

KAT: The K-Space Annotation Tool

Carsten Saathoff
University of Koblenz
//http://iweb.uni-koblenz.de
saathoff@uni-koblenz.de
2nd. author

Simon Schenk
University of Koblenz
sschenk@uni-koblenz.de

Ansgar Scherp
University of Koblenz
scherp@uni-koblenz.de

General Terms

Semantic Multimedia

Keywords

semantic annotation, semi-automatic annotation, efficient annotation

1. INTRODUCTION

It has been shown in numerous studies [3] that semantic annotation of multimedia content is crucial for enabling more efficient management of the growing amount of multimedia data being generated in companies, on personal desktops or in the web. The main focus of semantic multimedia research is currently the automatic annotation of multimedia content, with only few researchers focussing on semi-automatic approaches or efficient manual annotation. In this paper we present the K-Space Annotation Tool (KAT), which provides a framework for efficient and rich semantic annotation of multimedia content. The KAT is build around the Core Ontology for Multimedia (COMM) [1] as its underlying model, and offers the infrastructure to execute analyses and other services. The KAT basically provides a framework for visualising, modifying and processing COMM annotations. It is meant as a framework for researchers, providing the means to easily build prototypes that integrate different approaches for multimedia annotation and management. It is available as open source from <https://launchpad.net/kat>.

2. ARCHITECTURE

An overview of the KAT architecture is depicted in Figure 1. As one can see from the figure, the KAT core provides an API and a set of services that might be used by plugins, which implement all the end-user relevant functionality, such as processing of content, retrieval, or visualisation. The functionality provided by the core is the GUI Layouter and Tools that provide the means to register and display different views for user interaction, the PluginManager, which controls import, loading, removal and execution of plugins, the RDF

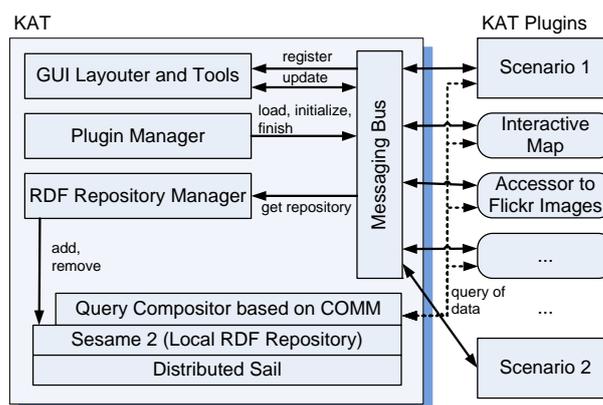


Figure 1: The KAT architecture overview.

Repository Manager, which controls initialisation, configuration and access to the underlying Sesame RDF repository¹, and finally the Sesame2 RDF repository itself configured with a distributed Networked Graphs Sail [4], which allows for the transparent access to remote repositories, while also providing local metadata storage.

A plugin implements the functionality that is relevant to an end-user, i.e. the analysis, visualisation, modification or retrieval of content and its metadata. The core component of a plugin is the controller, the only class that is directly instantiated by the KAT core. Further, plugins might register views that are displayed in the GUI and that are responsible for displaying content and metadata and for user interaction. Complex analyses can be encapsulated in workers that are executed by the KAT core sequentially in order to not overload the computer.

The model and storage layer of the KAT is based on the COMM and the Sesame2 RDF repository in conjunction with the Networked Graphs Sail. The common model for all components is the COMM itself, and communication among components is based on passing URIs of COMM objects around. The semantics of the URI is clearly defined by the COMM based on the available metadata in the repository, and in conjunction with the type of the message a component can clearly identify what should be done with the content item or COMM graph, respectively. The Networked Graphs Sail is used to structure the RDF repository into

¹<http://www.openrdf.org>

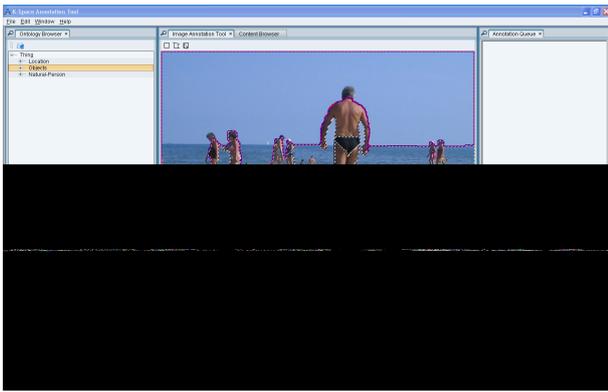


Figure 2: Screenshot of an analysed image with indicated regions.

easier to handle parts, to allow for the addition of provenance and administrative metadata, and to provide transparent read and write access to more than one underlying repository at once.

3. SCENARIO 1: SEMI-AUTOMATIC LABELLING OF IMAGE REGIONS

Automatic annotation of images is still an active research area. One known problem is the identification and labelling of image regions, which allows for exploitation of features such as spatial relations. Region-level annotations are e.g. known to be helpful for boosting global classification of images. Most approaches assume that classifiers are trained a priori with a large amount of training data, although one knows that training generic classifiers for a large amounts of concepts is a hard and yet unsolved problem. We therefore focus on classifiers that can be trained with low amounts of training examples. We have integrated one approach into the KAT which allows for training classifiers for new concepts by annotating respective regions of images, and to provide feedback to the classifier by correcting wrongly detected concepts. A screenshot is depicted in Figure 3.

4. SCENARIO 2: BROWSING AND ANNOTATION WITH LINKED DATA

Traditional tagging systems provide easy means to annotate content, but only weak semantics. Linked data provides concepts and individuals that might be used for annotation, and provides richer semantics and interlinking with related concepts. Annotation with linked data in an open infrastructure in the Semantic Web provides an added-value to the users as it gives them a tool at hand to browse through, interact with, and visually experience the data. As a consequence, we developed an interactive application based on the KAT architecture that allows for a faceted browsing and exploration [2] of images crawled from Flickr² based on linked open data (Figure 4). The application integrates different semantic data sources such as Geonames and DBpedia, which are mapped to the Flickr images using both the geo location from the EXIF metadata and the normal tags. However, the current mapping is insufficient, and only indicates the added value of annotation with linked data. Therefore, we

²<http://flickr.com>

Figure 3: An early version of the Linked Open Data browsing and annotation interface.

extend the prototype towards an annotation interface, that allows for annotation of content using linked data. We allow for efficient annotation, by reusing the blended browsing approach. We use already available information, to propose suitable concepts for annotation. We enable ease of annotation and interchange of semantic concepts used for annotation, by allowing the user to drag and drop the appropriate concept on the content to perform an annotation.

5. CONCLUSIONS

The presented K-Space Annotation Tool provides a flexible architecture for developing applications that allow for an efficient and semantically rich annotation of multimedia content. As demonstrated, the KAT can be applied for different scenarios incorporating both automatic media analysis methods and manual semantic annotation through the users. With the KAT architecture, we contribute towards a more efficient and faster development of application-specific tools for multimedia annotation.

Acknowledgements We kindly thank our students Anton Baumesberger and Frederik Jochum for implementing the KAT architecture and its user interface. This research is partly supported by the European Commission in FP6 in the K-Space project under contract No 027026.

6. REFERENCES

- [1] R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura. COMM: designing a well-founded multimedia ontology for the web. In ISWC 2007 + ASWC 2007, pages 30{43, 2008.
- [2] M. A. Hearst. Design recommendations for hierarchical faceted search interfaces. In Proc. SIGIR 2006, Workshop on Faceted Search, pages 26{30, August 2006.
- [3] L. Hollink, T. A. Schreiber, B. J. Wielinga, and M. Worringer. Classification of user image descriptions. International Journal of Human-Computer Studies, 61(5), 2004.
- [4] S. Schenk and S. Staab. Networked graphs: a declarative mechanism for sparql rules, sparql views and rdf data integration on the web. In WWW, pages 585{594. ACM, 2008.