Short on Information Extraction

Behrang QasemiZadeh
Heinrich-Heine Universität Düsseldorf

April, 2016

Information extraction (IE) proposes a pragmatic approach to text understanding. It is the process of distilling structured data (i.e., factual information) from unstructured or semi-structured text. IE is thus often application-oriented and generates outputs typically in the form of database templates. IE has applications in a wide range of domains and has been extensively studied in various research communities (McCallum, 2005). As a result, it covers a variety of tasks such as *entity extraction*, *relation extraction*, and *events detection*—or put simply information about *who* did *what* to *whom*, *when* and *where* (Hobbs and Riloff, 2010). In contrast to *information retrieval*, which is about identify documents relevant to a query from a document collection, IE produces structured data ready for some post-processing.

Many advances in IE is often credited to the DARPA-funded Message Understanding Conferences (MUC) (Grishman and Sundheim, 1996). MUC concentrated on IE by evaluating the performance of participating IE systems using a black box test: the MUC evaluation style (Hirschman, 1998). The task in MUC focused on extracting information from news about terrorist events, industrial joint ventures, and company management changes. The proposed approaches started off with rule-based methods and gradually moved to machine learning methods. The MUC was followed by the Automatic Content Extraction (ACE) evaluations (Doddington et al., 2004). The ACE evaluations focused on identifying named entities, extracting isolated relations, and coreference resolution. The tradition of evaluating IE systems in the MUC style has continued up until today through various events such as Text Analytic Conference.

The diverse research in IE can be classified based on the following features Hobbs and Riloff (2010): (a) Type of Input to IE System, (b) applied method for extraction of information, and (c) the extraction target. Input types to an IE system can be categorized into unstructured versus semi-structured text input, and single-document versus multi-document input. Examples of unstructured text include news stories, magazine articles, and books. Semi-structured text consists of natural language text that appears in a document where the physical layout of the text plays a role in its interpretation. Examples of semi-structured input are emails, job posts, ads, and resumes. The unstructured IE systems mostly rely on language analyses while IE from semi-structure

content rely also on positional features to capture information from the layout of input.

IE systems originally were designed to extract domain specific information from single documents. In this scenario, given a document as input, an IE system extracts facts and domain informations articulated in this document. Boosted by the application of IE over the Web, recent multi-document IE systems try to extract facts from multiple sources (Eikvil, 1999). The major difference in problem formulation for single-document IE and multi-document IE is *information redundancy*. In single-document IE, IE systems must extract specific information from each document as a fact may have been mentioned only once in one document; whereas, based on the assumption that many facts will be reported multiple times in different sources and in different forms (Ji, 2010), a multi document IE usually has several opportunities to find each piece of information. However, it is worthwhile mentioning that consolidating information from multiple sources can be a challenging research too.

IE systems can be also classified based on the method they employ for performing the extraction task. Generally speaking, IE methods can be classified into hand-crafted and learning-based approaches (i.e., by the degree of automation in the development of the method). Slightly different, one can also categorize these methods into rule-based and statistical-based methods (Sarawagi, 2008) (note that rules can be crafted manually or be devised automatically). Well known examples of rule based approaches are IE systems that employ finite state transducers, i.e., often based on hand-crafted regular expression patterns. Learning based approaches use statistical techniques and machine learning algorithms to automatically create IE systems for new domains or tasks. From this perspective IE systems can be further classified into four categories based on the automation degree in the development process (Chang et al., 2003): (1) systems that need programmers, (2) systems that need annotation examples, (3) annotation-free (unsupervised) systems, and (4) semi supervised systems. Recent methods in IE combine a loose mixture of text extraction and data mining. Categories (3,4) are mainly refer to methods that leverage whatever limited structured information is available and then use data mining techniques that are robust enough to operate directly on the raw text associated with this limited structure (McCallum, 2005).

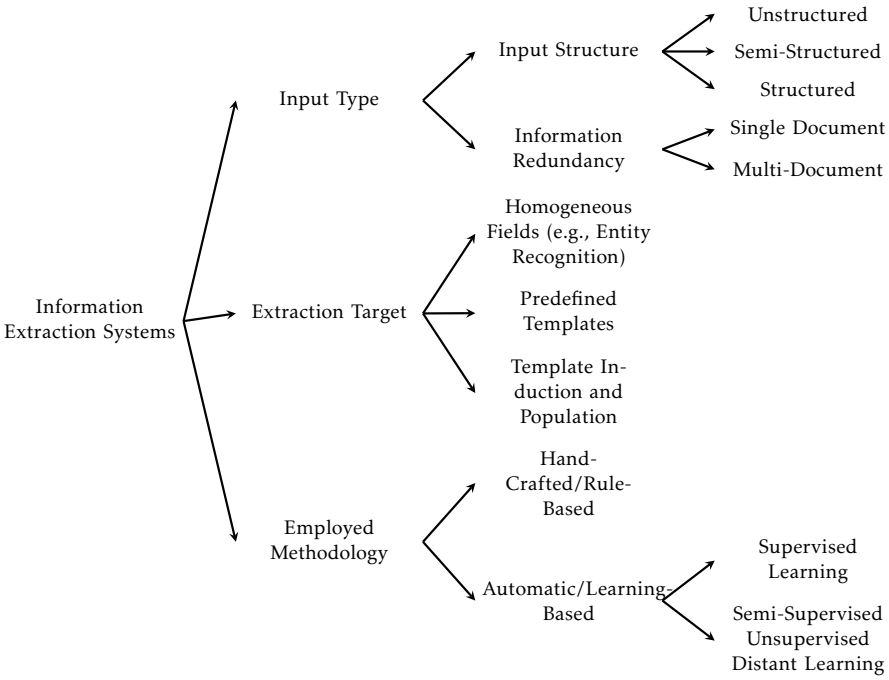

Figure 1: A mind map and categorisation of information extraction systems and methods.

Finally, IE methods can be classified by the type of their output (Chang et al., 2006). This categorization takes into the consideration the structure of outputs by IE systems (i.e., the complexity of templates that must be populated by an IE system). The extraction target can be in a range of isolated phrases to complex m-tuples. In its simplest form, an IE system populates a list of homogeneous items, e.g., a list of the names of companies, such as extracted by named entity recognition systems. In the m-tuple extraction task, where m is the number of attributes in a record, the output has a more complex structure (e.g., the name of companies and their CEOs). Lastly, a more challenging IE task is to automatically induce the extraction templates and then populate them (Chambers and Jurafsky, 2011). Figure 1 summarizes the mentioned categorizations for IE systems.

Bibliography

Chambers, N. and Jurafsky, D. (2011). Template-based information extraction without the templates. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 976–986, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chang, C.-H., Hsu, C.-N., and Lui, S.-C. (2003). Automatic information extraction from semi-structured web pages by pattern discovery. *Decis. Support Syst.*, 35(1):129–147.

Chang, C.-H., Kayed, M., Girgis, M. R., and Shaalan, K. F. (2006). A survey of web information extraction systems. *IEEE Trans. on Knowl. and Data Eng.*, 18(10):1411–1428.

Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., and Weischedel, R. (2004). The automatic content extraction (ace) program–tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pages 837–840.

Eikvil, L. (1999). Information extraction from world wide web - a survey. Technical report, Norwegian Computing Center.

Grishman, R. and Sundheim, B. (1996). Message understanding conference-6: A brief history. In *In Proceedings of COLING (Vol. 96).*, pages 466–471.

Hirschman, L. (1998). The evolution of evaluation: Lessons from the message understanding conferences. *Computer Speech & Language*, 12.4:281–305.

Hobbs, J. R. and Riloff, E. (2010). Information extraction. In Indurkhya, N. and Damerau, F. J., editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, FL. ISBN 978-1420085921.

Ji, H. (2010). Challenges from information extraction to information fusion. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 507–515, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Sarawagi, S. (2008). Information extraction. *Found. Trends databases*, 1(3):261–377.