SNExtractor: A Prototype for Extracting Semantic Networks from Web Documents

Chi Zhang(✉), Yanhua Wang, Chengyu Wang, Wenliang Cheng, and Xiaofeng He

Institute for Data Science and Engineering, East China Normal University, Shanghai, China
{51131500049,51141500045}@ecnu.cn

Abstract. Algorithms for extracting entities and relations from the Web heavily rely on semi-structured data sources or large-scale Web pages. It is difficult to extend these techniques to arbitrary Web documents in different domains. In this demonstration, we present SNExtractor, a prototype system for extracting semantic networks from documents related to any hot topics on the Web. Given a user query, it provides a comprehensive overview of relevant documents, including entities, a semantic network and a timeline summary. In the following, we will present internal mechanisms of SNExtractor and its application in the education domain.

1 Introduction

In recent years, the problem that how to harvest knowledge from the Web has attracted lots of attention. In knowledge graphs, the facts are either extracted from semi-structured data sources (in YAGO and DBPedia), or textual patterns based on the redundancy of Web data (in Probase and NELL). However, when it comes to a hot topic in a specific domain, only a (relatively small) collection of relevant Web documents can be retrieved from the Web. It is difficult to use existing techniques to extract knowledge from these documents.

Therefore, the natural question is *how to automatically extract semantic networks from Web documents related to any topics*. However, there are still several challenges to this work, illustrated as follows: (i) Entities in Web documents are appeared in the form of free text, and need to be recognized and normalized. (ii) The low redundancy nature of Web documents in a specified domain makes it difficult to apply traditional relation extraction methods. (iii) Additionally, given a query topic, a user is better served by a timeline summary of the topic, rather than having a collection of relevant documents.

In this demonstration, we present SNExtractor, a prototype system for extracting semantic networks from Web documents. SNExtractor has three novel features. (i) To improve the performance of NER, it employs a statistical method to remove noisy entities. It uses sub-string matching and entity disambiguation to resolve entity normalization. (ii) It detects potential relation instances based on linguistic and statistical features. In order to label the extracted relations,

B. Cui et al. (Eds.): WAIM 2016, Part II, LNCS 9659, pp. 527–530, 2016.
DOI: 10.1007/978-3-319-39958-4

we use keywords extracted from context with pre-defined patterns. (iii) For online summary generation, we detect the different stages of a hot topic by topic drift, and generate a timeline as the topic summary.

There are two parts in our demonstration. The first part is the backend of the system, including offline data analysis and online query processing modules. In the second part, we present an application based on SNExtractor for educational data analysis, which automatically collects and processes education-related Web pages and provides overviews of hot educational topics in China.

2 System Overview

As Fig. 1 illustrates, SNExtractor contains two major parts: (i) offline knowledge acquisition and (ii) online query processing. Given a user query, SNExtractor returns a semantic network relevant to the query with entities and relations, and a timeline as the summarization of relevant documents as output. The semantic network is extracted offline and stored in a knowledge repository. The timeline is generated online by detecting the topic drift phenomenon from the result of the topic modeling computed offline.

In the offline module, the data collector consists several distributed Web crawlers. Each crawler crawls Web pages of a certain website, parses the page structures, and stores the semi-structured data contents in databases. There are two stages after the data collection step. (i) We utilize NER techniques to recognize entities of various types in documents. Additionally, to improve the data quality of entities, we normalize entities by mapping surface forms to unambiguous references. Semantic relations are extracted and labeled with keywords of the context. (ii) We employ the Latent Dirichlet Allocation (LDA) to generate topic distributions for all the documents.

In the online part, a collection of relevant documents w.r.t a user query are returned by a search engine. Entities and semantic relations extracted from these documents are retrieved from the knowledge repository to form the semantic network. The timeline is generated online according to topic distributions of these documents.

In the following, we discuss our implementation details coping with some major challenges in SNExtractor.

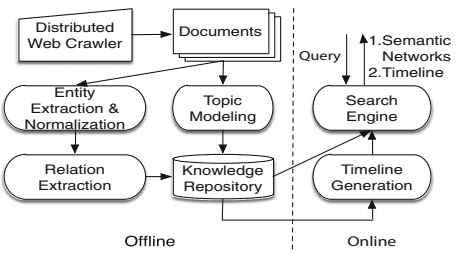

Fig. 1. System workflow of SNExtractor

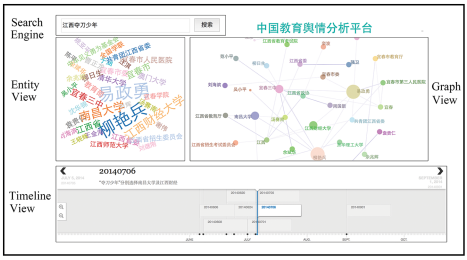

Fig. 2. Screenshot of educational public opinion analysis platform

Entity Extraction and Normalization. Given a collection of unnormalized entities M recognized by NER models, we filter out noisy or incorrect entities. We design a mapping function $f : m \rightarrow e$ such that for each entity m in the remaining entity set $M^{'}$ ($M^{'} \subseteq M$), it maps m to its normalized, unambiguous form e. The techniques for entity normalization include sub-string matching and entity disambiguation, introduced in [1].

Semantic Relation Extraction. Web documents related to a certain topic and in a specific domain are relatively sparse, making it hard to apply traditional pattern-based relation extraction methods. To solve this problem, SNExtractor analyzes linguistic and statistical features to identify candidate relation tuples [2], in the form of $(e_i, e_j, C_{i,j})$, where e_i and e_j are normalized entities, and $C_{i,j}$ are the contexts of e_i and e_j. In order to label the extracted candidate relations, we cluster entity pairs which have similar contexts together as a raw relation, i.e., $R = \{(e_i, e_j, C_{i,j})\}$. The keywords for the raw relation R are labeled by extracting the frequent keywords in $C_{i,j}$ for all e_i and e_j pairs.

Online Timeline Generation. We observe that when a hot topic is reported online, the issue of topic drift arises in these news articles, which gives us a signal to detect different stages of the topic in an efficient manner. For two documents d_i and d_j, the topic drift $t(i, j)$ can be measured by the change in topic distributions θ_i and θ_j, such as self-normalized KL divergence [3]. Given a collection of documents D, we order them chronically, based on publication time. We select top-k documents $D^{'} \subseteq D$ that indicates the drift of topics. The headlines of documents in $D^{'}$ are taken as the timeline.

3 Demonstration Scenario

Our demonstration includes the following two parts:

SNExtractor Backend. We will show how Web data are collected, processed, analyzed and managed in the system. The following three components will be illustrated: (i) collecting and processing data in a distributed environment, (ii) extraction of semantic network, and (iii) process of query processing, including relevant document search, semantic network retrieval and timeline generation.

SNExtractor Application. In this part, we will show an application of SNExtractor for educational public opinion analysis in China. A screenshot of this application is shown in Fig. 2.

Acknowledgment. This work is partially supported by Shanghai Agriculture Science Program (2015) Number 3-2.

References

1. Jijkoun, V., Khalid, M.A., Marx, M., de Rijke, M.: Named entity normalization in user generated content. In: AND 2008, pp. 23–30 (2008)
2. Shen, W., Wang, J., Luo, P., Wang, M., Yao, C.: REACTOR: a framework for semantic relation extraction and tagging over enterprise data. In: WWW 2011, pp. 121–122 (2011)
3. Knights, D., Mozer, M.C., Nicolov, N.: Detecting topic drift with compound topic models. In: ICWSM 2009 (2009)