J (2013) Thetherless World Constellation at RPI. http://tw.rpi.edu/wiki/Tetherless_World_Constellation_at_RPI
41. Etzioni O, Fader A, Christensen J, Soderland S, Mausam (2011) Open information extraction: the second generation. IJCAI 2011:3–10
42. Dalvi B, Cohen W, Callan J (2012) WebSets: extracting sets of entities from the web using unsupervised information extraction. In: Proceedings of the fifth ACM international conference on web search and data mining, pp 243–252
43. Peng F, McCallum A (2006) Accurate information extraction from research papers using conditional random fields. Inf Process Manag 42(4):963–979. http://people.cs.umass.edu/∼mccallum/papers/hlt2004.pdf
44. Giles CL, Bollacker KD, Lawrence S (1998) Citeseer: an automatic citation indexing system. In: Digital Libraries, pp 89–98
45. Han H, Giles CL, Manavoglu, E, Zha H, Zhang Z, Fox EA (2003) Automatic document metadata extraction using support vector machines, ACM/IEEE joint conference on Digital Libraries (JCDL 2003), pp 37–48
46. Khabsa M, Treeratpituk P, Lee Giles CL (2012) AckSeer: a repository and search engine for automatically extracted acknowledgments from digital libraries. In: Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries (JCDL 2012), pp 185–194
47. Teregowda PB, Councill IG, Fernandez JP, Khabsa M, Zheng S, Giles CL (2010) SeerSuite: developing a scalable and reliable application framework for building digital libraries by crawling the web. In: 1st USENIX conference on web application development
48. Ferrucci D et al (2006) Towards an interoperability standard for text and multi-modal analytics. IBM Research Report, RC24122 (W0611-188), 106 pp. http://domino.research.ibm.com/library/cyberdig.nsf/papers/1898F3F640FEF47E8525723C00551250/$File/rc24122.pdf
49. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The WEKA data mining software: an update. SIGKDD Explor 11(1):10–18
50. Alonso O, Rose DE, Stewart B (2008) Crowdsourcing for relevance evaluation. SIGIR Forum 42(2):9–15
51. Luger GF (2009) Artificial intelligence, 6th edn, pp 664–665
52. Cunningham H, Tablan V, Roberts A, Bontcheva K (2013) Getting more out of biomedical documents with GATE's full lifecycle open source text analytics. PLoS Comput Biol 9(2):e1002854
53. Khabsa M, Koppman S, Giles CL (2012) Towards building and analyzing a social network of acknowledgments in scientific and academic documents. In: Social computing, behavioral–cultural modeling and prediction—5th international conference (SBP 2012), pp 357–364
54. Radwanick S (2011) The rise of social networking in Latin America: how social media is shaping Latin America's digital landscape. In: http://www.comscore.com/Press_Events/Presentations_Whitepapers/2011/The_Rise_of_Social_Networking_in_Latin_America
55. Signorini A, Segre AM, Polgreen PM (2011) The use of Twitter to track levels of disease activity and public concern in the US during the Inlfuenza A H1N1 pandemic. PLoS One 6(5):e19467. doi:10.1371/journal.pone.0019467

56. Eccarius-Kelly V (2007) Counterterrorism policies and the revolutionary movement of Tupac Amaru: the unmasking of Peru's National Security State. In: Forest JJF (ed) Countering terrorism in the 21st century, vol 3. Praeger Security International, Westport, pp 463–484
57. Atkinson-Abutridy J, Mellish C, Aitken S (2004) Combining information extraction with genetic algorithms for text mining. IEEE Intell Syst 19(3):22–30
58. Downey D, Etzioni O, Soderland S (2010) Analysis of a probabilistic model of redundancy in unsupervised information extraction. Artif Intell 174(11):726–748
59. Yates A, Etzioni O (2009) Unsupervised methods for determining object and relation synonyms on the Web. J Artif Intell Res 34:255–296
60. Nahm UY, Mooney RJ (2000) A mutually beneficial integration of data mining and information extraction. In: Proceedings of the American association for artificial intelligence conference, pp 627–632
61. Medsker L, Small SG, Rivadereira C, Reynolds A, Afzali M (2012) The Siena College medical information retrieval system (MIRS). In: The twenty-first text retrieval conference proceedings (TREC2012)
62. Apache Lucene™ is an open source high-performance, full-featured text search engine. http://lucene.apache.org/
63. Hyvärinen A, Karhunen J, Oja E (2001) Independent component analysis. Wiley, New York
64. Hyvärinen A, Zhang K, Shimizu S, Hoyer PO (2010) Estimation of a structural vector autoregression model using non-gaussianity. J Mach Learn Res 11:1709–1731
65. Wilson N, Wang H, McGuinness DL (2012) Scientific names and descriptions for organisms on the Semantic Web. In: Proceedings of 2nd international workshop on linked science 2012—Tackling Big Data at ISWC 2012
66. Bizer C, Heath T, Berners-Lee T (2009) Linked data—the story so far, special issue on linked data. In: Heath T, Hepp M, Bizer C (eds) International Journal on Semantic Web and Information Systems (IJSWIS) 5(3). http://eventseer.net/e/4789/, http://tomheath.com/papers/bizer-heath-berners-lee-ijswis-linked-data.pdf
67. Sakaki T, Okazaki M, Matsuo Y (2010) Earthquake shakes Twitter users: real-time event detection by social sensors. In: WWW'10 Proceedings of the 19th international conference on World Wide Web, pp 851–860
68. Ritter A, Mausam, Etzioni O, Clark S (2012) Open domain event extraction from Twitter. In: ACM SIGKDD conference on knowledge discovery and data mining (SIGKDD), pp 1104–1112
69. Lin T, Mausam, Etzioni O (2012) Entity linking at web scale. In: Joint workshop on automatic knowledge base construction and Web-scale knowledge extraction
70. Small S, Strzalkowski T (2010) (Tacitly) Collaborative question answering utilizing Web trails. In: Proceedings of the international conference on language resources and evaluation; Workshop on web question answering, pp 36–42