

INFORMATION EXTRACTION

BYLINE

[Heng Ji, Computer Science Department, New York University, hengji@cs.nyu.edu]

SYNONYMS

NONE

DEFINITION

Information Extraction (IE) is a task of extracting pre-specified types of facts from written texts or speech transcripts, and converting them into structured representations (e.g., databases).

IE terminologies are explained via an example as follows.

- **Input Sentence:**

Media tycoon Barry Diller on Wednesday quit as chief of Vivendi Universal Entertainment, the entertainment unit of French giant Vivendi Universal whose future appears up for grabs.

- **IE output:**

- Entities:

Person Entity: {Media tycoon, Barry Diller}

Organization Entity: {Vivendi Universal Entertainment, the entertainment unit}

Organization Entity: {French giant, Vivendi Universal}

- “Part-Whole” relation:

{Vivendi Universal Entertainment, the entertainment unit} is part of {French giant, Vivendi Universal}.

- “End-Position” event.

The above sentence includes a “Personnel_End-Position” event mention, with the trigger word which most clearly expresses the event occurrence, the position, the person who quit the position, the organization, and the time during which the event happened:

Trigger	Quit	
Person	Barry Diller	Media tycoon
Organization	Vivendi Universal Entertainment	the entertainment unit of French giant Vivendi Universal
Position	Chief	
Time-within	Wednesday	

Table 1. Event Extraction Example

HISTORICAL BACKGROUND

The earliest IE system was directed by Naomi Sager of the Linguistic String Project group [1] in the medical domain. However, the specific task of information extraction was formally evaluated through the U.S. Defense Advanced Research Projects Agency (DARPA) sponsored Message Understanding Conferences (MUC) program from 1987 to 1998 [2].

There were four specific evaluations: Named entity, coreference and template element reflected in the evaluation tasks introduced for MUC-6, and template relation introduced in MUC-7.

The MUC tasks have been inherited by the U.S. National Institute of Standards and Technology (NIST) Automatic Content Extraction (ACE) program¹, with more general types of entities/relations/events defined. ACE includes the following tasks.

Entity Detection and Recognition

ACE defines the following terminologies for the entity detection and recognition task:

entity: an object or a set of objects in one of the semantic categories of interest

mention: a reference to an entity (typically, a noun phrase)

name mention: a reference by name to an entity

nominal mention: a reference by a common noun or noun phrase to an entity

Seven types of entities were defined: PER (persons), ORG (organizations), GPE ('geo-political entities' – locations which are also political units, such as countries, counties, and cities), LOC (other locations without governments, such as bodies of water and mountains), FAC (facility), WEA (Weapon) and VEH (Vehicle) mentioned in an input document. This task was proposed in 2000 and evaluated on English, and then expanded to include Chinese and Arabic in 2003, Spanish in 2007.

Relation Detection and Recognition

The relation detection task was proposed in 2002, aiming to find specified types of semantic relations between pairs of entities. ACE 2007 had 7 types of relations, with 19 subtypes. The following table lists some examples.

Relation Type	Example
Agent-Artifact (User-Owner-Inventor-Manufacturer)	Rubin Military Design, the makers of the Kursk
ORG-Affiliation (Employment)	Mr. Smith, the CEO of Microsoft
Gen-Affiliation (Citizen-Resident-Religion-Ethnicity)	Salzburg Red Cross officials
Physical (Near)	a town some 50 miles south of Salzburg

Table 2. Examples of the ACE Relation Types

Event Detection and Recognition

ACE defined 8 types of events, with 33 subtypes. Some examples are presented in Table 3:

Event Type	Example
Movement (Transport)	Homeless people have been moved to schools
Business (Start-ORG)	Schweitzer founded a hospital in 1913
Conflict (Attack)	The attack on Gaza killed 13 people
Personnel (Start-Position)	Cornell Medical Center recruited 12 nursing students
Justice (Arrest)	Zawahiri was arrested in Iran

Table 3. Examples of the ACE Event Types

Entity Translation

Entity Translation is a cross-lingual IE track at ACE 2007 to take in a document in a foreign language (e.g. Chinese or Arabic) and extract the English catalog of the entities.

¹ The ACE task description can be found at <http://www.nist.gov/speech/tests/ace/> and the ACE guidelines at <http://www ldc.upenn.edu/Projects/ACE/>.

SCIENTIFIC FUNDAMENTALS

There are two main approaches to develop IE systems, described separately as follows.

Pattern Matching based IE

Many IE systems during MUC evaluation use high-accuracy rules, dictionaries and patterns for each specific domain. For example, for the end-position event in Table 1, an IE system generates patterns such as

- [Person] quit as [Position] of [Organization]

Manually writing and editing patterns require some skill and considerable time. So some systems have moved on to learning these patterns automatically based on an annotated corpus pre-processed by syntactic and semantic analyzers. A more comprehensive survey of pattern matching based IE approaches can be found in [3].

The above pattern acquisition is still quite costly because for particular domain a separate annotated corpus is needed. Therefore some systems have used unsupervised learning approach [4, 5, 6]. The general idea is to obtain a pattern if a pair of arguments (mostly names) (Arg_1 , Arg_2) and their context C_{12} appear frequently in other instances of the event.

The idea of using bootstrapping to obtain patterns was first proposed by Riloff in [4]. [4] manually pre-classified the documents into relevant and irrelevant, then collect and score patterns around each noun phrase. In [5] Yangarber et al. used seed patterns to address the limitation of manual document classification. They started with a few initial seed patterns, and then applied an incremental discovery procedure to identify new set of patterns. Both of [4, 5] are based on predicate-argument or subject-verb-object structures. [6] presented a new Subtree model based on dependency parsing, and proved the Subtree model can obtain higher recall while preserve high precision.

Machine Learning based IE

The IE systems relying entirely on pattern matching have attempted some success in MUC domains. However these patterns cannot be easily adapted into new domains. Therefore, IE research has grown by splitting the task into several components and then applying machine learning methods to address each component separately.

Machine learning based IE systems typically include name identification and classification, parsing (or partial parsing), semantic classification of nominal mentions, coreference resolution, relation extraction and event extraction. A typical IE system pipeline is presented in Figure 1. For instance, state-of-the-art IE systems such as BBN system [7], IBM system [8] and NYU system [9] were developed in this pipeline style. This 'pipeline' design provides great opportunity to applying a wide range of learning models and incorporating diverse levels of linguistic features to improve each component. Large progress has been achieved on some of these components. In the following some typical learning methods are described for the important components.

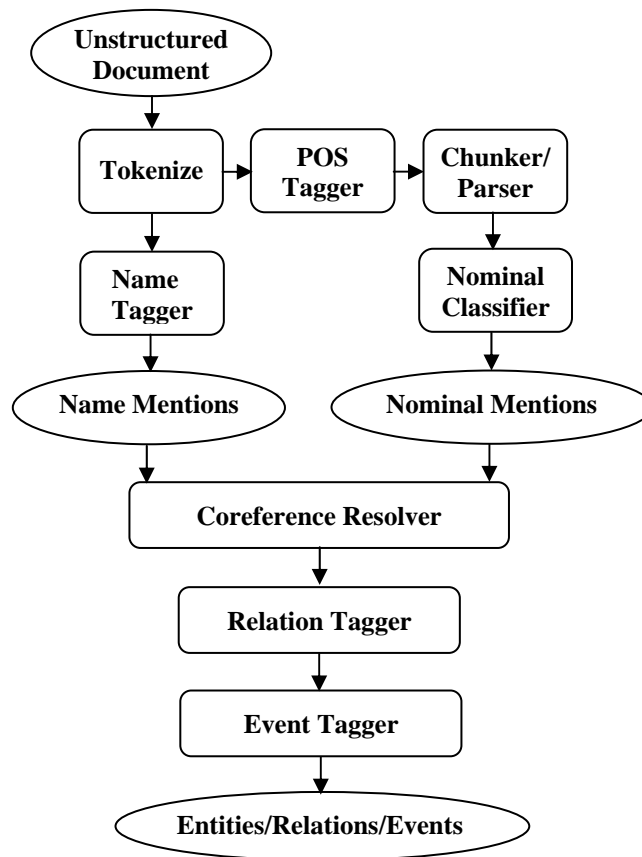


Figure 1. A Minimal Machine Learning based IE System Pipeline

Trainable Name Tagging

The problem of name recognition and classification has been intensively studied since 1995, when it was introduced as part of the MUC-6 Evaluation. A wide variety of unified learning algorithms have been applied to the name tagging task, including Hidden Markov Models (HMMs), Maximum Entropy Models, Decision Trees, Conditional Random Fields and Support Vector Machines.

The most well-known BBN's Nymble name tagger [10] used several methods to improve performance over a simple HMM. Within each of the name class states, a statistical bigram model is employed, with the usual one-word-per-state emission. The various probabilities involve word co-occurrence, word features, and class probabilities. Since these probabilities are estimated based on observations seen in a corpus, several levels of “back-off models” are used to reflect the strength of support for a given statistic, including a back-off from words to word features.

Trainable Coreference Resolution

Coreference Resolution is the task of determining whether two mentions refer to the same entity. For example in the sentence in Table 1, the name mention “Barry Diller” and the nominal mention “media tycoon” refer to the same person entity.

In a corpus-trained system, coreference resolution is usually converted into a supervised binary classification problem of determining whether a candidate mention is referring to an antecedent or not. Here an ‘antecedent’ can be another single mention, or a cluster of mentions which the system has generated. Each pair is assigned probability value by a supervised learning based classifier. If the sampling is constructed on each mention pair, then a separate clustering algorithm is applied to group coreferring mentions.

Most coreference resolution systems use representations built out of the lexical and syntactic attributes of the mentions for which reference is to be established [11]. A typical feature set includes:

- representing agreement of various kinds between mentions (number, gender)
- degree of string similarity
- synonymy between mention heads
- measures of distance between mentions (such as the Hobbs distance)
- the presence or absence of determiners or quantifiers

Though gains have been made with such methods, there are clearly cases where this sort of local information will not be sufficient to resolve coreference correctly. Coreference is by definition a semantic relationship, therefore a successful coreference system should exploit world knowledge, inference, and other forms of semantic relations in order to resolve hard cases. Since 2005 researchers have returned to the once-popular semantic-knowledge-rich approach, investigating a variety of semantic knowledge sources. For example, [12] incorporated the feedback from semantic relation detection to infer and correct coreference analysis. If, for example, two library mentions which are located in two different cities, then these mentions are less likely to corefer.

Trainable Relation Detection

For ACE-type relations, various machine learning methods have been used such as K-Nearest-Neighbor [9] and Support Vector Machines [13]. The typical features used to classify relations include:

- the heads of the mentions and their context words
- entity and mention type of the heads of the mentions
- the sequence of the heads of the constituents, chunks between the two mentions
- the syntactic relation path between the two mentions
- dependent words of the mentions

Trainable Event Detection

A typical event extraction pipeline includes three main steps:

- **Trigger Identification**

Identify the trigger word in a given sentence and assign event type using the probability computed from the training corpora.

- **Argument Identification**

For a given trigger and a mention, determine whether the mention is an argument of the trigger or not.

- **Argument Classification**

For an identified argument, classify the argument as a specific event role.

Event detection heavily relies high-quality deep parsing. [7, 9] have further shown that the predicate-argument structures can provide deeper linguistic analysis and therefore effectively enhance the performance of event detection.

KEY APPLICATIONS

An enormous amount of information is now available through the Web; much of this information is encoded in natural language, which makes it accessible to some people (those who can read the particular language), but much less amenable to computer processing (beyond simple keyword search). If we can enable a computer to extract and utilize the knowledge embedded in these texts, we will have unleashed a powerful knowledge resource for many fields. Some typical applications of IE are presented as follows.

- IE for Daily News

IE can also be applied to identify the events in the daily news articles. If an informative database can be returned based on the facts extracted by IE from multiple sources of news, it can be a very valuable result and save the time a user has to spend in browsing. For example, for the news articles about Olympic sport games, an IE engine can automatically provide a table of the player's person names, the team names they come from and the game results.

- IE for Financial Reports

Every year the U.S. government releases the annual reports from millions of industrial agencies. The financial analysis companies then gather all these reports and analyze the most up-to-date information such as the company start-up and merge events, the competition and cooperation relations among banks or companies. It will be very helpful if an automatic IE system is applied to compress these articles into data bases first. Recently such IE systems are widely applied in the financial domain to assist human analysts.

- IE for Biology Literatures

In the biology domain, thousands of new papers and data sets are published in natural language on a daily basis. It has become impractical for scientists to manually track all these new results and observations, and manually mine the data sets to construct a knowledge base. IE can play a significant role by automatically generating an accurate summary of facts (e.g. gene named entities) and predicting new results (e.g. Bio-nano structures of different peptide sequences), and thus assist scientists in decision making.

- IE for Medical Reports

Since the early work by Sager et al. [1], IE has obtained successful applications in processing the narrative clinical documents including patient discharge summaries and radiology reports. Some of these systems have shown positive impact on providing information to assist clinical decision, result analysis, error detection, etc.

FUTURE DIRECTIONS

For each IE component there are different aspects to improve. This section proposes some high-level directions in which IE can be further explored.

- Cross-document Information Extraction

One of the initial goals for IE was to create a database of relations and events from the entire input corpus, and allow further logical reasoning on the database. The artificial constraint that extraction should be done independently for each document was introduced in part to simplify the task and its evaluation.

However, almost all the current event extraction systems focus on processing single documents and, except for coreference resolution, operate a sentence at a time. Therefore, one interesting area worth exploring would be to gather together IE results from a set of related documents, and then apply inference and constraints to propagate correct results and fix the wrong information generated from the within-document IE system.

- IE for Noisy Input

Recently there has been rapid progress in applying text processing techniques on ‘noisy’ texts such as the output of automatic speech recognition (ASR) and machine translation (MT). The potential ASR transcription and machine translation errors, in particular name recognition errors, make IE more difficult. However, it’s possible to optimize the parameters in the ASR or MT systems for IE purpose. Another interesting direction would be using IE results to provide feedback to ASR and MT in a joint inference framework.

- Cross-lingual IE

A shrinking fraction of the world’s web pages are written in a language different from the user’s own, and so the ability to access information from foreign languages is becoming increasingly important. This need can be addressed in part by the research on cross-lingual IE (CLIE).

- Active Learning for Domain Adaptation

Since about one decade ago in MUC program, the ‘portability’ problem has become a noticeable bottleneck for IE techniques. Until today this problem has not been solved yet. There is an urgent need to develop effective adaptation algorithms to apply IE systems to a new domain with low cost. Active learning and semi-supervised learning techniques, which have achieved success on name tagging, may be worth expanding to all stages in the IE pipeline.

EXPERIMENTAL RESULTS

The state-of-the-art IE results can refer to the ACE evaluation results on NIST website². All IE results are given in terms of the entity/relation/event value scores, as produced by the official ACE scorer. These value scores include weighted penalties for missing extractions, spurious extractions, and for type errors in corresponding extractions³. The top systems obtained mention values in the range of 70-85, entity values in the range of 60-70, relation values in the range of 35-45, event values in the range of 15-30.

DATA SETS

- ACE IE: <http://projects.ldc.upenn.edu/ace/data/>
IE training data for English/Chinese/Arabic/Spanish
- CONLL 2002: <http://www.cnts.ua.ac.be/conll2002/ner.tgz>
Name tagging training data for Dutch and Spanish
- CONLL 2003: <http://www.cnts.ua.ac.be/conll2003/ner.tgz>
Name tagging training data for English and German

URL to CODE* (optional)

- UIMA: <http://incubator.apache.org/uima/svn.html>
IBM NLP platform
- Jet: <http://www.cs.nyu.edu/cs/faculty/grishman/jet/license.html>
NYU IE toolkit
- Gate: <http://gate.ac.uk/download/index.html>
University of Sheffield IE toolkit
- Mallet: http://mallet.cs.umass.edu/index.php/Main_Page
University of Massachusetts NLP toolkit
- MinorThird: <http://minorthird.sourceforge.net/>
Carnegie Mellon University NLP toolkit

² <http://www.itl.nist.gov/iad/894.01/tests/ace/>.

³ Scoring details can be found in the ACE07 evaluation plan: <http://www.nist.gov/speech/tests/ace/ace07/doc/ace07-evalplan.v1.3a.pdf>

CROSS REFERENCES

Text summarization
Text indexing & retrieval
Topic detection and tracking
Cross-language mining and retrieval
Structured and semi-structured document databases

RECOMMENDED READING

- [1] Naomi Sager. 1981. *Natural Language Information Processing: A Computer Grammar of English and its applications*. Reading, Massachusetts: Addison Wesley.
- [2] Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference – 6: A brief history. *Proc. of the 16th International Conference on Computational Linguistics (COLING 96)*.
- [3] Ion Muslea. 1999. Extraction Patterns for Information Extraction Tasks: A Survey. *Proc. of the National Conference on Artificial Intelligence (AAAI-99) Workshop on Machine Learning for Information Extraction*.
- [4] Ellen Riloff. 1996. Automatically Generating Extraction Patterns from Untagged Text. *Proc. of AAAI-96*, pp. 1044-1049.
- [5] Roman Yangarber; Ralph Grishman; Pasi Tapanainen; Silja Huttunen. 2000. Automatic Acquisition of Domain Knowledge for Information Extraction. *Proc. of the COLING 2000*.
- [6] Kiyoshi Sudo, Satoshi Sekine and Ralph Grishman. 2003. An Improved Extraction Pattern Representation Model for Automatic IE Pattern Acquisition. *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL 2003)*.
- [7] Elizabeth Boschee, Ralph Weischedel and Alex Zamanian. 2005. Automatic Evidence Extraction. *Proc. of the International Conference on Intelligence Analysis*.
- [8] Radu Florian, Hongyan Jing, Nanda Kambhatla and Imed Zitouni. 2006. Factorizing Complex Models: A Case Study in Mention Detection. *Proc. of the COLING-ACL 2006*, pp. 473-480.
- [9] Ralph Grishman, David Westbrook and Adam Meyers. 2005. NYU's English ACE 2005 System Description. *Proc. of the ACE 2005 Evaluation/PI Workshop*.
- [10] Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: a high-performance Learning Name-finder. *Proc. of the Fifth Conf. on Applied Natural Language Processing*. pp.194-201.
- [11] Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. *Proc. of the ACL 2002*, pp.104-111.
- [12] Heng Ji, David Westbrook and Ralph Grishman. 2005. Using Semantic Relations to Refine Coreference Decisions. *Proc. of the HLT/EMNLP2005*. pp. 17-24.
- [13] Guodong Zhou, Jian Su, Jie Zhang and Min Zhang. Exploring Various Knowledge in Relation Extraction. *Proc of the ACL 2005*. pp. 427-434.