Using EasyMiner API for Financial Data Analysis in the OpenBudgets.eu Project

Stanislav Vojíř[1], Václav Zeman[1], Jaroslav Kuchař[1,2], Tomáš Kliegr[1]

[1] University of Economics, Prague, nám. W Churchilla 4, Prague, Czech Republic
[2] Czech Technical University, Thákurova 9, 160 00, Prague, Czech Republic
first.last@{vse|fit.cvut}.cz

Abstract. This paper presents a use case for the data mining system EasyMiner in European project OpenBudgets.eu, which is concerned with publication and analysis of financial data of municipalities. EasyMiner is a web-based data mining system. This paper focuses on its new outlier detection functionality, which relies on frequent pattern mining. In addition, the system supports association rule discovery and building of rule-based classification models. The system exposes a REST API and can thus be easily integrated in third party applications.

Keywords: Association Rules, Classification, Outlier Detection, Data Mining, REST API.

1 Introduction

EasyMiner (*http://easyminer.eu*) is a web data mining system developed at the University of Economics, Prague. An earlier release was described in [1], this paper covers an updated version used for analysis of financial data in the OpenBudgets.eu project, which newly supports outlier detection based on frequent pattern mining. All data mining algorithms available in EasyMiner are also now exposed via new REST API.

This paper is organized as follows. Section 2 introduces the algorithms supported by EasyMiner. Section 3 describes the user workflow in the system. Section 4 is devoted to the OpenBudgets use case. The conclusions present a summary and an outlook for future work.

2 Algorithms

EasyMiner is a machine learning framework focusing on algorithms based on frequent pattern mining, such as association rule mining. The emphasis is on interactive web-based user interface and easily usable REST APIs.

2.1 Data Preprocessing

Algorithms based on frequent pattern mining require that the numerical data fields are discretized (binned). In some cases, the users also want to preprocess data fields with nominal values. The preprocessing methods available in EasyMiner API are *nominal enumeration*, *interval enumeration*, *equidistant intervals* and *equifrequent intervals*.

2.2 Association Rule Learning

The main algorithm used in EasyMiner to mine association rules is *Apriori* [2] as implemented in the R *Arules* package. To support larger datasets, we recently added support for *FP-Growth* [3] run on top of Spark/Hadoop. For building of classification models out of association rules, EasyMiner uses its own implementation of the *CBA* algorithm [4].

The association rule learning results are rules of the form *´antecedent → consequent´*, where antecedent and consequent are conjunctions of literals (attribute-value pairs). For classifier building, the consequent contains always one literal. The strength of the rules is described using *confidence*, *support* and optionally *lift* measures [2].

For task definition in EasyMiner, the user defines a *rule pattern*, which is a list of attributes that can appear in the antecedent and consequent of the discovered rules. The user also specifies minimum values of confidence, support and optionally lift. For classification models, the system can determine the thresholds automatically.

2.3 Outlier Detection

Newly implemented functionality for outlier detection is available as new package for the R framework called – *fpmoutliers*.[1] This package contains implementations of six existing algorithms as baselines (FPCOF, FPOF, MFPOF, WCFPOF, WFPOF, LFPOF [1]) and one innovated approach (FPI).

The results are data rows from the preprocessed dataset (bins of data attributes with concrete values) sorted using *outlier score*.

3 EasyMiner Workflow

For use of the API[2], the user must register for an account and get a unique *API key*. The API key must be sent as part of all requests send to the API. The data mining process using API is similar to the one performed in the graphical user interface. The main steps for association rule mining and outlier detection are shown on the following figure (**Fig. 1**):

[1] Available at *https://github.com/jaroslav-kuchar/fpmoutliers*.

[2] The API documentation in Swagger is available at *<easyminer-server>/easyminercenter/API*

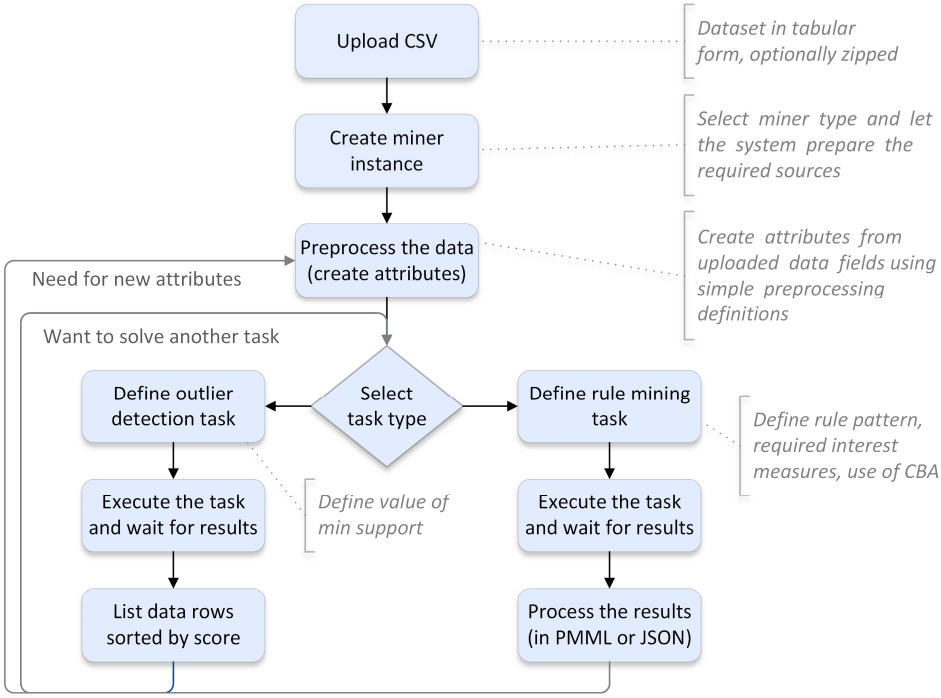

Fig. 1. Data mining process using REST API

4 Use Case: Detecting Outliers in Financial Data

European project OpenBudgets.eu[3] is a project aimed at accessing and analysing financial data and budgets of municipalities. EasyMiner is one of the systems used in the project's work package WP2: *Infrastructure for Data Collection and Mining.*

4.1 Integration of EasyMiner into OpenBudgets.eu

Based on survey of suitable algorithms for analysis of financial data, the functionality of EasyMiner has been extended with outlier detection. Several outlier detection algorithms were implemented, made available as an R package, and then integrated into the architecture of the EasyMiner system.

An important requirement for the development and customization of data mining and data analysis tools for OpenBudgets.eu users is the possibility to integrate all the tools into one, consistent platform. To meet this requirement, EasyMiner functionality had to be made available not only via a graphical user interface but also via a REST API callable by the project's integrating platform DAM[4].

[3] Project web page: *http://openbudgets.eu*

[4] Data Analysis and Mining Interface - available at: *https://github.com/openbudgets/DAM*

4.2 Analyzed Data

All financial data analyzed in the project are made available in RDF.[5] Using the data preparation tools used in the project, the collected data are converted into RDF and extended with supplementary information. For use in EasyMiner the data are converted to the CSV format using the LinkedPipes ETL software [1].

One example dataset used for data mining analysis is dataset `ESF-CZ-2007-2013`. This dataset contains information on Czech projects financed by the European Structure Funds. In the data preprocessing phase, the original data were extended using information from the *Czech Register of Economic Subjects* (ARES).[6]

The ESF dataset contains 107311 rows. The attributes used for following outlier detection example are:

- *partner_type* – type of the project partner (money recipient), according to the information from ARES
- *operational_programme* – identification number of operational program
- *amount* – amount of money allocated from operational program for the given project

4.3 Results

To illustrate the results that we obtained with outlier detection, Table 1 presents top outliers found in the ESF dataset using minimum support threshold of 0.0001. The last row in the table contains one regular (non-anomalous) instance. The high scores that the outlying instances have are caused by less frequent values of attributes associated with those instances. The low score assigned to the regular instance is a result of the fact that the attribute values describing this instance are quite frequent. For example, there are many other instances that have value "Other" in the *partner_type* attribute.

Table 1. Outliers detected using minSupport 0.0001

partner_type	operational _programme	amount	scores
Educational and research institutions	5-1	[3.34e+09,3.34e+09)	71552.57
Educational and research institutions	5-1	[3.34e+09,3.34e+09)	71552.57
Educational and research institutions	5-1	[5.96e+08,6.03e+08)	71552.57
…	…	…	…
Other	7-1	[0.00e+00,7.45e+06)	3.204

[5] RDF – Resource Description Framework - standard model for data interchange on the semantic web. *https://www.w3.org/RDF/*

[6] ARES is available at: *http://wwwinfo.mfcr.cz/ares/*

5 Conclusion and Future Work

EasyMiner available at *http://easyminer.eu* is an experimental academic data mining system for association rule learning, building of classifiers composed of association rules and for detection of outliers based on frequent pattern mining. The new version supports outlier detection and features a new REST API covering complete functionality of the system. A tutorial for using the REST API using examples in Python is available at *https://easyminer.eu/api-tutorial*.

There is an ongoing work on improving the scalability of the system and on supporting direct upload of data in the RDF format.

Acknowledgements

The research and development were supported by OpenBudgets.eu project (No. H2020-645833), and IGA grant 29/2016 of the University of Economics Prague. Tomáš Kliegr acknowledges long term institutional support of research activities by Faculty of Informatics and Statistics, University of Economics Prague.

References

1. Vojíř, S., Zeman, V., Kuchař, J., Kliegr, T.: EasyMiner/R Preview: Towards a Web Interface for Association Rule Learning and Classification in R. In: Challenge+DC@ RuleML. CEUR-WS (2015).
2. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: SIGMOD (1993) pp. 207-216.
3. Han, J., Pei, J., Yin, Y., Mao, R.: Mining frequent patterns without candidate generation: A frequent-pattern tree approach. In: Data Mining and Knowledge Discovery 8, pp. 53–87, (2004).
4. Kliegr, T., Kuchař, J., Sottara, D., Vojíř, S.: Learning Business Rules with Association Rule Classifiers. In: International Workshop on Rules and Rule Markup Languages for the Semantic Web, Springer (2014), pp. 236–250.
5. Klímek, J., Škoda, P., Nečaský, M.: LinkedPipes ETL: Evolved linked data preparation. In: International Semantic Web Conference, Springer (2016), pp. 95-100.
6. Zhang, W., Wu, J., Yu, J.: An Improved Method of Outlier Detection Based on Frequent Pattern. In: Information Engineering (ICIE), 2010 WASE International Conference on, Beidaihe, Hebei (2010), pp. 3-6.