Enabling HATEOAS Through HTTP OPTIONS, Link Elements, And The HTTP Vocabulary In RDF

Thomas Steiner
Universitat Politécnica de Catalunya
Department LSI
08034 Barcelona, Spain
tsteiner@lsi.upc.edu

Jan Algermissen
NORD Software Consulting
Kriemhildstrasse 7
22559 Hamburg
info@nordsc.com

ABSTRACT

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: On-line Information Services

General Terms

Experimentation

Keywords

RDF, LOD, Linked Data, Semantic Web, NLP, Video

1. INTRODUCTION

With SemWebVid [?] we introduced a client-side interactive Ajax application for the automatic generation of RDF video annotations. For this paper we have re-implemented and vastly improved the annotation logic on the server-side, resulting in a RESTful read/write-enabled Web service for RDF video annotations. A YouTube video is described by a Google Data Atom feed[1]. In order to semantically annotate the various elements of this feed, we concentrated on the following fields (in XPath syntax): title `/entry/media:group/media:title`, description `/entry/media:group/media:description`, tags `/entry/media:group/media:keywords`. YouTube offers an automatic audio transcription service and users can also upload audio transcriptions on their own. This allows for closed captions in several languages (we differentiate between subtitles and closed captions, where subtitles are hard-encoded into the video, and closed captions separate resources). In addition to the previously mentioned elements of the Google Data Atom feed, we thus use closed captions[2] when they are available.

The remainder of this paper is structured as follows:

2. CONCLUSION

3. ACKNOWLEDGMENTS

This work is partly funded by the EU FP7 I-SEARCH project (project reference 248296).

[1] E.g., `http://gdata.youtube.com/feeds/api/videos/Rq1dow1vTHY`

[2] E.g., `http://www.youtube.com/watch_ajax?action_get_caption_track_all&v=Rq1dow1vTHY`