Generation of teaching courses using ontology and stream processing

Aleksandar Pejaković, Milan Segedinac, Goran Savić

Faculty of technical science/Department, Novi Sad, Serbia

aleksandar114@gmail.com, milansegedinac@uns.ac.rs, savicg@uns.ac.rs

Abstract— **This paper proposes a new approach towards linking existing learning management systems with ontology based knowledge. Ontology is a central part of the system as it contains knowledge represented as the rules used during generation of teaching courses. Stream processing platforms are used as bridges between LMS and ontology. Stream platforms are used to query data from LMSs, process them and add insert information about available teaching materials into the ontology. With this approach existing LMSs can be extended with the possibility to generate teaching courses without modifications.**

I. Motivation

The main goal of this research was to propose a plugin for existing learning management systems, such as CanvasLMS which is used in the paper that provides the functionality of generating educational courses by using ontology and stream processing frameworks.

The proposed plugin comes with a UI that enables users to define their intended outcomes. The outcomes will serve as a starting point in the process of course generation. The plugin enables users to obtain educational resources collected from learning management systems and arranged in the appropriate order that correspond to the generated course.

II. Research questions

In traditional educational settings when creating a new educational course user had to manually combine educational resources. First, the user has to find appropriate educational resources which will lead to the achievement of stated educational objectives. After that, he or she has to sort educational resources in appropriate order, thus forming a learning path. In such an approach the appropriate educational resource are being discovered based on their name, date of creation, content, amount of time required to learn them, etc. If it is a student who designs his or her educational course without teachers help, he or she proceeds in the process of learning step by step often without knowing whether the resources are going to lead to the intended outcomes or do they have some missing preconditions.

Educational resources can be mapped to the set of educational objectives and using ontologies enables users to easily achieve such a mapping. The plugin that we propose in this paper enables educational resources to be organized automatically in accordance with the organization of the objectives thus facilitating building appropriate learning paths. The plugin does not require educational resources to be stored in one particular LMS, but allows users to form learning paths that spread throughout multiple learning management systems.

III. Methodology

The research presented in this paper consists of developing a prototype plugin that enables automatic building of learning paths for stated educational outcomes. The plugin is composed of three main modules:

- The data creation and manipulation module,
- The ontology,
- The user interface.

A. Data creation and manipulation module

The first module is made of two components. Data creation component connects to the data stores of the existing learning management systems (since we are dealing with Canvas LMS in this paper, the prototype plugin connects to the PostgreSQL database), listens to the event of creating new files and sends important data to Kafka topic. The data manipulation component is Samza job that processes incoming messages from the mentioned Kafka topic. Processed messages are stored into the plugins store (in current prototype implementation MongoDB is used).

B. Ontology

Reference [1]: "A vocabulary for exposing IEEE LOM metadata as Linked Open Data" from University of Alcala is used as base for ontology in this paper. Ontology maps IEEE LOM elements, a metadata standard for educational contents, to RDF ones based on Linked Data principles. The ontology is intended as a bridge between educational metadata and Linked Open Data (LOD) and represents the knowledge required for the creation of educational courses.

In the ontology annotated educational resource are stored, therefore issuing sparql queries, execution engine generates learning courses.

C. User interface

User interface (UI) can be implemented totally independent of rest of the pipeline, it needs to be able to issue REST calls. In this paper UI is an Angular client application through which a user can select starting and desired educational objectives and preview the educational resources proposed in the generated course. The UI also provides a page view for showing newly added resources and enables users to annotate the resources with the educational objectives.

IV.   SOLUTION

Data creation in the prototype plugin is achieved by using Kafka Connect. Kafka Connect is a stream processing framework that provides the ability to connect existing systems to Apache Kafka. It provides two operations:

- Sink – storing data from Kafka to the system,
- Source – exporting data from the system to Kafka.

Since our prototype plugin supports only Canvas LMS, and Canvas LMS uses PostgreSQL for storing all information about users, files, events and activities, in this research, we use only Kafka Connect source for PostgreSQL which already comes with Confluent kafka-connect-jdbc. Confluent kafka-connect-jdbc supplies sinks and sources for many widely used systems, and a user can easily implement their own based on existing ones if needed.

Apache Kafka is used as a messaging platform. It is distributed streaming platform that enables horizontal scaling across multiple servers. Messages that arrive in Apache Kafka topics represent the creation and upload of new files into the Canvas LMS. Each arrival of message in Apache Kafka will trigger Apache Samza jobs process method.

The Apache Samza is in charge of manipulation of incoming data. Depending on created data from Kafka Connect, Apache Samza job can enrich, modify or exclude unnecessary data. Modified messages are sent to MongoDB. Added entities in MongoDB represent unannotated files. Files that are used in the creation of new educational courses are ones that have been annotated with educational objectives.

Inside ontology stored on OpenLink Virtuoso server is stored entire logic behind creation learning paths. Plugin uses ontology as it stores all annotated materials within itself and with single SPARQL query ontology creates desired learning path.

A. Extension

To connect multiple LMS to the plugin for each LMS separate Kafka Connect job is necessary. Each LMS stores incoming materials differently, using different databases, therefore data extraction is different. Kafka Connect can be configured to extract data from different databases. As Kafka Connect only connects existing systems to Kafka, all data transformations need to be inside Apache Samza job.

Entire transformation logic can be implemented inside one Apache Samza job or divided into multiple jobs. Transformed data is stored in same MongoDB collection regardless of source LMS. Also, system treats all data the same regardless of origin.

REFERENCES

[1] "IEEE Standard for Learning Object Metadata", Electrical and Electronics Engineers Standards Association, New York, vol. 1484, pp. 21, January 2002.

[2] C. Brooks, G. McCalla, "Towards flexible learning object metadata," IJCEELL, vol. 16, February 2006.

[3] Enayat Rajabi, Miguel-Angel Sicilia, and Salvador Sanchez-Alonso, "Interlinking Educational Resources to Web of Data through IEEE LOM", Information Engineering Research Unit, Computer Science Department, University of Alcala, 28805, pp. 02, February 2015.

[4] E. Juergens, M. Felikas, M. Herrmannsdoerfer, F. Deissenboeck, R. Vaas, and K.-H. Prommer, "Feature profiling for evolving systems", Program Comprehension, IEEE 19th International Conference, June 2011.

[5] CA. El Saddik, A. Ghavam, S. Fisher, R. Steinmetz, "Metadata for Smart Multimedia Learning Objects," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].