Datenbank Spektrum (2012) 12:81–88
DOI 10.1007/s13222-012-0090-x

SCHWERPUNKTBEITRAG

Introduction to Information Extraction: Basic Notions and Current Trends

Wolf-Tilo Balke

Received: 6 May 2012 / Accepted: 8 May 2012 / Published online: 19 May 2012

Abstract Transforming unstructured or semi-structured information into structured knowledge is one of the big challenges of today's knowledge society. While this abstract goal is still unreached and probably unreachable, intelligent information extraction techniques are considered key ingredients on the way to generating and representing knowledge for a wide variety of applications. This is especially true for the current efforts to turn the World Wide Web being the world's largest collection of information into the world's largest knowledge base. This introduction gives a broad overview about the major topics and current trends in information extraction.

Keywords Information extraction

1 Introduction

With the advent of the World Wide Web as an almost unlimited information source, information extraction (IE) has become one of the most active research areas in database and information system research [1, 2]. Whereas research and development in databases mostly used to be focused on the scalability of data management and retrieval technology, the actual process of creating high-quality databases was increasingly explored during the last ten years. Besides the plentitude of new research challenges for academia, the demand of many industrial applications to intelligently incorporate knowledge into tasks instead of simply processing clear-cut data can be considered as a major driver for this change. Hence, also topics like data provenance or lineage [3] and data quality [4] are heavily discussed today.

The basic task of information extraction is to automatically extract structured information from unstructured and/or semi-structured machine-readable documents. That means data which can be detected in one or more texts should be classified, transformed, and stored for further use usually into some database. Starting with simple indexing tasks in business intelligence applications, the extraction of all kind of knowledge from the Web as the world's largest database has become the holy grail of information extraction. Generally, this task is easier if the information is explicitly stated, but advanced IE techniques should also work for implied information.

Traditionally, information extraction has relied on a large amount of human involvement that can be traced back to manually curated databases for specific sciences like CAS's database of chemical substances and chemical documents (CAS Registry and SciFinder, http://www.cas.org/) or the United States National Library of Medicine's document collection using MeSH annotations (MEDLINE/PubMed, http://www.ncbi.nlm.nih.gov/pubmed/). Here also all interesting relations between entities have been provided by domain experts. With the Web-scale extension of information extraction, first interactive extraction algorithms have been designed. For instance, some frameworks use rules in the form of regular expressions (e.g., for phone numbers or dates) or learn where to find certain entities or a specific type of information on specific Web sites directly from users and are then able to generate adequate wrappers for subsequent extraction [5–7].

However, in order to cater for a large set of possible queries from *all* kinds of users, IE methods have to move from requiring entities and relations to be specified prior to query time towards discovering all sensible entities and their

W.-T. Balke (✉)
Institut für Informationssysteme, Technische Universität Braunschweig, Braunschweig, Germany
e-mail: balke@ifis.cs.tu-bs.de
url: http://www.ifis.cs.tu-bs.de

possible relations to each other. But what defines a reasonable relation or an entity that some user might be interested to query for? Although there is no perfect answer (in fact most noun phrases can be considered entities in some respect), an interesting observation can be made: a steadily growing number of real-world entities whether they are persons, locations, animals, chemical substances, or artefacts is present in manually curated Web portals or encyclopedias like, e.g., Wikipedia. Moreover, these entities can be considered to be agreed upon by a larger number of users.

Indeed, one of the early approaches of building a Web-scale knowledge base was actually DBPedia (http://dbpedia.org), which tries to extract structured information directly from Wikipedia and allows for linking other data sets on the Web to Wikipedia data [8]. Taking the value of curated collections for algorithm training into account, many recent IE approaches start from curated knowledge and try to extend the observed structures to other Web sources to find also instances that are not (yet) present in the original source.

In the following, we will give a short introduction to research on information extraction and point out basic notions, current trends, and future challenges. Please note that most of the extraction techniques described are based on thorough natural language processing (NLP) of the documents or texts from which the information needs to be extracted (for a comprehensive introduction see [9]). There is a wide variety of NLP techniques that allow to dissect texts and to look at the grammatical roles of all words and phrases. The most important grammatical analysis is referred to as part-of-speech (POS) tagging and allows making first distinctions, e.g., entities are usually given by nouns or noun phrases, attributes are often expressed in adjectives and relationships often by verbs or verb phrases. To better understand a sentence's actual meaning, the semantic role labeling or shallow semantic parsing detects the semantic arguments associated with a sentence's verb and their classification into specific roles [10]. Finally, for deeper semantic insights into a sentence's structure, lexical dependency sometimes parsers like RelEx (http://wiki.opencog.org/w/RelEx) are used.

2 What Can Be Extracted?

2.1 Named Entities

Already at an early stage in developing information retrieval systems, the use of finding specific terms in texts as first step for automatic text understanding tasks became clear. Named entity recognition (NER) or entity identification aims at finding real-world objects in texts and classifying these objects into predefined categories such as names of persons, organizations, locations, temporal expressions, products, etc. Specifically for business intelligence and special business applications, e.g., involving the stock market, also quantities and monetary values like prices or percentages are often considered. The usefulness of these simple extractions in the following lead to a series of DARPA-sponsored conferences on the topic of message understanding (MUC, 1987–1997 [11]), where similar to the TREC conferences different systems competed in correctly recognizing named entities.

Generally speaking, named entity recognizing software builds on a variety of extractors for all interesting items. For example, geographic names are often simply extracted using gazetteers, i.e. a geographical dictionary or directory. A prime example of such a gazetteer is the GEOnet Names Server provided by the National Geospatial-Intelligence Agency and the U.S. Board on Geographic Names (http://earth-info.nga.mil/gns/html/). Such dictionaries also exist for other non-geographic kinds of entities like common names or currencies. These basic operators already allow for extracting the most common items with very good precision. To get better recall advanced extractors heavily rely on statistical models that look at the local surrounding of extraction candidates, i.e., how entities are embedded into sentences. These models can then be extended to also consider occurrences in previous sentences. Typical methods for the probabilistic sequence model NER are hidden Markov models, maximum entropy Markov models, or conditional random fields (see e.g., [12–15]).

With advanced extractors, specific named entities such as persons, products, and organizations can be extracted with high reliability in the 90th percentile range. Thus, the initial NER problem from documents is often considered solved. A commonly used implementation in many Web extraction applications is provided by business data provider Thomson Reuters: the OpenCalais Web Service framework developed by text analytics company ClearForest (http://www.opencalais.com/). Based on semantic Web service technology, unstructured text can be submitted; it undergoes several extractors and OpenCalais returns all results identifying entities, facts, and events within the text. The set of recognized entities ranges from persons, organizations, and locations via structural content like email addresses, medical conditions, or phone numbers, to events like anniversaries, product recalls, or bankruptcies.

In summary, it has to be said that the basic problem of what makes an entity which might be of interest to some users is still remaining. However, once some entity has become important for some reason, extractors can usually be designed very quickly and effectively. This allows for simple querying and document retrieval, e.g., all documents mentioning a certain person, news items about a certain event, or pieces about a certain stock being sold or bought in high quantities.

2.2 General Entities

The problem of named entity recognition generally depends on the domain of the documents from which the information is extracted. But for general information sources like the Web, the question of what named entities to look for had to be extended: how to extract domain-independent entities with highest possible precision, but acceptable recall?

Based on generic language patterns inspired by [16] the KNOW-IT ALL system in [17] used a set of eight simple and domain-independent extraction patterns to generate candidates for extraction. Testing these candidates for plausibility using pointwise mutual information statistics computed over a Web corpus allowed for good precision, while pattern learning techniques allowed to learn domain-specific extraction rules and thus improved recall. Subsequent work in the TextRunner or OpenIE framework [18] made the extraction of general entities more scalable and dealt with the heterogeneity problems of Web corpora, which caused severe challenges on linguistic methods for extraction. Using a self-supervised learning approach on a small corpus sample allows training a classifier that then can subsequently decide the trustworthiness of candidates extracted by simple part-of-speech methods.

A special problem arising when extracting entities from text is entity disambiguation, i.e., the problem of whether two extracted entity occurrences sharing the same name really refer to the same entity. For example (amongst other possible entities), there are two American presidents the named entity 'George Bush' might refer to. Word sense disambiguation for homonyms of course is always a difficult problem in Web search and information retrieval and can only be resolved by looking at the context of the words in question, e.g., surrounding text, type and date of the document, or the kind of document collection. For entities, one possible way is a disambiguation by relating the documents an entity was extracted from to relevant information known about some specific entities. For a wide range of popular entities, such relevant information or typical contexts (like being a president or living in the White House) can be provided by suitable encyclopedias [19, 20], knowledge bases [21], or existing ontologies [22]. Moreover, to some degree also the structure of documents, in particular the links between Web documents, can be exploited [23].

Finally, another problem for entity extraction is term evolution. The problem here is that a term used for some entity may change over time. As the often cited example of the city of St. Petersburg (which was called Leningrad and Petrograd at different points in time) shows this is true especially for long-lived entities like locations, but can also happen to persons (especially artists) or even abstract entities. Similar to the case of entity extraction, a pattern-based approach [24] can help when looking at corpora from different time intervals: entities co-occurring in lists (connected by ',', 'and', or 'or') can be expected to share a semantic context and taken from documents of different time periods can be candidates for term evolution.

2.3 Characteristics and Attributes of Entities

Besides the extraction of the actual entity, it is often also important to harvest characteristics or attribute values for specific entities in the sense of structured data from unstructured sources. A major application for gathering entity-centered characteristics is entity-centered Web search, often referred to as object-level vertical search (as opposed to the traditional page-level search). The basic idea is automatically building highly complete profiles for entities. Since knowing what characteristics usually determines an entity type is extremely helpful in gathering the respective information, this kind of extraction is generally used only for named entities like persons, locations, organizations, or products. More general types of relationships will be considered in a later section.

There is quite a number of projects or prototypes especially in the area of Web search providers which naturally have a high interested also in specialized search tasks like, e.g., person search or product search. A good example for the focused extraction of entity characteristics is Microsoft's EntityCube [25, 26] which automatically summarizes information about entities even if they only have a modest Web presence. For example, biography pages of persons can to some degree be automatically generated and using the social-network graph for some person also relationship paths between people can be discovered.

Having found characteristics for some entity, the above mentioned problem of entity disambiguation can be seen from a different angle: reference reconciliation or record linkage is the problem of identifying when different references (i.e., sets of attribute values) in a dataset correspond to the same entity. Also for this topic, several methods [27–29] have been proposed for measuring the distance between entities given sets of attribute values or calculating a probability or plausibility for two sets of characteristics describing the same entity. Since a high-quality solution to this problem is especially useful for the demanding problem of schema matching in databases, a lot of effort is currently invested into developing and testing novel solutions.

2.4 Classes of Entities

Not only do entities show different characteristics, they may also be categorized in classes with respect to several concepts. For instance, a person might be classified by gender, profession, nationality, lifetime, and many more. Only in this way, abstract queries like retrieving all female scientists of the 19th century or all German Nobel Prize winners can

be answered. Hence, a further challenge when dealing with entity extraction is classifying all extracted entities with respect to suitable concepts. Again similar to the case of extracting entities, it has to be considered an unsolved problem how to anticipate at extraction time which classes will be of interest to users in future queries.

However, some typical classes for often-used concepts can and should be extracted. For answering queries using entity classes, ontologies have since long been used for knowledge representation. Such representations started with simple hierarchical descriptions in the form of taxonomies ('is-a' or 'subclass-of') or mereologies ('has-a' or 'part-of'), sometimes even with crosslinks like in MeSH (http://www.nlm.nih.gov/mesh), CheBI (http://www.ebi.ac.uk/chebi), or WordNet (http://wordnet.princeton.edu). The use of full-fledged ontologies containing all kinds of relationships is still unusual in practical applications due to their high complexity in terms of generation, maintenance, and subsequent use. The main problem when employing such ontologies, e.g., for reasoning tasks in the Semantic Web, is that their use is often inefficient and the usefulness directly dependent on their quality and interoperability.

Today, almost all large-scale ontologies in use are manually curated and maintained, which of course is an expensive task. To automatically derive ontologies for classification relationships between salient terms in documents have to be extracted and suitably represented. The earliest, yet still quite effective techniques directly rely on specific patterns (called lexicosyntactic patterns) in the text [30–33]. For example '... an X such as Y ...' or '... all X, including Y ...' define simple patterns to express a 'is-a' hierarchy between concepts X and Y. Once found in some text, the respective concepts and their relationships can be harvested and stored for further use.

While such purely pattern-based methods suffer from the sparseness of occurrences for all possible classifications even in such large corpora as the Web, probabilistic models or statistical models try to boost the number of possible classifications extracted. The PANKOW (Pattern-based Annotation through Knowledge on the Web) [34] method combines the idea of the above-mentioned linguistic patterns with collecting evidence for a connection between candidate concepts in an unsupervised fashion. Purely statistical methods even give up on the language patterns for candidate generation relying on so-called shallow semantics only. The most often-used statistics is co-occurrence of terms [35, 36] that basically state that some terms have a common background. By interpreting the way in which terms co-occur to some degree, also the nature of their connection can be derived, if only the underlying corpus is big and heterogeneous enough. During the last years, research has also heavily focused on so-called folksonomies [37], which derive shallow semantics directly from user-generated metadata (e.g., tags) usually in the form of a concept cloud. Folksonomies are considered light-weight ontologies forming a useful compromise between real classification knowledge in the ontological sense and unstructured metadata.

The major problems of automatically extracted concepts for classification are, on the one hand, measuring the extracted ontology's quality and, on the other hand, the integration of ontologies derived from different corpora or in different domains (in the sense of creating a common upper ontology). Here only few and hardly convincing solutions exist. The measuring of quality for ontology extraction methods—besides manual inspection—generally needs a comparison of all derived classifications against some recognized standard ontology (the gold standard or ground truth). For the integration of ontologies, some techniques from the heavily related area of schema matching or data integration in databases [38] have been adapted. Advanced techniques also consider the loss of focus when moving from some ontology to another [39–41].

2.5 General Relationships Between Entities

While classes of entities already define simple relationships (like 'is-a' or 'part-of'), general relationships between entities can be arbitrary complex like, for instance, a person 'married_to' another person, a person 'born_in' a city, a company 'producing' a product, or a drug 'inhibiting' some illness. In contrast to the entity extraction where the natural language processing usually focuses on the noun phrases, for relationships verb phrases become important as can be seen in the examples above. Since relationships are stated with respect to entities, a powerful entity extraction forms the base of all relationship extraction tools. When dealing with relationships, there are generally three problems that need to be considered:

- The technical challenge: how to detect general relationships in heterogeneous and mostly unstructured data?
- The usability challenge: which relationships are valid and valuable for future querying or reasoning?
- The semantic challenge: what does a relationship really describe and is this inherently transparent for future users?

General answers to these questions are difficult to provide. While for the technical challenge to some degree the POS tagging, language patterns, and machine learning algorithms used for entity extraction can be readily adapted, the other two challenges are somewhat harder than the entity extraction case.

Consider the usability challenge: relationships indeed range from most general and obvious connections to explicit and valuable knowledge. If, for instance, relationships between a certain person and cities (both clear-cut types of

entities) should be extracted, general relationships could be: a person 'was_born_in' a city, a person 'is_living_in' a city, a person 'having_visited' a city, a person 'knowing' a city, a person 'having_heard_of' a city, etc. On one hand, the variety of possible relationships leads to an explosion of knowledge base sizes, on the other hand, the desirability of the contained information seems to be vastly different.

Moreover, considering the semantic challenge it is easy to see that there is a vast variety of ways to express the same relationship information (called paraphrasing): a person 'was_born_in' a city, a city 'is_the_birthplace' of a person, etc. This flexibility of natural language often makes it hard to reliably extract relationships. Moreover, subtleties in language might involve a certain semantic loss in paraphrasing like a person 'being_familiar' with a task, 'knowing' a task, or 'being_an_expert' in a task. But there are more semantic problems than the semantic reconciliation of paraphrasing: unlike someone's place of birth, many relationships are only true with respect to a certain context. Considering, for instance, time as a context is necessary for relationships like a person 'is_living_in' a city or a person 'is_president_of' a country. Such relationships often change over time; still they may be reflected by different sources.

In the following, we will take a closer look at current work in relationship extraction. All systems of course deal with the technical challenge and to some degree try to deal with the usability and semantic challenge. In particular, this is successful in terms of relationship disambiguation and reconciliation and, by simple heuristics like exploiting Wikipedia infoboxes, also for the suitable choice of interesting relationships. Harder challenges like context-based subtleties between relationships, issues of validity contexts, or finding all typical relationships, are however still active areas of research.

Following up the entity extraction of TextRunner, the ReVerb framework [42] improves relationship extraction by avoiding incoherent or meaningless relations and uninformative statements. This is done enforcing certain syntactic and lexical constraints like focusing on specific POS patterns in multi-word relations or omitting all relationships that rarely occur across a corpus. The Yago-Naga [43] framework relies on Wikipedia infoboxes and category names that define suitable relationships for extraction and reconciles them with the taxonomic backbone of WordNet. This ensures a highly consistent class system. Using a set of well-defined rules like "If person X has spouse Y, then X 'married_to' Y" then allows deriving a large number of facts stored in the YAGO knowledge base, which can be queried using the NAGA engine. A similar attempt, but using machine learning technology to capture even more Wikipedia relationships, was made by the Kylin/KOG framework [44].

Both frameworks, Yago-Naga and Kylin/KOG, maintain a hierarchy of classes or ontology to give facts a clean taxonomic structure. Based on the instances in such high-precision knowledge bases, it becomes possible to build even better patterns for extraction. In terms of simple lexical language patterns, the co-occurrence of entities from known facts in some text can be used to generate candidates for new extraction patterns. These new patterns can then be confirmed by applying the patterns to other entity pairs of the same fact type and searching for respective instances in the text corpus [45]. The StatSnowball [46] approach builds on the Snowball system [47] and tries to learn syntactic patterns consisting of POS tags by optimizing the specificity and coverage trade-off. For finding such patterns, discriminative Markov logic networks (MLNs) are employed allowing for the combination of logic rules and probabilistic models. The SOFIE [48] methodology combines new pattern generation with consistency reasoning over the YAGO ontology. New facts are only admitted to the knowledge base if the statistical evidence from new facts, known facts, patterns, and applicable constraints is sufficiently high. Finally, the PROSPERA knowledge harvester [49] aims at making the combination of improving recall by pattern-based harvesting while controlling precision with constraint-based reasoning in SOFIE scalable. This is, on one hand, done by improving pattern representation and rule weighting and, on the other hand, by allowing for a large degree of parallelization distributing subtasks to Map-Reduce frameworks during the actual harvesting.

Similar to entity reconciliation in entity extraction tasks, the problem of relationship paraphrasing has received a lot of attention recently. Learning paraphrases requires to ensure the identity of meaning for certain phrases. But semantic interpretation is still a problem for machines due to the flexibility and richness of natural language as well as the necessary feeling for subtleties in language. Early pattern-based attempts simply considered all phrases as paraphrases that often occur in the same highly discriminating left and right contexts (called distributional similarity). Also multilingual corpora with bi-lingual translation tables have been considered: all terms mapped to the same term in one or more translation tables are good candidates for paraphrases. Today, there is a growing number of approaches employing more powerful machine learning-based techniques to reliably detect paraphrases even in the context of unsupervised information extraction [50, 51], as well as scalable lightweight approaches based on clustering lexical patterns tailored for specific applications like analogy queries [52].

The question of contexts for relationship validity remains still an unsolved problem at large. Mining contexts from unstructured data has been used for a verity of context-aware tasks, but attempts at incorporating contexts into knowledge bases are currently only starting like, for instance, the annotation of validity time for facts in the Timely Yago knowledge base [53].

3 Hybrid Extraction Techniques

Despite the tremendous advances of information extraction techniques in recent years, open issues still remain regarding quality, reliability, and complexity, which hamper the widespread adoption of IE techniques. The task of building a knowledge base generally requires selecting of data sources, extracting the respective entities, classes, and relationships, and then finally integrating and linking the new information into the already existing knowledge base. While each of these steps is covered by some of the methods presented in the previous sections, their result quality does often not live up to the required standards of the respective applications. Therefore, many companies still rely on manual curation of their knowledge bases (e.g., the New York Times employs a special team for manually linking and creating their subject header knowledge base, see http://data.nytimes.com).

This need for human assistance for bridging the final quality gap has given rise to the extraction techniques discussed in this section which rely on hybrid architectures, transparently combining the efficiency of current *algorithms* with the cognitive power and flexibility of *humans*. Here, generally two design directions are popular [54]:

- Using human input for improving the steps performed by an information extraction algorithm by providing training samples[55], answering questions about ambiguous results [56], or by providing relevance feedback [57].
- Involving humans directly into the information extraction process, explicitly outsourcing some of the required tasks of the extraction process.

Early examples like Simple/DBLife [58–60] automatically extract structured data from the Web, which is then exposed as Wiki pages. Users of the Wiki platform are then asked to correct and augment the extracted data. Here, one of the major open challenges lies in recruiting and retaining the user pool for performing this task. This issue can be addressed by exploiting general purpose crowd-sourcing platforms like Amazon Mechanical Turk (https://www.mturk.com), crowdflower (http://crowdflower.com), or SamaSource (http://samasource.org). The base idea is that a larger task is divided into many microtasks (called HITs, Human Intelligence Tasks). The HITs are then assigned to a pool of human workers which are paid for their efforts. This allows the design of flexible systems which can dynamically request HITs during runtime without the necessity of building and retaining a user community—i.e., if the need for human cognitive power arises, it can simply be bought. While these crowd-sourcing platforms cheaply deliver human workers in an ad-hoc fashion, many of those users are malicious and aim at maximizing their monetary income by cheating. Therefore, strict *quality control* and detection of unreliable workers is mandatory.

Furthermore, controlling the *costs* of the task execution is an important challenge as each issued HIT will have to be paid: in typical hybrid systems utilizing crowd-sourcing like crowd-enabled databases [61], a HIT is often issued every time the systems requires a piece of missing information or some kind of intelligent decision. This behavior will quickly result in dramatically increasing operations costs. Therefore, when approaching Web-scale information extraction tasks, it is crucial to dynamically determine if the costs are justified and only resort to human workers in a selective fashion.

In the hybrid system ZenCrowd [62], entities and relationships are automatically extracted and linked to existing entities using state-of-the-art algorithms as described in the previous section. The results of the automatic linking are then processed by a probabilistic network which computes the confidence in the correctness of the automatic linking. If the confidence is high, the results are directly incorporated into the knowledge base. If the confidence is very low, the results are discarded. In the case that the result looks promising, but cannot be validated by the probabilistic network, HITs are a dynamically issued to a crowd-sourcing platform for recruiting human workers to verify and correct the results obtained automatically.

In contrast, in [63] crowd-sourcing is used to efficiently adjust and train a complex suite of extraction and learning algorithms on-the-fly. The basic challenge is similar to general crowd-enabled databases where missing information in database tables has to be obtained during query runtime. But here the system automatically extracts perceptual and subjective feedback on database items from the Social Web. For instance in a movie database like IMDb (http://www.imdb.com), not only structured information, but also user ratings and reviews of all movies are collected. Using this information, a perceptual space can be constructed, encoding the subjective and consensual perception of the Social Web users in a generic fashion using recommender system techniques. If a query requires some attribute which is not part of the initial database table (e.g., a movie's fun factor or excitement), the required attribute values can be extracted from the perceptual space as long as the attribute is dependent on subjective human perception (and not a ground fact like the release date or director). For extracting this information, a machine learning algorithm to extract the desired values can be trained on-the-fly relying on relatively small training sets generated by crowd-sourcing.

In summary, integrating a certain amount of human intelligence directly into the information extraction task promises to improve the quality of the resulting knowledge bases. Moreover, using a hybrid approach, even traditionally extremely hard-to-extract perceptual information which is usually only implicitly available on the Web can be made accessible efficiently and cheaply.

4 Conclusions

Without claiming completeness, this survey aims at providing a good overview of how and what to extract automatically from mostly unstructured information sources like the World Wide Web. Basic elements are (named) entities, classes or categories of entities, entity attributes and characteristics, and relationships between entities. Methods are mostly based on lexical and/or syntactical patterns together with a wide variety of statistical machine learning algorithms, suitable and usually simple inference rules, and general constraints.

Looking at the diversity and complexity of problems in information extraction, it can be stated that although huge advances have been made in extraction techniques and methodologies in recent years, the problem at large remains a challenge. While basic extraction techniques for entities are already usable and can create collections with high precision and recall (e.g., for business intelligence or content management), the extraction of general relationships between entities is still far from being mature. Especially, the problem of deciding what makes a useful relationship and the semantic problems given by the often unclear context of facts are important to solve for many practical applications.

References

1. Craven M, DiPasquo D, Freitag D, McCallum A, Mitchell T, Nigam K, Slattery S (1999) Learning to construct knowledge bases from the World Wide Web. Artif Intell
2. Weikum G, Theobald M (2010) From information to knowledge: harvesting entities and relationships from web sources. In: Proc of ACM symposium on principles of database systems (PODS), Indianapolis, USA
3. Buneman P, Khanna S, Tan WC (2001) Why and where: a characterization of data provenance. In: Proc of 8th international conference on database theory (ICDT), London, UK
4. Tayi GK, Ballou DP (1998) Examining data quality. Commun ACM 41(2)
5. Soderland S (1999) Learning information extraction rules for semi-structured and free text. Mach Learn 34(1–3)
6. Kushmerick N (2000) Wrapper induction: efficiency and expressiveness. Artif Intell 118(1–2)
7. d'Oro L, Ruffolo M, Staab S (2010) SXPath—extending XPath towards spatial querying on web documents. In: Proc of international conference on very large data bases (VLDB), Singapore. PVLDB, vol 4(2)
8. Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R, Ives ZG (2007) DBPedia: a nucleus for a web of open data. In: The semantic web (ISWC/ASWC 2007). LNCS, vol 4825. Springer, Berlin
9. Jurafsky D, Martin JH (2008) Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition. Prentice Hall, New York
10. Gildea D, Jurafsky D (2000) Automatic labeling of semantic roles. In: Proc of annual meeting of the association for computational linguistics (ACL), Hong Kong, China
11. Grishman R, Sundheim B (1996) Message understanding conference—6: a brief history. In: Proc of international conference on computational linguistics (COLING), Kopenhagen, Denmark
12. Malouf R (2002) Markov models for language-independent named entity recognition. In: Proc of conference on natural language learning (CoNLL), Taipei, Taiwan
13. Curran JR, Clark S (2003) Language independent NER using a maximum entropy tagger. In: Proc of conference on natural language learning (CoNLL), Edmonton, Canada
14. Bunescu RC, Mooney RJ (2004) Collective information extraction with relational Markov networks. In: Proc of annual meeting of the association for computational linguistics (ACL), Barcelona, Spain
15. Finkel JR, Grenager T, Manning C (2005) Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proc of annual meeting of the association for computational linguistics (ACL), Ann Arbor, MI, USA
16. Hearst M (1992) Automatic acquisition of hyponyms from large text corpora. In: Proc of international conference on computational linguistics (COLING), Nantes, France
17. Etzioni O, Cafarella M, Downey D, Popescu A, Shaked T, Soderland S, Weld DS, Yates A (2005) Unsupervised named-entity extraction from the web: an experimental study. J Artif Intell 165(1)
18. Banko M, Cafarella M, Soderland S, Broadhead M, Etzioni O (2007) Open information extraction from the web. In: Proc of international joint conference on artificial intelligence (IJCAI), Hyderabad, India
19. Bunescu RC, Pasca M (2006) Using encyclopedic knowledge for named entity disambiguation. In: Proc of conference of the European chapter of the association for computational linguistics (EACL), Trento, Italy
20. Cucerzan S (2011) Large-scale named entity disambiguation based on Wikipedia data. In: Proc of conference on empirical methods in natural language processing (EMNLP), Edinburgh, UK
21. Hoffart J, Yosef M, Bordino I, Fürstenau H, Pinkal M, Spaniol M, Taneva B, Thater S, Weikum G (2011) Robust disambiguation of named entities in text. In: Proc of conference on empirical methods in natural language processing (EMNLP), Edinburgh, UK
22. Hassell J, Aleman-Meza B, Arpinar IB (2006) Ontology-driven automatic entity disambiguation in unstructured text. In: Proc of international semantic web conference (ISWC), Athens, GA, USA
23. Bekkerman R, McCallum A (2005) Disambiguating web appearances of people in a social network. In: Proc of international conference on World Wide Web (WWW), Chiba, Japan
24. Dorow B, Widdows D (2003) Discovering corpus-specific word senses. In: Proc of conference of the European chapter of the association for computational linguistics (EACL), Budapest, Hungary
25. Nie Z, Ma Y, Shi S, Wen J, Ma W (2007) Web object retrieval. In: Proc of international conference on World Wide Web (WWW), Banff, Canada
26. Nie Z, Wen J, Ma W (2007) Object-level vertical search. In: Proc of biennial conference on innovative data systems research (CIDR), Asilomar, CA, USA
27. Dey D, Sarkar S, De P (2002) A distance-based approach to entity reconciliation in heterogeneous databases. IEEE Trans Knowl Data Eng 14(3)
28. Dong X, Halevy A, Madhavan J (2005) Reference reconciliation in complex information spaces. In: Proc of ACM international conference on management of data (SIGMOD), Baltimore, MD, USA
29. Chaudhuri S, Ganti V, Xin D (2009) Mining document collections to facilitate accurate approximate entity matching. In: Proc of international conference on very large data bases (VLDB), Lyon, France. PVLDB, vol 2(1)
30. Hearst M (1992) Automatic acquisition of hyponyms from large text corpora. In: Proc of international conference on computational linguistics (COLING), Nantes, France

31. Charniak E, Berland M (1999) Finding parts in very large corpora. In: Proc of annual meeting of the association for computational linguistics (ACL), College Park, MD, USA
32. Cederberg S, Widdows D (2003) Using LSA and noun coordination information to improve the precision and recall of automatic hyponymy extraction. In: Proc of conference on natural language learning (CoNLL), Edmonton, Canada
33. Stoica E, Hearst M, Richardson M (2007) Automating creation of hierarchical faceted metadata structures. In: Proc of human language technology conference of the association of computational linguistics, Rochester, NY, USA
34. Cimiano P, Handschuh S, Staab S (2004) Towards the self-annotating web. In: Proc of international conference on World Wide Web (WWW), New York, NY, USA
35. Sanderson M, Croft B (1999) Deriving concept hierarchies from text. In: Proc of international ACM SIGIR conference on research and development in information retrieval, Berkeley, CA, USA
36. Diederich J, Balke W (2007) The semantic GrowBag algorithm: automatically deriving categorization systems. In: Proc of European conference on research and advanced technology for digital libraries (ECDL), Budapest, Hungary
37. Jäschke R, Hotho A, Schmitz C, Ganter B, Stumme G (2008) Discovering shared conceptualizations in folksonomies. J Web Seman 6(1)
38. Cohen W (1998) Integration of heterogeneous databases without common domains using queries based on textual similarity. In: Proc of ACM international conference on management of data (SIGMOD), Seattle, WA, USA
39. Mena E, Kashyap V, Illarramendi A, Sheth A (2000) Imprecise answers in distributed environments: estimation of information loss for multi-ontology based query processing. Int J Cooperat Inf Syst 9(4)
40. Rodriguez M, Egenhofer M (2003) Determining semantic similarity among entity classes from different ontologies. IEEE Trans Knowl Data Eng 15(2)
41. Gracia J, d'Aquin M, Mena E (2009) Large scale integration of senses for the semantic web. In: Proc of international conference on World Wide Web (WWW), Madrid, Spain
42. Fader A, Soderland S, Etzioni O (2011) Identifying relations for open information extraction. In: Proc of conference on empirical methods in natural language processing (EMNLP), Edinburgh, UK
43. Kasneci G, Ramanath M, Suchanek F, Weikum G (2008) The YAGO-NAGA approach to knowledge discovery. SIGMOD Rec 37(4)
44. Wu F, Weld D (2007) Autonomouslly semantifying Wikipedia. In: Proc of ACM international conference on information and knowledge management (CIKM), Lisbon, Portugal
45. Brin S (1998) Extracting patterns and relations from the World Wide Web. In: Proc of international workshop on the World Wide Web and databases (WebDB), Valencia, Spain
46. Zhu J, Nie Z, Liu X, Zhang B, Wen J (2009) StatSnowball: a statistical approach to extracting entity relationships. In: Proc of international conference on World Wide Web (WWW), Madrid, Spain
47. Agichtein E, Gravano L (2000) Snowball: extracting relations from large plain-text collections. In: Proc of ACM international conference on digital libraries (DL), San Antonio, TX, USA
48. Suchanek F, Sozio M, Weikum G (2009) SOFIE: a self-organizing framework for information extraction. In: Proc of international conference on World Wide Web (WWW), Madrid, Spain
49. Nakashole N, Theobald M, Weikum G (2011) Scalable knowledge harvesting with high precision and high recall. In: Proc of ACM international conference on web search and data mining (WSDM), Hong Kong, China
50. Kok S, Domingos P (2008) Extracting semantic networks from text via relational clustering. In: Proc of European conference on machine learning and knowledge discovery in databases (ECML/PKDD), Antwerp, Belgium
51. Yates A, Etzioni O (2009) Unsupervised methods for determining object and relation synonyms on the Web. J Artif Intell Res 34
52. Bollegala D, Matsuo Y, Ishizuka M (2009) Measuring the similarity between implicit semantic relations from the web. In: Proc of international conference on World Wide Web (WWW), Madrid, Spain
53. Wang Y, Zhu M, Qu L, Spaniol M, Weikum G, (2010) Timely YAGO: harvesting, querying, and visualizing temporal knowledge from Wikipedia. In: Proc of international conference on extending database technology (EDBT), Lausanne, Switzerland
54. Doan A, Ramakrishnan R, Halevy AY (2011) Crowdsourcing systems on the World-Wide Web. Commun ACM 54
55. Raykar VC, Yu S, Zhao LH, Valadez GH, Florin C, Bogoni L, Moy L (2010) Learning from crowds. J Mach Learn Res 11
56. McCann R, Shen W, Doan A (2008) Matching schemas in online communities: a web 2.0 approach. In: Proc of the international conference on data engineering (ICDE), Cancun, Mexico
57. Alonso O, Rose DE, Stewart B (2008) Crowdsourcing for relevance evaluation. In: ACM SIGIR forum. ACM, New York
58. Chai X, Gao BJ, Shen W, Doan A, Bohannon P, Zh X (2008) Building community Wikipedias: a machine-human partnership approach. In: Proc of int conf on data engineering (ICDE), Cancun, Mexico
59. DeRose P, Shen W, Chen F, Lee Y, Burdick D, Doan A, Ramakrishnan R (2007) DBLife: a community information management platform for the database research community. In: Proc of conference on innovative data systems research (CIDR), Asilomar, CA, USA
60. Chai X, Vuong B, Doan A, Naughton JF (2009) Efficiently incorporating user feedback into information extraction and integration programs. In: Proc of ACM international conference on management of data (SIGMOD), Providence, RI, USA
61. Franklin M, Kossmann D, Kraska T, Ramesh S, Xin R (2011) CrowdDB: answering queries with crowdsourcing. In: Proc of ACM international conference on management of data (SIGMOD), Athens, Greece
62. Demartini G, Difallah DE, Cudré-Mauroux P (2012) ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In: Proc of international World Wide Web conference (WWW), Lyon, France
63. Selke J, Lofi C, Balke W (2012) Pushing the boundaries of crowd-enabled databases with query-driven schema expansion. In: Proc of international conference on very large data bases (VLDB), Istanbul, Turkey. PVLDB, vol 5(6)