Proceedings of the Federated Conference on Computer Science and Information Systems pp. 173–176
DOI: 10.15439/2017F351
ISSN 2300-5963 ACSIS, Vol. 11

Towards a Keyword Extraction in Medical and Healthcare Education

Martin Komenda, Matěj Karolyi, Roman Vyškovský, Kateřina Ježová, Jakub Ščavnický
Institute of Biostatistics and Analyses, Faculty of Medicine, Masaryk University, Kamenice 5, 625 00
Email: {komenda, karolyi, vyskovsky, jezova, scavnicky}@iba.muni.cz

Abstract—**Medical and healthcare study programmes cover various curricula consisting of many theoretically focused courses and clinical teaching training. Curriculum attributes usually contains thousands of requirements on the form of knowledge and skills which fully define a complete graduate profile. It is not humanly possible to go through the entire curriculum or to imagine how the individual courses, learning units, outcomes and branches of medicine are interrelated. This paper introduces an innovative analytical approach which helps to identify automatically the most frequent topics based on keyword extraction. Moreover, the transparent and clear web-based visualisation of achieved results is shown in practice.**

INTRODUCTION

CONTINUOUS enhancement of quality of medical education is a long-term and extremely challenging issue. Higher education institutions, especially in medical and healthcare domains, require a process of lifelong learning, starting from undergraduate professional education and continuing in graduate study and in clinical practice [1]. Generally, study programmes in medicine cover a set of learning outcomes combining theoretically focused courses and clinical teaching training. These skills and knowledge fully define a complete graduate profile. Unfortunately, such study programmes usually involve hundreds or even thousands elements of unstructured information which need to be understood, evaluated and optimised. As a result, it is very difficult to look at any given programme, to identify the main topics across the curriculum, and to find one's way through it to see what is actually being taught and how is it done [2,3]. From the perspective of human cognition abilities, it is not possible to carefully read and remember every single detail of all learning units and outcomes including their linkages and co-dependencies. The MEDCIN project[1] (Medical Curriculum Innovations) brings a new way of viewing and evaluating medical curricula. We have designed and implemented a web-based platform that makes information about a given curriculum accessible to curriculum designers, teachers and guarantors alike.

This paper addresses the need for an innovative methodology proposal which helps to identify automatically the most frequent topics taught over the six-year study of medicine and healthcare. The research problem was defined by the following questions: (i) How to apply data mining and analytical methods effectively for a crucial keyword-based exploration of medical curriculum data? (ii) Which visualisation components can be used for a transparent and clear web-based presentation of achieved results?

METHODS

A. Technological background

The MEDCIN platform is a web-based application with a client-server network architecture that consists of several different parts. Some of them are separated on the software layer, but we decided to use more than one physical environment due a more effective performance and manageability. Its three main parts involve an application core, a database server and an OpenCPU server [4]: (i) The application core is based on a Symfony framework[2] written in PHP, Javascript and HTML, which uses the Twig template engine[3]. Currently we work with a Symfony framework in version 3.2 and all developed modules are PHP 7.* compliant. The platform architecture has been implemented using the Model-View-Controller software architectural pattern [5]. Doctrine[4] was applied for PHP entities mapping in Model part (data access layer). All Twig templates are located in View part. Routing and data passing is covered by Controller part. (ii) The MEDCIN platform database is located on the database server and provides one public scheme where all descriptive curriculum data are stored. PostgreSQL[5] (version 9.5) represents a database system providing extensive data retrieval for this platform. (iii) The OpenCPU server represents the last physically separated part. The R statistical software environment (version 3.3.3) has been installed on this server, and all R scripts are stored and executed here. Every particular script is available on a specific URL thanks to the OpenCPU and its interface. The communication with this component goes on through REST API[6] methods.

A request-response POST method with concrete input data is used for RPC[7]. The OpenCPU either sends output data (in the form of JSON[8] or another format) directly to the web application or stores all outputs of script to a temporary folder structure which is accessible by a token. The R server is used

[1] http://www.medcin-project.eu
[2] http://symfony.com/what-is-symfony
[3] https://twig.sensiolabs.org/
[4] http://www.doctrine-project.org/about.html
[5] https://www.postgresql.org/about/
[6] https://www.opencpu.org/api.html
[7] Remote Procedure Call
[8] JavaScript Object Notation

especially during the text analysis of the curriculum, where the retrieval of results is the most complicated and the most time-consuming issue.

B. Keyword Extraction

Keyword extraction is an important technique for document retrieval [6]. By extracting appropriate keywords, we are able to show the most frequent and potentially relevant topics occurring in the entire curriculum. Before we describe data processing and analysis phases, data structure of the curriculum needs to be introduced. The MEDCIN platform provide the following building blocks which serve as common parameters for a standardised specification of the curriculum: A `sequence block` (SQB) contains sub-elements that define an organisational component of the curriculum, such as a course, a module, a learning unit or a learning block (e.g. Anatomy I – Lecture). An `event` contains information about educational and assessment events that make up the curriculum (e.g. Abdominal Radiology). `Competency objects` (CO) involve learning outcomes, competencies, learning objectives, professional roles, topics or classifications, which define what students should be able to know or demonstrate in terms of knowledge, skills, and values (e.g. Student analyses benefits and issues of to the up-to-date development of the healthcare system).

The MEDCIN platform contains a complete curriculum of the General Medicine master's degree programme at the Faculty of Medicine of the Masaryk University. The database contains more than 140 courses which are described by 1,347 events (learning units) and 6,974 competency objects (learning outcomes); in total, this makes up more than 2,500 pages of text. We have proposed a robust data model which portrays all fundamental attributes as well as relations between the individual items.

1) Data queries

Data for the MEDCIN platform are stored in the PostgreSQL database system running on a database server. The structure of the database is mapped to the Symfony application by a Doctrine 2 ORM[9] framework. First of all, the XML[10] metadata files are generated from the existing database; these files are subsequently transformed to PHP classes. The connection between the stored data and the web application is ensured by DQL[11] and SQL[12]. DQLs are typically used for easier queries such as to return the content of a single table using the findOneBy and findBy methods. SQLs are more suitable to execute more complex queries, when many tables are joined and several columns selected.

An important application of SQL in the context of this paper is data retrieval for keyword analyses. Two queries are implemented: (i) The first SQL query takes titles, keywords and text descriptions of sequence blocks, events and competency objects, puts it together and returns it as a single text. The keyword analyses are based on selected sequence blocks. All text descriptions of events and its COs related to selected sequence blocks and all text descriptions of COs related to selected sequence blocks enter the text analyses. A particular event can appear in the final text field multiple times depending on the number of connections to COs. Similarly, competency objects can appear many times because each of them can be linked to both SQB and event. However, text representation of sequence blocks is always distinct. The text field prepared in this way serves as an input for a word cloud and frequency analyses.

(ii) The second SQL query prepares data for a similarity analysis of events. It returns titles, keywords and descriptions of each event and its COs in a single text. Since SQB can be selected by the app user, the SQL query returns only texts relevant to this selection. Descriptions of the events can be included multiple times, depending on the number of linked COs.

2) Data analysis procedures

Data analysis in based on a proven CRISP-DM reference model [7], which provides the life cycle guideline of a data mining project. Special attention was paid to the pre-processing phase, where many unexpected issues appeared. This task includes data collection, description, exploration and verification. In general, we aim to obtain a clean final dataset with meaningful words in their basic forms. It also means reduction of data size and computational time. The input text covers the following metadata, which were subsequently mapped to the designed database structure: courses, learning units, learning outcomes and their descriptive attributes. First of all, non-text expressions – such as HTML characters, punctuation (! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~) and numbers (0-9) – were replaced by spaces. Since R is case-sensitive, the next step was to transform the words to lower case, in order to ensure that data are stored in a unified format. In the following step, the so-called stop-words were removed. Finally, data cleaning such as additional white space removing and replacing multiple spaces by a single one was performed. The second challenging part is to get stems from the mined keywords (a stem was considered a form of the word that never changes even when morphologically inflected). We used a procedure called stemming, specifically the Porter's algorithm [8], which is an iterative series of simple rules that chops off the suffix from a word and leaves a stem. A set of stems input to the data analysis itself.

3) Data visualisation

Text analysis of the curriculum requires an input in the JSON format, which is passed into POST method used with RPC. OpenCPU server automatically parses all JSON objects using the `jsonlite` R package afterwards. Therefore, we do not need any additional JSON processing. The keyword extraction R package generates two output sources: (i) A JSON data file (front-end visualisations are made on the basis of this file); we use the d3.js library as well as a native

[9] Object-relational mapping
[10] eXtensible Markup Language
[11] Doctrine Query Language
[12] Structured Query Language

HTML5 functionality to create some graphs and data tables. (ii) A SVG[13] file (created completely by the R script and passed to the web application screen).

R package visualisations depend heavily on several R graphics packages (see Fig. 1). In particular, these include `wordcloud` and `ggplot2`. Firstly, the `wordcloud` package creates good-looking word clouds and avoids over-plotting of texts in scatter plots. One main drawback is that it uses the base R graphics and therefore cannot be exported into fully responsive and reusable SVG objects. Secondly, the `ggplot2` package is a plotting system for R which provides a powerful model of graphics and makes it easy to produce complex multi-layered outputs. This package plots dendrograms and uses grid graphics, which can be successfully exported into responsive SVG objects. In order to have all visualisations fully responsive, we use a modified version of the wordcloud function that computes the necessary coordinates, but plots graphics using the `grid` and `ggplot2` packages afterwards. The entire visualisation process can be represented by the following dependency tree:

Fig. 1 Scheme of R package graphic dependency tree.

After the OpenCPU server performs the keyword extraction analysis, it creates necessary visuals and stores them in temporary folders. All visual results are afterwards accessible by individual tokens for their further use in the website environment.

RESULTS

The MEDCIN platform is divided into four linked modules providing medical and healthcare curriculum overview from different perspectives: (i) Summary report; (ii) Building blocks' context; (iii) Search by keyword; (iv) Text analysis. The fourth module is completely based on the keyword extraction algorithm, which was described in the Methods section. The user is allowed to select particular sequence blocks for a detailed analytical report (at least one, at most three SQBs). The visual representation of the most frequent keywords covering a wordcloud, a histogram and a data table is then displayed. These three graphical interpretations provide three different points of view on the same dataset.

The wordcloud-producing part of the R frequency analysis procedure is parameterised; namely, image margins, word orientations, font size and word size boundaries are customisable. Therefore, we are able to modify the output for various purposes (web application environment, PowerPoint presentations, printed materials etc.). All R procedures can be called directly from graphic user interface of the OpenCPU server (OpenCPU API Explorer), which is involved in the default server installation.

Moreover, the MEDCIN platform visualises the content similarity between all related SQB subset (events) using a dendrogram. It draws a tree diagram illustrating hierarchical clusters based on a term-document matrix of keywords frequency vectors representing the occurrence of keywords in particular events. Euclidean distance has been used to compute events dissimilarities (see Fig. 2). The distances between two particular events visualise how similar these events are (based on the keyword occurrence). The achieved results will provide the possibility of an effective evaluation of the curriculum by senior curriculum designers and guarantors of a given medical and healthcare discipline. The final online analytical reports must be assessed in terms of meaning, interpretation and visual transparency.

DISCUSSION

The main limitation concerning R-based visualisations is the non-availability fully responsive and reusable vector outputs. One of the possibilities to get the required SVG object is to use a graphic output of the ggplot2 package. We managed to plot the dendrograms using this package. In terms of wordclouds, ggplot2 does not have any feature for these types of graphs yet. But with the massive package expansion, ggwordcloud feature is expected to be developed. In order to keep the wordclouds both informative and good-looking, we decided to use the wordcloud package, which limits the SVG output responsiveness due to the usage of the base R graphics. Solving the issue of SVG export might be one of our goals for the future.

Moreover, stemming is not necessarily a user-friendly approach for further visualisations; rather, it is desirable to obtain a meaningful word again. Therefore, our future plans are to add a stem completion function, which will take all words with the same stem, find the equivalent of their original forms and complete the stems to these respective forms.

In future, the main challenge in terms of produced information visualisations will be to create outputs which would combine the standardised MedBiquitous vocabulary with an understandability to wider groups of users. The problem is that some of the defined terms are either too concrete and not well-known or, by contrast, too common and abstract (e.g. sequence block). That is why the visualisations or web page blocks are followed by text with additional information and examples.

CONCLUSION

We described the main research questions of automatic keyword exploration on medical and healthcare education data. We showed a pilot R package analysis including web-based visualisations in accordance with a proven data-mining methodology, which led to real results in practice. We demonstrated a powerful tool for the visualisation of curriculum data, which makes an overview of comprehensive keywords very simple to be critically evaluated by an expert in a given field. The selected algorithm for keyword exploration was successfully implemented as a module in the MEDCIN platform.

[13] Scalable Vector Graphics

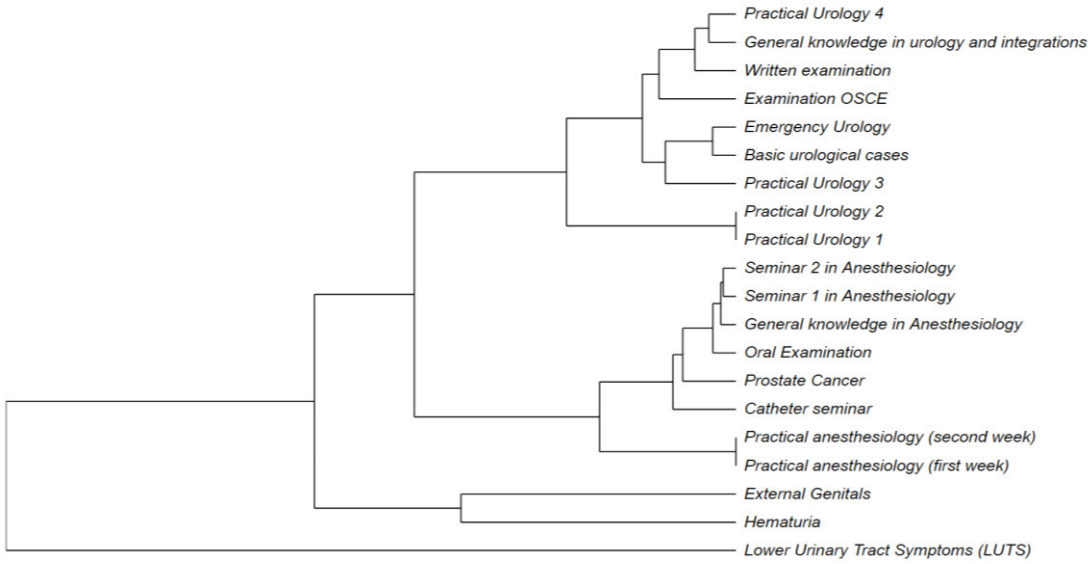

Fig. 2 Events similarity dendrogram for Clinical Medicine.

DISCUSSION

The main limitation concerning R-based visualisations is the non-availability fully responsive and reusable vector outputs. One of the possibilities to get the required SVG object is to use a graphic output of the `ggplot2` package. We managed to plot the dendrograms using this package. In terms of wordclouds, `ggplot2` does not have any feature for these types of graphs yet. But with the massive package expansion, `ggwordcloud` feature is expected to be developed. In order to keep the wordclouds both informative and good-looking, we decided to use the `wordcloud` package, which limits the SVG output responsiveness due to the usage of the base R graphics. Solving the issue of SVG export might be one of our goals for the future. Moreover, stemming is not necessarily a user-friendly approach for further visualisations; rather, it is desirable to obtain a meaningful word again. Therefore, our future plans are to add a stem completion function, which will take all words with the same stem, find the equivalent of their original forms and complete the stems to these respective forms.

CONCLUSION

We described the main research questions of automatic keyword exploration on medical and healthcare education data. We showed a pilot R package analysis including web-based visualisations in accordance with a proven data-mining methodology, which led to real results in practice. We demonstrated a powerful tool for the visualisation of curriculum data, which makes an overview of comprehensive keywords very simple to be critically evaluated by an expert in a given field. The selected algorithm for keyword exploration was successfully implemented as a module in the MEDCIN platform.

ACKNOWLEDGMENT

The authors were supported from the following grant projects: (i) MEDCIN – Medical Curriculum Innovations – Project No.: 2015-1-CZ01-KA203-013935, which is funded by the European Commission ERASMUS+ programme; (ii) OPTIMED portal – Project No.: MUNI/FR/1568/2016 and MERGER – Project No.: MUNI/A/1339/2016, which are funded by the Masaryk University. We are also thankful to partners of the MEDCIN project, namely Christos Vaitsis (Karolinska Institutet), Luke Woodham (St George's University of London) and Dimitris Spachos (Aristotle University of Thessaloniki).

REFERENCES

[1] R. H. Ellaway, S. Albright, V. Smothers, T. Cameron, and T. Willett, "Curriculum inventory: Modeling, sharing and comparing medical education programs," *Med. Teach.*, vol. 36, no. 3, pp. 208–215, nor 2014.

[2] M. Komenda, "Towards a Framework for Medical Curriculum Mapping," Doctoral thesis, Masaryk University, Faculty of Informatics, 2015.

[3] M. Komenda, M. Karolyi, A. Pokorná, M. Víta, and V. Kríž, "Automatic keyword extraction from medical and healthcare curriculum," in *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on*, 2016, pp. 287–290.

[4] J. Ooms, "The OpenCPU System: Towards a Universal Interface for Scientific Computing through Separation of Concerns," *ArXiv14064806 Cs Stat*, Jun. 2014.

[5] "Model View Controller(MVC) in PHP." [Online]. Available: http://php-html.net/tutorials/model-view-controller-in-php/. [Accessed: 24-Mar-2011].

[6] Y. Matsuo and M. Ishizuka, "Keyword extraction from a single document using word co-occurrence statistical information," *Int. J. Artif. Intell. Tools*, vol. 13, no. 01, pp. 157–169, 2004.

[7] A. I. R. L. Azevedo, "KDD, SEMMA and CRISP-DM: a parallel overview," 2008.

[8] A. Deyasi, S. Mukherjee, P. Debnath, and A. K. Bhattacharjee, *Computational Science and Engineering: Proceedings of the International Conference on Computational Science and Engineering (Beliaghata, Kolkata, India, 4-6 October 2016)*. CRC Press, 2016.