Graphia: Extracting Contextual Relation Graphs from Text

Danilo S. Carvalho[1], André Freitas[3], and João C.P. da Silva[2]

[1] PESC/COPPE
[2] Computer Science Department
Universidade Federal do Rio de Janeiro (UFRJ), Brazil
[3] Digital Enterprise Research Institute (DERI)
National University of Ireland, Galway, Ireland

Abstract. This demo presents *Graphia*, an information extraction pipeline targeting an RDF representation of unstructured data in the form of structured discourse graphs (SDGs). It combines natural language processing and information extraction techniques with the use of linked open data resources and semantic web technologies to enable discourse representation as a set of contextualized relationships between entities.

Keywords: Open Information Extraction, Linked Open Data, Natural Language Processing.

1 Introduction

The Linked Data Web brings the vision of a Web-scale semantic data graph layer which can improve the ability of users and systems to access and semantically interpret information. Most of the information available on the Web today is in an unstructured text format. The integration of this information into the Linked Data Web is a fundamental step towards enabling the Semantic Web vision. The semantics of unstructured text, however, does not easily fit into structured datasets. The representation of information extracted from texts needs to take into account large terminological variation, complex context patterns, fuzzy and conflicting semantics and intrinsically ambiguous sentences.

Typical information extraction (IE) approaches for extracting relations from unstructured text have either focused on the extraction of simple relations (triples) or on specific patterns which are going to feed a well structured ontology (e.g. events), scenarios where accuracy, consistency and a high level of lexical and structural normalization are primary concerns. The purpose of Graphia is to show that these IE approaches can be complemented by alternative information extraction scenarios where accuracy, consistency and regularity are traded by domain-independency, context capture, wider extraction scope and maximization of the text's semantic dependencies representation, where data semantics and data quality can be improved over time.

The demo focuses on the contextual capture gain that can be attained by the combination of linguistic information, linked open data and a flexible and extensible graph relation representation model. The differences between Graphia and

P. Cimiano et al. (Eds.): ESWC 2013, LNCS 7955, pp. 236–241, 2013.

other relation extraction systems are examined in the light of these motivations, and the extraction workflow is described. We also look into the entity-centric integration of the extracted graphs into the existing Linked Data Web.

2 Overview

Graphia is an application for graph relation extraction from natural language text, meaning that it takes factual text as input and produces entity-centric data graphs as output. Each node of the graph represents an entity (named or non-named) and the edges indicate the relations (e.g. actions, locations, pertinence) between the nodes.

The main difference in the Graphia relation extraction, when compared to other relation extraction tools, is its ability of capturing relations that are not in the scope of simple triples, either as a triple context or entity context. To accommodate the contextual representation, a principled entity-centric structured data graph (SDG) representation and interpretation model is introduced [1]. Figure 1, Figure 2 and Figure 3 depict examples of Graphia's SDG output.

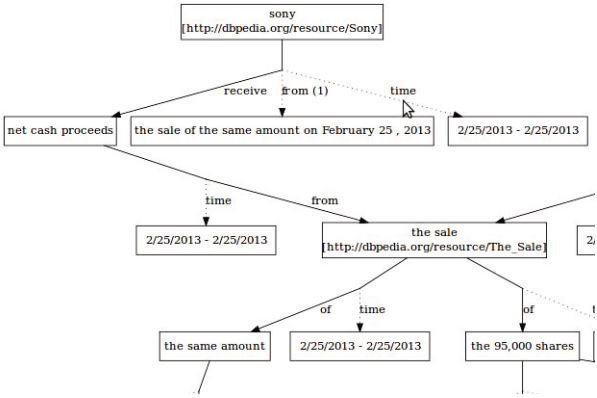

Fig. 1. Fragment of a graph extracted from a sentence on a *sec.gov* report: *. . . Sony will receive net cash proceeds from the sale of the same amount on February 25, 2013.* Nodes represent *entities*, filled edges represent *relations* and dotted edges are *reifications* (i.e. context of relations).

The most common way of expressing relations in a information extraction system is the *subject-predicate-object (s,p,o) triple*, where the predicate expresses the relation between the subject and the object. Graphia builds on top of this representation model by using SDGs. The use of SDGs supports a representation which focuses on the capture of complex contextual dependencies without commiting to a specific conceptual model. The system is designed to make use of syntactic information where possible, and helps to bridge the terminological, context, and semantic gap by connecting this information with the entities offered by linked open data such as DBpedia [2].

3 Demonstration and Workflow Description

Graphia is implemented as an open information extraction pipeline, in which each step does a well defined task and makes available a set of new data to the next step. The sequence of steps is described next, with the output for the sample sentence "*John Doe is a very important person in USA and he was born in the UK.*":

Syntactic Analysis: The first step in the extraction process is the syntactic parsing of the natural language text into syntactic trees (C-Structures). This component uses the Probabilistic Context-Free Grammar (PCFG) implemented in the Stanford parser [3]. The C-Structures for the sentences are passed to the next components.

Named Entity Resolution: This component resolves named entities text references to existing DBpedia URIs. The first step consists in the use of the DBpedia Spotlight service[1] where the text is sent and is returned annotated with URIs. The second step consists in the use of Part-of-Speech tags together with C-Structures to aggregate words into entity candidates which were not resolved by the DBpedia Spotlight service. The entity candidates' strings are resolved by using a local entity index which indexes all DBpedia URIs using TF/IDF over labels extracted from the URIs, and validated through a TF-IDF threshold check. The output of this component is the original text with a set of named entity terms annotated with URIs. In this step, *John Doe*, *USA* and *the UK* gets annotated with DBpedia URIs.

Personal Co-reference Resolution and Normalization: This component resolves pronominal co-references including personal, possessive and reflexive pronouns. Personal pronouns instances are substituted by the corresponding entities. Possessive and reflexive pronouns are annotated with the corresponding entities that will later define the co-reference links. The co-reference resolution process is done by the pronoun-named entity gender and number agreement (by taking into account gender information present in a name list from the public USA census data) and position-based heuristics. The output of this component are C-Structures with annotated named entities, co-reference substitutions for personal pronouns and possessive and reflexive pronouns annotated with named entities. In this step, *He* gets substituted by the annotated *John Doe*.

Graph Extraction: The graph extraction module takes as input the annotated C-Structures and generates the triple trees for each sentence by the application of a set of ten manually designed transformation rules based on syntactic conditions through a DFS traversal of the C-Structure. Instead of focusing on terminology-dependent patterns, these rules are based on syntactic patterns. In this step,

[1] `http://dbpedia.org/spotlight`

two graphs are extracted, as the sentence can be divided by the coordinating conjunction *and*: one centered on the relation *is* and the other on the relation *was born*.

Graph Construction: This component receives the triple trees from the previous component and outputs the final graph serialization. Additionally, local URIs are created for each resource which was not resolved to a DBpedia URI. The output is depicted in Figure 2.

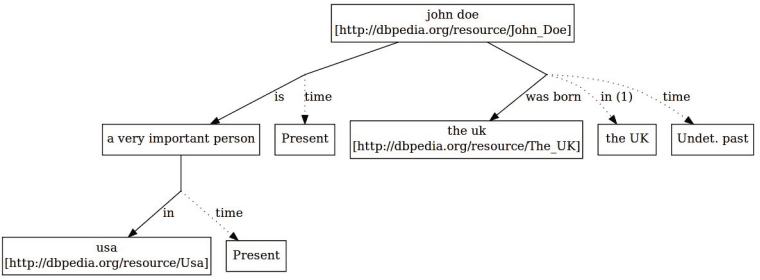

Fig. 2. Example of extracted graph

The Demonstration[2,3] of the application running can be found on the web.

4 Related Applications

The three applications most closely related to Graphia are ReVerb [4] [4], the relation extraction module of Alchemy API [5] and FRED[6] [5]. ReVerb algorithms are able to extract long relations, like the relation between Neil Armstrong and the Moon: "the first man to walk on", while Alchemy API can only extract short (verbal) relations. Both focus on simple binary relations, generating a list of (s,p,o) triples. FRED produces RDF/OWL ontologies and linked data from natural language sentences, sharing the goal of integrating unstructured data with open web ontologies but focusing on fitting the text data on a ontology model and discarding what doesn't fit. In contrast, Graphia tries to maximize the amount of extracted information, mapping contextual dependencies such as location, dates and quantities for the relations, and outputting graphs that admit the transformations: a) edges to simple relations and b) edge paths to long relations.

[2] `http://graphia.dcc.ufrj.br`
[3] `http://graphia.dcc.ufrj.br/eswcdemo`
[4] `http://reverb.cs.washington.edu`
[5] `http://www.alchemyapi.com`
[6] `http://wit.istc.cnr.it/stlab-tools/fred`

A extraction example for the three applications is shown on Figure 3.

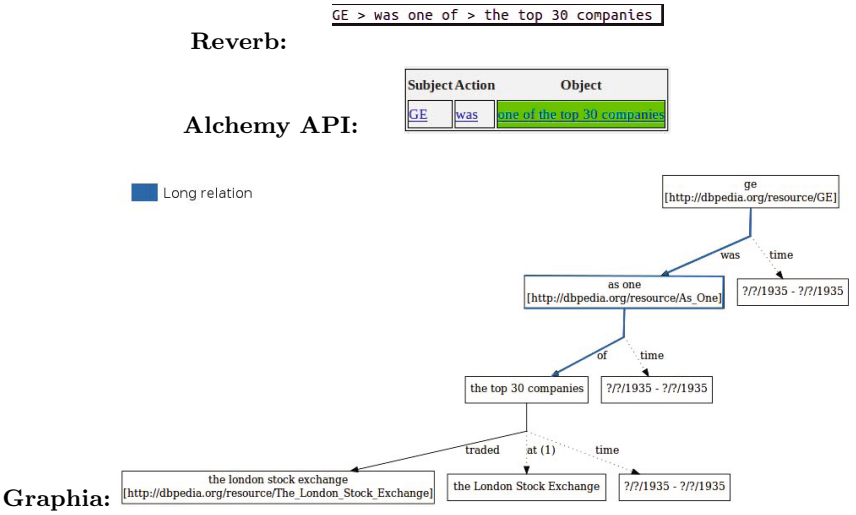

Fig. 3. Example of extractions on ReVerb, Alchemy API and Graphia for the sentence "In 1935, GE was one of the top 30 companies traded at the London Stock Exchange"

5 Conclusions and Future Work

Graphia builds on top of existing open information extraction technologies, offering more contextualized relation extraction results. The goal of enabling discourse representation as a set of contextualized relationships between entities is attained by the combination of linguistic information, linked open data and a flexible and extensible representation model. Broadening context capture is the main point of improvement for future work. Another major improvement is the inclusion of subordinate sentences. Currently integration with existing Linked Data resources is focused on instances and classes, by means of entities alignment with DBpedia.

References

1. Freitas, A., Carvallho, D.S., da Silva, J.C.P., O'Riain, S., Curry, E.: A Semantic Best-Effort Approach for Extracting Structured Discourse Graphs from Wikipedia. In: Proc. of the 1st Workshop on the Web of Linked Entities (WoLE) at the 11th International Semantic Web Conf., ISWC (2012)
2. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia – A Crystallization Point for the Web of Data. Journal of Web Semantics: Science, Services and Agents on the World Wide Web (7), 154–165 (2009)

3. Klein, D., Manning, C.D.: Accurate Unlexicalized Parsing. In: Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423–430 (2003)
4. Fader, A., Soderland, S., Etzioni, O.: Identifying Relations for Open Information Extraction. In: Conf. on Empirical Methods in Natural Language Processing (2011)
5. Presutti, V., Draicchio, F., Gangemi, A.: Knowledge extraction based on discourse representation theory and linguistic frames. In: ten Teije, A., Völker, J., Handschuh, S., Stuckenschmidt, H., d'Acquin, M., Nikolov, A., Aussenac-Gilles, N., Hernandez, N. (eds.) EKAW 2012. LNCS, vol. 7603, pp. 114–129. Springer, Heidelberg (2012)