Building a Knowledge base through Open Information Extraction techniques

Chua, Alloy
De La Salle University Manila
alchua01@yahoo.com

Cheng, Charibeth
De La Salle University Manila
chari.cheng@delasalle.ph

ABSTRACT
The emergence of the Open-IE paradigm has given way to domain independent techniques for information extraction. Its capability of extraction new relations from text shows potential towards the task of building a knowledge base. However, its advantages also come with its disadvantages such as the lack of context expressed by the relations extracted by Open-IE systems. This research attempts to tackle the different problems associated with the incorporation of Open-IE towards the building of a knowledge base.

Keywords
Knowledge base, Open Information Extraction, Relation Extraction, Natural Language Processing

1. INTRODUCTION AND MOTIVATION
The application and usefulness of knowledge bases such as ontologies have been recognized in many fields such as in Artificial Intelligence (AI) and Natural Language Processing (NLP). These knowledge bases represent knowledge within the real world, such as objects/concepts and the relationships between them, and are used by machines to perform tasks that require a certain level of understanding. However, knowledge bases such as ontologies are traditionally constructed manually which is inefficient in terms of scalability and maintenance. Given this bottleneck, automated approaches for building such knowledge bases have been explored. One such approach has been the usage of NLP techniques to construct knowledge bases from text. As the field of NLP involves analysis of text in order to perform content extraction, certain NLP techniques or tasks have been used in the past towards knowledge acquisition for building knowledge bases. One such NLP task used is Information Extraction (IE). Previous researches have already incorporated IE in knowledge acquisition. Albeit being effective towards knowledge acquisition, issues arise which limit the performance of IE systems. The following discusses in detail the issues and challenges that arise in the incorporation of IE in the task of building a knowledge base and how this research attempts to tackle them.

Domain Dependence Approaches towards context extraction in systems are mostly linguistically dependent on the targeted corpus or domain such as the need for the creation of extraction rules or templates based on the corpus specifying the desired information to be extracted. These approaches also heavily rely on contextual resources to determine the appropriate information to be extracted from the text. These resources however are limited in terms of coverage as they capture or reflect only the domain of interest. Such approaches lead to traditional IE systems being domain-specific thus, adapting to a new domain would require new resources or a different approach altogether. Such tailor-fitting of the approaches in an IE system to a specific domain introduces problems in terms of scalability [2]. Moreover, given the limitations of relation specificity in traditional IE systems, most systems have focused only towards populating a predefined knowledge base.

In recent years, a new IE paradigm, called the Open-IE, has been introduced in the attempt of overcoming the problems in scalability involved in traditional IE systems. Compared to the traditional IE paradigm, the Open-IE paradigm involves no tailor-fitting as its motivation is domain independence. Instead of relying on approaches that are derived from or dependent on a specific corpus or domain, the Open-IE paradigm relies on domain-independent approaches in extracting important information. Given this approach, an Open-IE system would not require other external resources that are domain-specific, which limits scalability [2].

The Open-IE paradigm has already been used in the collection of information from the web. KnowItAll [5] is an Open-IE system that automatically extracts entities (or arguments involved in an instance-of relationship) from the web given an initial ontology of the classes of interest. It was able to extract entities (e.g. Paris is an instance of the class City) using a set of domain-independent rule patterns (e.g. “city such as X”). Improvements in the KnowItAll system enabled it to also extract subclasses of the classes in the pre-defined ontology [4]. For instance, the subclasses Supplier and Exporter have been extracted for the class Person. Subclass extraction was done using an approach similar to the one used in entity extraction. Besides from entity extraction, the Open-IE paradigm has also been applied in relation extraction (RE). However, Open-RE does not deal with the usual relation-specificity in traditional RE as relations of interest differ across domains. In comparison, the traditional RE relies on pre-specified relations that reflect the domain and the information of interest, whereas the Open-RE does not and instead focuses on the extraction of relations as well as the participating entities on any given corpus. For example, traditional RE systems would extract John and Mary as participants on the pre-defined relation husband of. Other linguistic resources are used to determine that the entities John and Mary are indeed appropriate participants. In contrast, Open-
RE systems would extract not only John and Mary but also the relationship husband of. As there are no pre-defined relations prior to extraction, the task of determining and extracting the relation between the entities is now also part of Open-RE. TEXTRUNNER [2] and O-CRF [1] are systems that focus on the Open-RE task and extract relational tuples on any given corpus. These systems perform Open-RE mainly through statistical and pattern-learning approaches.

The capability of Open-IE to determine new relational tuples from text shows potential towards the task of building a knowledge base. Relation specificity in traditional IE systems has limited most systems towards only population of a knowledge base. In contrast, being able to extract new relational tuples from text could give way to methods that would build a knowledge base from scratch. Given this potential, this research aims to incorporate Open-IE into the task of knowledge acquisition in the goal of building a knowledge base and solve the problems involved in traditional IE’s limitation of domain dependence.

Information to Knowledge representation The potential of Open-IE systems lies on their capability of extracting new relational tuples from text. However, the information expressed by such tuples are still limited. Since most Open-IE approaches involve pattern analysis and without making use of any contextual resources or approaches to determine possible relations, the extracted relational tuples result to having a lack of context. For example, the argument ball can exist in the two tuples ball-has-size and ball-has-dancers. The first tuple ball-has-size pertains to the ball that is an object whereas the second tuple ball-has-dancers pertains to the ball that is an event. Since tuples are independent from each other and there is no underlying context to distinguish the two, it cannot be determined from the tuples alone that the two tuples pertain to two different senses of the argument ball. In contrast to relational tuples, knowledge bases are able to express higher levels of information due to their structured form as the structure itself expresses a certain level of context such as relatedness. For example, a knowledge base can be structured in such a way that the different senses of the word ball are separated from each and wherein each sense of the word is grouped with its appropriate relations. Given these, it is difficult to build a knowledge base able to express a certain level of context through relational tuples alone which lack context. Traditional IE techniques have been applicable towards the task of populating a knowledge base through the use of contextual resources and domain knowledge. Context is needed to build a knowledge base however, Open-IE techniques cannot make use of the same resources as it will result to problems in scalability. A different source of context must be used towards the incorporation of Open-IE techniques to the task of building a knowledge base that will not result to a compromise in scalability.

Given this problem, this research aims to use paragraph features as a source of context. Paragraphs naturally convey coherence or relatedness between information expressed in their sentences. Given this paragraph feature, it can be used for the task of linking together extracted assertions and their arguments into a graph. For example, each paragraph contains a main topic which all sentences explain in further detail. The main topic of a paragraph serves as the context of all ideas conveyed by the sentences of the paragraph. Given this, the main topic of a paragraph can be used as the localizing context of any information extracted from the sentences of the paragraph and thus can also be treated as the main node of the graph. Using the concept of coherence between sentences within a paragraph, assertions found in each sentence of the paragraph can then be associated and linked to the main node of the graph. Each paragraph will effectively result to a graph in which its information is localized to the context of the main topic of the paragraph thus addressing the problem of the lack of context in assertions that are extracted.

This paper reports on preliminary results of a system architecture which follows the proposed solutions to the problems stated earlier. The following section introduces the system architecture in detail.

2. OPEN-IE KNOWLEDGE BASE BUILDER

As seen in Figure 1, the proposed system architecture is composed of four main modules which are namely (1) the Crawler Module (2) the Extractor Module (3) the Builder Module and (4) the Resolver module. The following subsections discuss each module in detail as well as the theoretical framework detailing possible approaches towards fulfilling the module requirements and goals.

2.1 Crawler Module

The Crawler Module is responsible for the task of web-crawling as well as the task of parsing the gathered web pages to extract blocks of paragraphs and their sentences. The module is composed of two sub-modules which are (1) the Pre-extraction Cleaner and (2) the HTML Parser. It takes in as input a web page and outputs extracted paragraphs as well as their sentences.

2.1.1 Pre-extraction Cleaner

One problem in getting paragraphs from a web page is the existence of noisy and unnecessary data such as advertisements. In addition, some web pages do not make use of the intended HTML tag “<p>” for displaying paragraphs. Given this, there is a need to clean up the web pages prior to the extraction of paragraphs in order to remove unnecessary data and arrange together text that are intended as belonging to one paragraph. The Pre-extraction Cleaner sub-module performs this task by taking as input a web page and outputs a new web page containing only potentially important information arranged as a list of paragraph tags. This new web page is then used as input by the next sub-module.

2.1.2 HTML Parser

Taking as input a cleaned web page, the HTML Parser sub-module extracts the paragraphs from the web page by extracting all the paragraph tags. Sentences are also extracted from each paragraph through a sentence chunker. Each paragraph with its associated sentences are then stored for processing by the succeeding module.
2.2 Extractor Module
The Extractor Module is responsible for the task of extracting information from text retrieved from web pages. The Extractor Module is composed of two sub-modules which are (1) the Concept Recognizer and (2) the Association Recognizer. It takes in as input a paragraph as well as its sentences and outputs a collection of assertions extracted from the sentences.

2.2.1 Concept Recognizer
Concepts and entities are essential parts of a knowledge base. The Concept Recognizer sub-module extracts concepts as well as entities from sentences which will make up the nodes in the resulting knowledge base.

2.2.2 Association Recognizer
Associations between concepts and entities are also an important part of a knowledge base. Potential assertions between extracted concepts are discovered and extracted from sentences.

2.3 Builder Module
The Builder Module is responsible for the task of building a context graph for each paragraph by connecting together the extracted information from the preceding module through the analysis of paragraph features. The Builder Module is composed of two sub-modules which are (1) the Coreferencer and (2) the Graph Builder. It takes as input a paragraph as well as its sentences and the extracted information from the preceding module. Its output is a context graph representing the information reflected by the input paragraph.

2.3.1 Coreferencer
Before creating a context graph for a paragraph, concepts/entities taking the form of pronouns must be resolved first to their appropriate references. The Coreferencer sub-module performs this task on both sentence level and paragraph level, replacing extracted pronouns into their appropriate references.

2.3.2 Graph Builder
Using the extracted concepts/entities as well as the associations between them, the Graph Builder sub-module links them together into a context graph. Extracted information are linked together through the analysis of sentence and paragraph features. Resulting context graphs are then stored for processing by the next module.

2.4 Resolver Module
The Resolver Module is responsible for linking context graphs together forming the resulting knowledge base. Graphs are compared wherein similar graphs are merged such as merging two graphs identified to be pertaining to the same context.

3. PRELIMINARY RESULTS
A system prototype was developed during the course of this research following the architecture mentioned in the previous section. The following subsections discuss the implementation of the architecture in detail.

3.1 Techniques for extraction
During the implementation of the Extractor module, two features were considered to be used namely (1) POS tags and (2) Typed dependencies. Different techniques are applied for each feature with each having its own advantages and disadvantages.

3.1.1 POS tags
The system is able to get the POS tags within a given sentence through the use of a POS tagger and a phrase chunker. To extract information through POS tags, the system simply maps a certain POS to an element in the knowledge base (i.e. Concept, Relation). Table 1 shows the different mappings that were initially used in extraction. For example, given the sentence “Radar is an object-detection system”, the system will map the NP “Radar” as a Concept, the PP “is” as a Relation and the NP “an object-detection system” as another Concept.
4. CONCLUSION AND FINAL REMARKS
In terms of performance in extracting information, the use of dependencies has been more effective as compared to the use of POS tags. Dependencies provide much more information that can be used for extraction as compared to what POS tags do.

Further works considered in developing the system include adding more extraction rules using dependencies. For the task of co-reference, information provided by dependencies also show potential use. Roles played by different words as expressed by dependencies can be used in co-reference such as the role of being the subject of a sentence. Co-referencing techniques can take advantage of such information. For example, a subject that is a pronoun is more likely to refer to the subject of the previous sentence.

5. Bibliography