Semi-supervised Bootstrapping of Relation Triples from the Web, Query Languages over these Noisy Triples, their Semantics, and Query Execution Systems

Amit Kumar Singh (07305046)
amitsingh@cse.iitb.ac.in

December 25, 2007

Abstract

Information Extraction (IE) is the process of retrieving structured information from unstructured text. IE has traditionally relied on extended human interposition to extract small set of predefined relations from the corpus. Now with *Web* coming in to picture, methods and goals of IE have taken a slight detour, with increasing focus on following challenges

1. Domain independent/Open Information Extraction (minimize-obviate human intervention)
2. Leverage redundant (implicitly structured) information on the ever growing web.
3. Capable of dealing with large level of ambiguity/noise in Natural Language.
4. Scalability of the IE systems to web-size dynamic corpora
5. Ability to Extract/Integrate Relations of arbitrary-arity
6. Ability to efficiently retrieve answers to arbitrary structured queries on the extracted data/schema

In the remaining portion of this document, I intend to highlight some of the significant developments [1] made in this area starting from domain specific, hand-tagged IE systems to state of the art self-supervised Open IE framework and beyond.

[1]Instead of describing the algorithms used in the IE Systems, I have tried to enlist some significant characteristics inferred while studying these systems.

Contents

1 DIPRE : Seed Tuple based technique

A powerful approach to reduce the training cost in extraction of structured relations from large text collection was pioneered by Brin in [4]. The idea in Dual Iterative Pattern Relation Extraction (*DIPRE*) was simple yet effective, rather than demand hand-tagged corpora, start with just a handful of seed tuples for the relation of interest, and automatically discover extraction patterns for the task. These patterns, in turn, help discover new tuples for the relation, which could be used as new seed tuples for a next iteration of the process.

This could "potentially" reduce the effort needed to adapt an IE system for a new target relation to simply providing the initial seed tuples for the relation. A set of hand-constructed extraction patterns could also be used to kick-start the training process.

Potential pitfalls :

1. Degrade quality of tuples on subsequent iterations due to introduction of bogus/bad tuples.
2. No automatic way to select high quality tuples to use as new seed for subsequent iteration.
3. Naive way to generalize patterns was suggested

Several other approaches similar to *DIPRE* were suggested. One of them was [11]. Apart from exploiting the duality that exists between patterns and target tuples (similar to *DIPRE*), topic specific knowledge is used to formulate new dualities and use it in the IE process to improve the bootstrapping results. e.g. For extracting Relation (acronym, expansion) we have following dualities that are used.

- Duality between related pairs and their patterns
- Dulaity between related pairs and acronym formulation rules.

2 SnowBall : Seed Tuple based technique

This paper [8] describes Snowball, a substantial generalization and extension of *DIPRE*, used to extract relational tuples from large text corpus. Just like *DIPRE*, it exploits regularity in natural-language text to extracts predefined relations by starting with just a handful of seed tuples.

The key secernating point of snowball from *DIPRE* are as follows

1. Rules for determining, whether context around entity to be extracted is "appropriate" : SnowBall uses *Vector Space rules* (feature vector to describe valid text contexts) in lieu of *String Match Rules(LCS)* used in *DIPRE*, later is empirically shown to exhibit poor performance on unstructured text.
2. Strategy for evaluating relational tuples.
3. SnowBall regards target-tuples as entities instead of plain text. This further improves the quality of pattern since it automatically ignores unwanted entities.
4. Automatically Evaluation of Extraction Patterns (constraint based evaluation)

Potential pitfalls :

1. Use of Entity tagger limits the scope of use of system
2. Success of the system depends on many parameters. (Many things can go wrong)
3. Adapting Snowball for new scenarios with as little manual re-tuning was not possible.
4. Suffers from "concept drift" problem i.e overly general patterns for some domains
5. Localized patterns generated by snowball might not be sufficient for some relations as it disregards larger contexts.

Some of these pitfalls were addressed in *Snowball-Auto*, system built for automatically estimating many of the *Snowball* parameters to additionally reduce the amount of tuning required for porting *Snowball* to new domains.

There were also attempts [7] to successfully combine *SnowBall* with a hand-crafted, domain-specific system, resulting in recall and precision exceeding that of the original systems.

Approaches to automatically evaluate Extraction Patterns using EM were suggested in [1]

All the above efforts to make *Snowball*, "kind of" *domain independent* were not very successful. It was found that there is *no single configuration* of the *Snowball* system, which performed best in all domains. Some approaches were superior in one domain and not quite so in others.

3 QXtract: General Keyword Index-based technique

QXtract [10] presents a novel query-based technique for efficient IE. Some of the salient features are as follows

1. Successfully demonstrates use of *Generic Indexes* available via existing commercial search engines for IE.
2. Can work over non-crawlable "*Hidden Web*" databases
3. Uses sampling of documents instead of "Tuples" strategy to deal with *Reachability* problem described in [9]
4. *Automatically Query Generation (AQG)* to retrieve useful documents
5. Suitable for problems with *low reachability* [9] and *sparse data*

Potential pitfalls :

1. To achieve acceptable recall, large amounts of documents needs to be retrieved.
2. Generic Indexes (keywords) are approximate so the queries generated need not be accurate
3. Useful Documents collection may be dominated by results of one query thereby possibly giving skewed results.

4 KnowItAll: Self-Supervised General Keyword Index-based technique

KnowItAll [16] is a self-supervised, domain-independent system for IE i.e instead of relying on hand-tagged data, *KnowItAll* autonomously creates its own labeled examples and uses it to train NB classifier.

Salient Features :

1. The primary focus of the system is on extracting *Entities* (unary predicates)
2. Does not carry the load of heavy linguistic machinery like *NER* and *dependency parsers* instead relies on *part of speech recognizers.*
3. Its the *first fully automatic bootstrapping techniqu*e of its kind which does not require any seed tuples.
4. *Relation Specific IE* : Only input to the system is, definition of relations to be extracted, so potentially can be adapted to new domains without much human intervention.
5. *KnowItAll* uses a set of manually-built generic rules, which are instantiated with the target predicate names, producing queries, patterns and discriminator phrases.
6. It also demonstrates the idea of automatically evaluating the validity of the extracted facts (Assessor module) using *PMI* statistics computed leveraging existing web search engines.
7. Laborious bootstrapping process to train itself hence scalability of system to web size corpus is an issue (proposed PMI based assessment is expensive)
8. Uses heuristics like *Signal to Noise Ratio* of each class to steer the *IE* process in the form of *Resource Allocation* to most productive relations in subsequent iteration and overall process *termination criteria*

Direct successor of *KnowItAll* is *URES* [3] which exhibits better recall than *KnowItAll* at equal or better levels of precision.

5 KnowItNow: Specialized Index-based technique

KnowItNow [14] is a *self-supervised* learner, modeled to improve the efficiency of *KnowItAll* by forfending its query related bottlenecks. As time to process a query is same in case of home-grown or existing search engines, limit on *rate of issuing queries* is identified as the *performance bottleneck*.

KnowItNow introduces novel idea of using *Specialized Local Indexes* i.e *Binding Engine (BE)* , optimized for finding instances of *KnowItAll* rules and patterns in lieu of existing commercial search engines for query based IE.

Salient Features :

1. Neighbouring indexes in *BE* helps in significant improved query times but at the cost of extra space.(can be problematic as the size of corpus grows)
2. Use of new *URNS* model in *Acessor* in lieu of expensive *PMI* used in KnowItAll, can lead to increased errors although improved efficiency. Instances generated from each extractor are assumed to be a mixture of two distributions (Random Noise and other with some known structure like Uniform). EM can be employed to estimate mixture probabilities and parameters of non-noisy data.
3. Due to limited size of specialized indexes, *KnowItNow* can work poorly on sparse data problems.

6 TextRunner: Open Information Extraction

TextRunner [15] is the *First* domain independent, self-supervised Information extraction system in which relations (along with its instances) are learn't ***on the fly***.

Salient Features :

1. Highly efficient single pass approach w.r.t earlier systems e.g KnowItAll, with same/better quality of extracted relations.
2. The sole input to an Open IE system is a corpus, and its output is a set of extracted relations.
3. Self-Supervised Classifier module uses rich representation (parse trees) for training the Naive Bayes classifier, used in extractor module to label extracted-candidate tuples.
4. Uses *specialized indexes* (*Lucene* Search engine) distributed across machine-clusters.(to obviate any dependency on commercial search engine).
5. Uses *URNS* model for accessing the quality of extractions
6. Difficult to learn Relations with *very-sparse data* because for *TextRunner*, rare extractions are considered errors and receive a low probability.
7. *Synonymity* problem in Relation and its extracted instances is ascertained.

This *Synonymity* problem was addressed through domain-independent, unsupervised synonym resolution system called "*Resolver*" [2]. *Resolver* introduces a probabilistic relational model for predicting whether two strings are co-referential based on the similarity of the respective extraction contexts.

7 ExDB : Structured Queries Over Web Text

ExDB [13], a general purpose query framework , that supports structured queries over data extracted from unstructured web is proposed in this paper. It is based on *Extraction Database Model* that takes the middle approach between the two extremes, *Query Based models* and *The Schema Extraction Model*.

Salient Features :

1. It uses any IE system as a black box to extract structured data which is then stored in *Probabilistic Databases*, These DB's can be queried with language similar to SQL but adapted to comprehend probabilistic queries.
2. It proposes a data model (*WebDB*) which consists of Objects, Binary Relations, Semantic Types as base concepts that are extracted by IE systems.
3. It also proposes extraction of other relationships like Synonyms,Inclusion dependencies,Functional dependencies, to aid in query resolution (rewrites/expansion) by providing a probabilistic similarity mapping between tuples extracted and user query.
4. Probabilistic representation of uncertainty at all levels in *WebDB* (Data, schema, constraints)
5. Query language supports *SPJU* semantics of RDBMS .
6. *WebDB* model supports complex queries which cannot be answered by search engines. (e.g join of extracted facts from many diverse sources)
7. Proposes an optimizing framework that uses *Panel of Expert* semantics
8. Ability to reason the extracted answers, using *query lineage* and *extraction lineage*

8 Miscellaneous Systems

8.1 Learning Dictionaries for Information Extraction by Multilevel Bootstrapping

This paper [6] introduces 2-level *mutually bootstrapping* technique that builds on duality of extraction patterns and target words. It is used to create domain specific dictionary of extraction pattern and semantic lexicon.

Salient Features :

1. Meta-bootstrapping is done to make it more robust to noise in the extracted patterns/tuples.
2. Inner-bootstrapping procedure selects the best patterns for a set of seed words. ($RLogF$ metric introduced in *AutoSlog-TS*[18] is used to score patterns)
3. Outer-bootstrapping procedure repeats the inner procedure with refined seed words adding *Top-K* target-words to dictionary per iteration,leading to a snowball effect. This encourages *re-evaluation* of extraction patterns with better seed tuples over successive iterations, improving quality of the dictionary generated.

Potential pitfalls :

1. Stopping condition for bootstrapping not well defined
2. Function to score tuples is susceptible if score is skewed by one pattern that extracts large amount of information

8.2 BASILISK

BASILISK [5] is an improvement over the Multilevel Bootstrapping technique discussed earlier.

Salient Features :

1. Here instead of trusting individual extraction pattern like *Meta-bootstrapping* to make individual decisions, collective evidence is used from large set of patterns to infer semantic category of the extraction.

2. It alleviates the problem caused due to skewness in scoring tuples using a better scoring function.

3. In addition to this, it can be used to learn semantic dictionaries for *multiple categories simultaneously*. This combined with sophisticated conflict resolution strategy (if a word belongs to multiple dictionaries which one it should be assigned to) can improve the quality of the extracted word considerably by steering the bootstrapping process away from ambiguous parts of search space.

4. Presence of only *one bootstrapping loop* as compared to earlier strategy which had 2, leads to improved efficiency.

Potential pitfalls :

1. Many bootstrapping iterations required before multi category search has benefit.

2. "One Sense per domain" assumption might not be valid in some cases.

8.3 Organizing and Searching the WWW of Facts

This paper [12] adopts the scan based approach similar to SnowBall.

Salient Features:

1. It demonstrates technique to up the convergence rate to desired recall by *generalizing the candidate patterns* extracted from the text.
2. For the purpose of generalization, *automatically constructed thesaurus of distribution similar words* is used.
3. It also shows that higher recall can be achieved by using various heuristics such as fiddling with the left,middle,right context of the tuples extracted.

A sequel to this paper was [17]. It describes unsupervised, domain-independent, method to *mine search queries* instead of web corpus to *extract attributes of target relations.*

References

[1] Eugene Agichtein. Confidence estimation methods for partially supervised relation extraction. 2005.

[2] Oren Etzioni Alexander Yates. Unsupervised resolution of objects and relations on the web. 2007.

[3] Ronen Feldman Benjamin Rosenfeld. URES : an unsupervised web relation extraction system. 2006.

[4] Sergey brin. Extracting patterns and relations from WWW. 1998.

[5] Michael Thelen Ellen Riloff. A bootstrapping method for learning semantic lexicons using extraction pattern contexts. 2002.

[6] Rosie Jones Ellen Riloff. Learning dictionaries for information extraction by multilevel bootstrapping. 1999.

[7] Eleazar Eskin Eugene Agichtein, Luis Gravano. Combining strategies for extracting relations from text collections. 2000.

[8] Luis Gravano Eugene Agichtein. Snowball: Extracting relations from large plain-text collections. 2000.

[9] Luis Gravano Eugene Agichtein. Modelling query based access to text databases. 2003.

[10] Luis Gravano Eugene Agichtein. Qxtract: Querying text databases for efficient information extraction. 2003.

[11] Neel Sundaresan Jeonghee Yi. Mining the web for acronyms using the duality of patterns and relations. 1999.

[12] Jeffrey Bigham Andrei Lifchits Alpa Jain Marius Pasca, Dekang Lin. Organizing and searching the WWW of facts - step one: the one-million fact extraction challenge. 2006.

[13] Dan Suciu Michael Cafarella, Oren Etzioni. Structured querying of web text. 2006.

[14] Stephen Soderland Oren Etzioni Michael Cafarella, Doug Downey. KnowItNow: fast, scalable information extraction from the web. 2005.

[15] Stephen Soderland Matt Broadhead Michele Banko, Michael J. Cafarella and Oren Etzioni. Open information extraction from the web. 2007.

[16] Doug Downey Ana-Maria Popescu Tal Shaked Stephen Soderland Daniel S. Weld Alexander Yates Oren Etzioni, Michael Cafarella. KnowItAll : Unsupervised named-entity extraction from the web: An experimental study. 2005.

[17] Marius Pasca. Organizing and searching the worldwideweb of facts - step two: Harnessing the wisdom of crowds. 2007.

[18] Ellen Riloff. AutoSlog: Automatically generating extraction patterns from untagged text. 1996.