**Comparable Entity Mining from Comparative Questions**

**LITERATURE SURVEY**

**Relational Learning of Pattern-Match Rules for Information Extraction**

Information extraction systems process natural language documents and locate a specific set of relevant items. Given the recent success of empirical or corpus-based approaches in other areas of natural language processing, machine learning has the potential to significantly aid the development of these knowledge-intensive systems. This paper presents a system, Rapier that takes pairs of documents and filled templates and induces pattern-match rules that directly extract fillers for the slots in the template. The learning algorithm incorporates techniques from several inductive logic programming systems and learns unbounded patterns that include constraints on the words and part of-speech tags surrounding the filler. Encouraging results are presented on learning to extract information from computer job postings from the newsgroup misc.jobs.offered.

The ability to extract desired pieces of information from natural language texts is an important task with a growing number of potential applications. Tasks requiring locating specific data in newsgroup messages or web pages are particularly promising applications. Manually constructing such information extraction systems is a laborious task; however, learning methods have the potential to help automate the development process. The Rapier system described in this paper uses relational learning to construct unbounded pattern-match rules for information extraction given only a database of texts and filled templates. The learned patterns employ limited syntactic and semantic information to identify potential slot fillers and their surrounding context. Results on extracting information from newsgroup jobs postings have shown that for one realistic application, fairly accurate rules can be learned from relatively small sets of examples. Future research will hopefully demonstrate that similar techniques will prove useful in a wide variety of interesting applications.

**Collective Information Extraction with Relational Markov Networks**

Most information extraction (IE) systems treat separate potential extractions as independent. However, in many cases, considering influences between different potential extractions could improve overall accuracy. Statistical methods based on undirected graphical models, such as conditional random fields (CRFs), have been shown to be an effective approach to learning accurate IE systems. We present a new IE method that employs Relational Markov Networks (a generalization of CRFs), which can represent arbitrary dependencies between extractions. This allows for collective information extraction" that exploits the mutual influence between possible extractions. Experiments on learning to extract protein names from biomedical text demonstrate the advantages of this approach.

We have presented an approach to collective information extraction that uses Relational Markov Networks to reason about the mutual influences between multiple extractions. A new type of clique template the logical OR template was introduced, allowing a variable number of relevant entities to be used by other clique templates. Soft correlations between repetitions and acronyms and their long form in the same document have been captured by global clique templates, allowing for local extraction decisions to propagate and mutually influence each other.

**Identifying Comparative Sentences in Text Documents**

This paper studies the problem of identifying comparative sentences in text documents. The problem is related to but quite different from sentiment/opinion sentence identification or classification. Sentiment classification studies the problem of classifying a document or a sentence based on the subjective opinion of the author. An important application area of sentiment/opinion identification is business intelligence as a product manufacturer always wants to know consumers’ opinions on its products. Comparisons on the other hand can be subjective or objective. Furthermore, a comparison is not concerned with an object in isolation. Instead, it compares the object with others. An example opinion sentence is “the sound quality of CD player X is poor”. An example comparative sentence is “the sound quality of CD player X is not as good as that of CD player Y”. Clearly, these two sentences give different information. Their language constructs are quite different too. Identifying comparative sentences is also useful in practice because direct comparisons are perhaps one of the most convincing ways of evaluation, which may even be more important than opinions on each individual object. This paper proposes to study the comparative sentence identification problem. It first categorizes comparative sentences into different types, and then presents a novel integrated pattern discovery and supervised learning approach to identifying comparative sentences from text documents. Experiment results using three types of documents, news articles, consumer reviews of products, and Internet forum postings, show a precision of 79% and recall of 81%. More detailed results are given in the paper.

This paper proposed the study of identifying comparative sentences. Such sentences are useful in many applications, e.g., marketing intelligence, product benchmarking, and e-commerce. We first analyzed different types of comparative sentences from both the linguistic point of view and the practical usage point of view, and showed that existing linguistic studies have some limitations. We then made several enhancements. After that we proposed a novel rule mining and machine learning approach to identifying comparative sentences. Empirical evaluation using diverse text data sets showed its effectiveness.

**Mining Comparative Sentences and Relations**

This paper studies a text mining problem, comparative sentence mining. A comparative sentence expresses an ordering relation between two sets of entities with respect to some common features. For example, the comparative sentence “Canon’s optics are better than those of Sony and Nikon” expresses the comparative relation: (better, {optics}, {Canon}, {Sony, Nikon}). Given a set of evaluative texts on the Web, e.g., reviews, forum postings, and news articles, the task of comparative sentence mining is (1) to identify comparative sentences from the texts and (2) to extract comparative relations from the identified comparative sentences. This problem has many applications. For example, a product manufacturer wants to know customer opinions of its products in comparison with those of its competitors. In this paper, we propose two novel techniques based on two new types of sequential rules to perform the tasks. Experimental evaluation has been conducted using different types of evaluative texts from the Web. Results show that our techniques are very promising.

This paper studied the new problem of identifying comparative sentences in evaluative texts, and extracting comparative relations from them. Two techniques were proposed to perform the tasks, based on class sequential rules and label sequential rules, which give us syntactic clues of comparative relations. Experimental results show that these methods are quite promising.

**Semantic Class Learning from the Web with Hyponym Pattern Linkage Graphs**

We present a novel approach to weakly supervised semantic class learning from the web, using a single powerful hyponym pattern combined with graph structures, which capture two properties associated with pattern-based extractions: popularity and productivity. Intuitively, a candidate is popular if it was discovered many times by other instances in the hyponym pattern. A candidate is productive if it frequently leads to the discovery of other instances. Together, these two measures capture not only frequency of occurrence, but also cross-checking that the candidate occurs both near the class name and near other class members. We developed two algorithms that begin with just a class name and one seed instance and then automatically generate a ranked list of new class instances. We conducted experiments on four semantic classes and consistently achieved high accuracies.

Combining hyponym patterns with pattern linkage graphs is an effective way to produce a highly accurate semantic class learner that requires truly minimal supervision: just the class name and one class member as a seed. Our results consistently produced high accuracy and for the states and countries categories produced very high recall. The singers and fish categories, which are much larger open classes, also achieved high accuracy and generated many instances, but the resulting lists are far from complete. Even on the web, the doubly anchored hyponym pattern eventually ran out of steam and could not produce more instances. However, all of our experiments were conducted using just a single hyponym pattern. Other researchers have successfully used sets of hyponym patterns (e.g., (Hearst, 1992; Etzioni et al., 2005; Pas¸ca, 2004)), and multiple patterns could be used with our algorithms as well. Incorporating additional hyponym patterns will almost certainly improve coverage, and could potentially improve the quality of the graphs as well. Our popularity-based algorithm was very effective and is practical to use. Our best-performing algorithm, however, was the 2-step process that begins with an exhaustive search (reckless bootstrapping) and then ranks the candidates using the Out degree scoring function, which represents productivity. The first step is expensive, however, because it exhaustively applies the pattern to the web until no more extractions are found. In our evaluation, we ran this process on a single PC and it usually finished overnight, and we were able to learn a substantial number of new class instances. If more hyponym patterns are used, then this could get considerably more expensive, but the process could be easily parallelized to perform queries across a cluster of machines. With access to a cluster of ordinary PCs, this technique could be used to automatically create extremely large, high-quality semantic lexicons, for virtually any categories, without external training resources.

**Comparable Entity Mining from Comparative Questions**

Comparing one thing with another is a typical part of human decision making process. However, it is not always easy to know what to compare and what are the alternatives. To address this difficulty, we present a novel way to automatically mine comparable entities from comparative questions that users posted online. To ensure high precision and high recall, we develop a weakly-supervised bootstrapping method for comparative question identification and comparable entity extraction by leveraging a large online question archive. The experimental results show our method achieves F1- measure of 82.5% in comparative question identification and 83.3% in comparable entity extraction. Both significantly outperform an existing state-of-the-art method.

In this paper, we present a novel weakly supervised method to identify comparative questions and extract comparator pairs simultaneously. We rely on the key insight that a good comparative question identification pattern should extract good comparators, and a good comparator pair should occur in good comparative questions to bootstrap the extraction and identification process. By leveraging large amount of unlabeled data and the bootstrapping process with slight supervision to determine four parameters, we found 328,364 unique comparator pairs and 6,869 extraction patterns without the need of creating a set of comparative question indicator keywords.

**Mining Knowledge from Text Using Information Extraction**

An important approach to text mining involves the use of natural-language information extraction. Information extraction (IE) distills structured data or knowledge from unstructured text by identifying references to named entities as well as stated relationships between such entities. IE systems can be used to directly extricate abstract knowledge from a text corpus, or to extract concrete data from a set of documents which can then be further analyzed with traditional data-mining techniques to discover more general patterns. We discuss methods and implemented systems for both of these approaches and summarize results on mining real text corpora of biomedical abstracts, job announcements, and product descriptions. We also discuss challenges that arise when employing current information extraction technology to discover knowledge in text.

In this paper we have discussed two approaches to using natural-language information extraction for text mining. First, one can extract general knowledge directly from text. As an example of this approach, we reviewed our project which extracted a knowledge base of 6,580 human protein interactions by mining over 750,000 Medline abstracts. Second, one can first extract structured data from text documents or web pages and then apply traditional KDD methods to discover patterns in the extracted data. As an example of this approach, we reviewed our work on the DiscoTEX system and its application to Amazon book descriptions and computer-science job postings and resumes. Research in information extraction continues to develop more effective algorithms for identifying entities and relations in text. By exploiting the lastest techniques in human-language technology and computational linguistics and combining them with the latest methods in machine learning and traditional data mining, one can effectively mine useful and important knowledge from the continually growing body of electronic documents and web pages.

**Object-Level Ranking: Bringing Order to Web Objects**

In contrast with the current Web search methods that essentially do document-level ranking and retrieval, we are exploring a new paradigm to enable Web search at the object level. We collect Web information for objects relevant for a specific application domain and rank these objects in terms of their relevance and popularity to answer user queries. Traditional PageRank model is no longer valid for object popularity calculation because of the existence of heterogeneous relationships between objects. This paper introduces PopRank, a domain-independent object-level link analysis model to rank the objects within a specific domain. Specifically we assign a popularity propagation factor to each type of object relationship, study how different popularity propagation factors for these heterogeneous relationships could affect the popularity ranking, and propose efficient approaches to automatically decide these factors. Our experiments are done using 1 million CS papers, and the experimental results show that PopRank can achieve significantly better ranking results than naively applying PageRank on the object graph.

This paper studies how to calculate the object popularity scores of Web objects based on their Web popularity and the object relationship graph. Traditional PageRank algorithms are no longer valid because of the existence of heterogeneous relationships between objects. We propose to automatically assign a popularity propagation factor for each type of object relationship. Specifically the contributions of the paper are: (i) A PopRank model which considers both the Web popularity of an object and the object relationship graph to calculate the PopRank score of the Web object; (ii) An automated approach for assigning popularity propagation factors for different types of object relationships using partial ranking lists from domain experts. We propose to use a subgraph of the entire object relationship graph to efficiently search for good propagation factors; (iii) The experiments are done in the context of Libra, an object-level Web search prototype indexing 1 million papers.

**Probabilistic Question Answering on the Web**

Web-based search engines such as Google and NorthernLight return documents that are relevant to a user query, not answers to user questions. We have developed an architecture that augments existing search engines so that they support natural language question answering. The process entails five steps: query modulation, document retrieval, passage extraction, phrase extraction, and answer ranking. In this article, we describe some probabilistic approaches to the last three of these stages. We show how our techniques apply to a number of existing search engines, and we also present results contrasting three different methods for question answering. Our algorithm, probabilistic phrase reranking (PPR), uses proximity and question type features and achieves a total reciprocal document rank of .20 on the TREC8 corpus. Our techniques have been implemented as a Web-accessible system, called NSIR.

We presented a probabilistic method for Web-based Natural Language Question Answering. It has been implemented in a robust system and has been tested on a realistic corpus of questions. One thing we didn’t address in this article is the scalability issue. Even though the current system performs relatively faster than other Web-based question answering systems, the current system’s performance for real-time question answering remains to be improved. One thing that deserves further attention is that, after extensive testing, we found that many preprocessing steps such as page downloading, sentence segmentation, part of speech tagging, and so on, take most of the response time. Even though parallel processing can be used to speed up the downloading phase, the dependence of existing Web search engines as answer sources is really the bottleneck of our system. We expect to improve performance significantly by using a prebuilt snapshot of a search engine’s content. NSIR currently takes between 5 and 30 seconds per question depending on the (user-specified) number of documents to be downloaded from the Web and on the (again userspecified) number of phrases to extract. The current version of NSIR doesn’t include query modulation (Radev et al., 2001b; the process of converting a question to the best query for a given search engine).

**Learning Subjective Nouns using Extraction Pattern Bootstrapping**

We explore the idea of creating a subjectivity classifier that uses lists of subjective nouns learned by bootstrapping algorithms. The goal of our research is to develop a system that can distinguish subjective sentences from objective sentences. First, we use two bootstrapping algorithms that exploit extraction patterns to learn sets of subjective nouns. Then we train a Naive Bayes classifier using the subjective nouns, discourse features, and subjectivity clues identified in prior research. The bootstrapping algorithms learned over 1000 subjective nouns, and the subjectivity classifier performed well, achieving 77% recall with 81% precision.

This research produced interesting insights as well as performance results. First, we demonstrated that weakly supervised bootstrapping techniques can learn subjective terms from unannotated texts. Subjective features learned from unannotated documents can augment or enhance features learned from annotated training data using more traditional supervised learning techniques. Second, Basilisk and Meta-Bootstrapping proved to be useful for a different task than they were originally intended. By seeding the algorithms with subjective words, the extraction patterns identified expressions that are associated with subjective nouns. This suggests that the bootstrapping algorithms should be able to learn not only general semantic categories, but any category for which words appear in similar linguistic phrases. Third, our best subjectivity classifier used a wide variety of features. Subjectivity is a complex linguistic phenomenon and our evidence suggests that reliable subjectivity classification requires a broad array of features.