KES: The Knowledge Explorer System

Marcio Moreno[1], Rodrigo Santos[2], Wallas Santos[3], Renato Cerqueira[4]

IBM Research | Brazil, Av Pasteur 146 Rio de Janeiro - RJ, Brazil
{mmoreno[1], rcerq[4]}@br.ibm.com
{rodrigo.costa[2], wallas.sousa[3]}@ibm.com

Abstract. Generally, to properly perform information extraction from multimedia content, it is necessary to not only understand the media but also how they correlate with each other in time and space to compose the multimedia data. After extracting this information, it is necessary to structure and align it with ontologies and RDF repositories such as dbpedia.org to enhance quality of question answering and information retrieval. The main goal of this demo is to present a system named KES, capable of handling this scenario where a hybrid knowledge representation comes in handy. KES supports exploring and curating such representations stored in graph databases.

Keywords: Hyperknowledge; Hybrid knowledge representation; Knowledge Management; Knowledge Visualization; Knowledge Curation

1 Introduction

A considerably amount of information is structured as multimedia data (videos, images, audios, texts, etc.). How to process and understand this type of data aiming at extracting semantic information is an issue that has been faced by many research projects such as [1]. Generally, to properly perform this task, it is necessary to not only understand the media but also how they correlate with each other to compose the multimedia data. Even for isolated media data, there is the requirement of correlating the extracted information with its source in time and space.

Extracting concepts from media data is just one step in the process. It is also essential to combine these mechanisms with other knowledge engineering techniques. That is, the extracted information can be then aligned with ontologies and RDF (Resource Description Framework) [2] repositories such as dbpedia.org to enhance quality of question answering or information retrieval.

Traditional proposals for knowledge representation do not properly promote the relationship among multimedia content and abstract concepts. In general, they are designed either for representing low-level features of media data (e.g., MPEG-7 [3], Dublin Core [4], PBCore [5]) or for specifying the description of abstract concepts and semantic relations (RDF [2], OWL [6]). The former does not have means for representing high-level abstract concepts and richer media relations (causality, synchronization, etc.). The latter lacks appropriate integration with multimedia content and concept specifications.

Trying to tackle this problem, Moreno et al [7] presented a model named hyperknowledge, which supports hybrid knowledge representation. It promotes the specification of relationships among multimedia content and abstract concepts, as well as the orchestration of multimedia applications with knowledge description. This model also allows the relation of fragments of media content (called *anchors*) with concepts, giving meanings to those fragments. The foundation of hyperknowledge is the usual hypermedia concepts of nodes and links. The former represents information fragments, while the latter has the purpose of defining relationships among interfaces (anchors, ports or properties) of nodes.

In this work we present the features of a system called KES: The Knowledge Exploration System. KES was designed for exploring and managing hyperknowledge specifications, although also being capable of handling OWL and RDF. It implements a graphical representation that allows end users to collaboratively interact and visualize the information stored in a given hyperknowledge base. KES also offers to its end users features to curate knowledge, by adding, removing or editing information, creating patterns to be applied in multiple occurrences. KES uses the Hyperknowledge Platform (HP): a set of microservices designed for supporting the development of systems based on the hyperknowledge model. Thus, in this work we first present the HP architecture, and then we discuss how KES fits in it.

2 Hyperknowledge Platform Architecture

The HP has two main goals: i) to manage hyperknowledge base instances, maintaining them consistently; and ii) to support the development and to provide interoperability of services built upon this model. Figure 1 depicts the onion architecture devised for the HP.

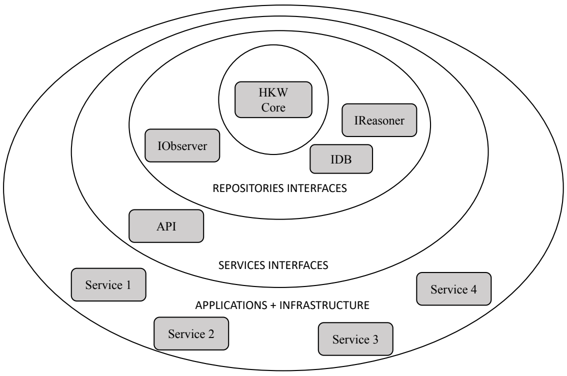

Figure 1. Hyperknowledge Platform (HP) Architecture.

The Hyperknowledge Core (*HKW Core*, in Figure 1) is the main HP component, being responsible for maintaining multiple instances of hyperknowledge bases. That is, the data structures that represent entities and relationships in a given specification. Through a well-defined API (*IDB*), different data bases can be coupled to the Core for storing hyperknowledge specifications. The *IReasoner* interface allows the integration of different reasoning engines to the Core allowing different types of inferences in the hyperknowledge base.

Applications and services in the outer-most layer communicate with the Core through a set of CRUD-like (create, retrieve, update and delete) APIs. Each modification in any hyperknowledge instance triggers a notification message to inform other services about the change. Thus, systems have to implement the *IObserver* interface to properly receive these notification messages.

We have implemented and deployed the introduced architecture as a set of composable microservices. The communication between components in the outer-most layer and the Core is performed through RESTful APIs. This simplifies the implementation of the communication layer while providing low dependency among the components. For storage and query support we are currently using the graph database JanusGraph with Gremlin.

The Core implements a *Distributed Observer Pattern* to notify services about changes in any of its managed hyperknowledge instances. It implements consistency checking and also guarantees that all services receive proper notification messages, allowing multiple systems to concurrently interact with the same Core microservice instance.

KES is a system that logically fits in the outer-most layer of the HP architecture. When a user does any modification in a hyperknowledge instance using KES, the service calls a REST endpoint to update the Core. If the user has added a multimedia content to the hyperknowledge base, the Core calls an appropriate AI Service to extract semantic information from that content. The update received by the Core and any additional information extracted from multimedia content are notified to all instances of services registered.

3 KES: The Knowledge Explorer System

KES is the first system implemented to assist end users on understanding and managing hyperknowledge specifications. It has a flexible architecture that allows different visualizations of the same knowledge specification. Additionally, users can generate new knowledge by specifying new concepts and relationships between them through a graphical interface.

KES has a frontend that follows Burkhard’s terminology [8]. That is, KES graphical visualization fits in the Concept Mapping category, which serves as a “guide to [...] an organization’s internal or external repositories of sources of information or knowledge”.

KES was devised to be a collaborative system, implementing the usual idea of sessions. In KES, a session corresponds to a shared visualization and edition of a given hyperknowledge base. All users that have joined the same session interact with a shared visualization. In practice, it means that whenever one moves a graphical entity or adds new concepts or relationships, these modifications are synchronously seen by all users in a session. The point is to allow users to have similar experiences, working together while augmenting the overall knowledge regarding a specific topic.

The system also allows its end users to import knowledge representations specified in different languages and frameworks such as OWL and RDF files. The goal is to provide interoperability with well-established knowledge description formats.

KES interface provides a search space. When an end user inputs a query, the Core processes it and delivers key elements to reasoning engines. Because of its multimedia focus, a default reasoning engine that is embedded in the HP is the support for spatio-temporal reasoning. In the demo script we will show how this can be of value for different domains, varying from Oil&Gas to entertainment.

Demo script

The main goal of this demo is to show KES main features when exploring a hybrid knowledge representation. In this sense, we defined a dataset containing seismic [1] images, concepts extracted from these images, and RDF documents to show KES in action during a knowledge exploration in the Oil&Gas domain. The demo shows the collaborative facet of the system by illustrating different users having a synchronized view of the same base. The images will be exploited to demonstrate KES capability of integrating with information extraction services as well as showing and browsing through concepts related to spatial anchors of multimedia content. The use of the RDF documents is to illustrate the conversion process from RDF to hyperknowledge. Queries using KES will demonstrate its capability of returning facts and results inferred from the existent representation. Finally, the demo shows how KES provides support for PDF document injection and visualization, as well as filtering concepts.

Demo video 1: Information extraction integration and synchronized collaboration
https://ibm.box.com/s/yompns519tmknn9ihzgbrmqevft1r7iz
Demo video 2: Importing RDF documents. Visualizing the graph according to its provenance.
https://ibm.box.com/s/4gq1nfnjc5608py9cxt27e0yvpa8qksu
Demo video 3: Injecting, visualizing, and filtering.
https://ibm.box.com/s/v2b7w6jgh12z8qgy3t2s573hia1z8hpv

References

1. Chevitarese, D. S., et al. *Deep Learning Applied to Seismic Facies Classification: a Methodology for Training. Saint Petersburg*. 2018.
2. J. Majidpour, E. Khezri, H. Hassanzade and K. S. Mohammed, "Interactive tool to improve the automatic image annotation using MPEG-7 and multi-class SVM," in Information and Knowledge Technology (IKT), 2015 7th Conference on, 2015.
3. F. A. Arakaki, P. L. V. A. da Costa and R. C. V. Alves, "Evolution of Dublin core metadata standard: an analysis of the literature from 1995-2013," 2015.
4. "PBCore 2.1," [Online]. Available: http://pbcore.org/.
5. W3C, "RDF," 2017. [Online]. Available: https://www.w3.org/RDF/.
6. W3C, "OWL," [Online]. Available: https://www.w3.org/OWL/.
7. M. F. Moreno, R. R. M. Brandão and R. Cerqueira. "Extending hypermedia conceptual models to support hyperknowledge specifications". In: IJSC. Vol. 11, Iss 01, March 2017.
8. R. Burkhard, Knowledge visualization-the use of complementary visual representations for the transfer of knowledge. A model, a framework, and four new approaches, Swiss Federal Institute of Technology (ETH Zurich), 2005.