campos_2020_recommendation_system_for_knowledge_acquisition_in_moocs_ecosystems

Year

2020

Author(s)

Rodrigo Campos and Rodrigo Santos and Jonice Oliveira

Title

Recommendation System for Knowledge Acquisition in MOOCs Ecosystems

Venue

SBSI

Topic labeling

Fully automated

Focus

Secondary

Type of contribution

Novel approach

Underlying technique

Topic labeling parameters

Label generation

(12)

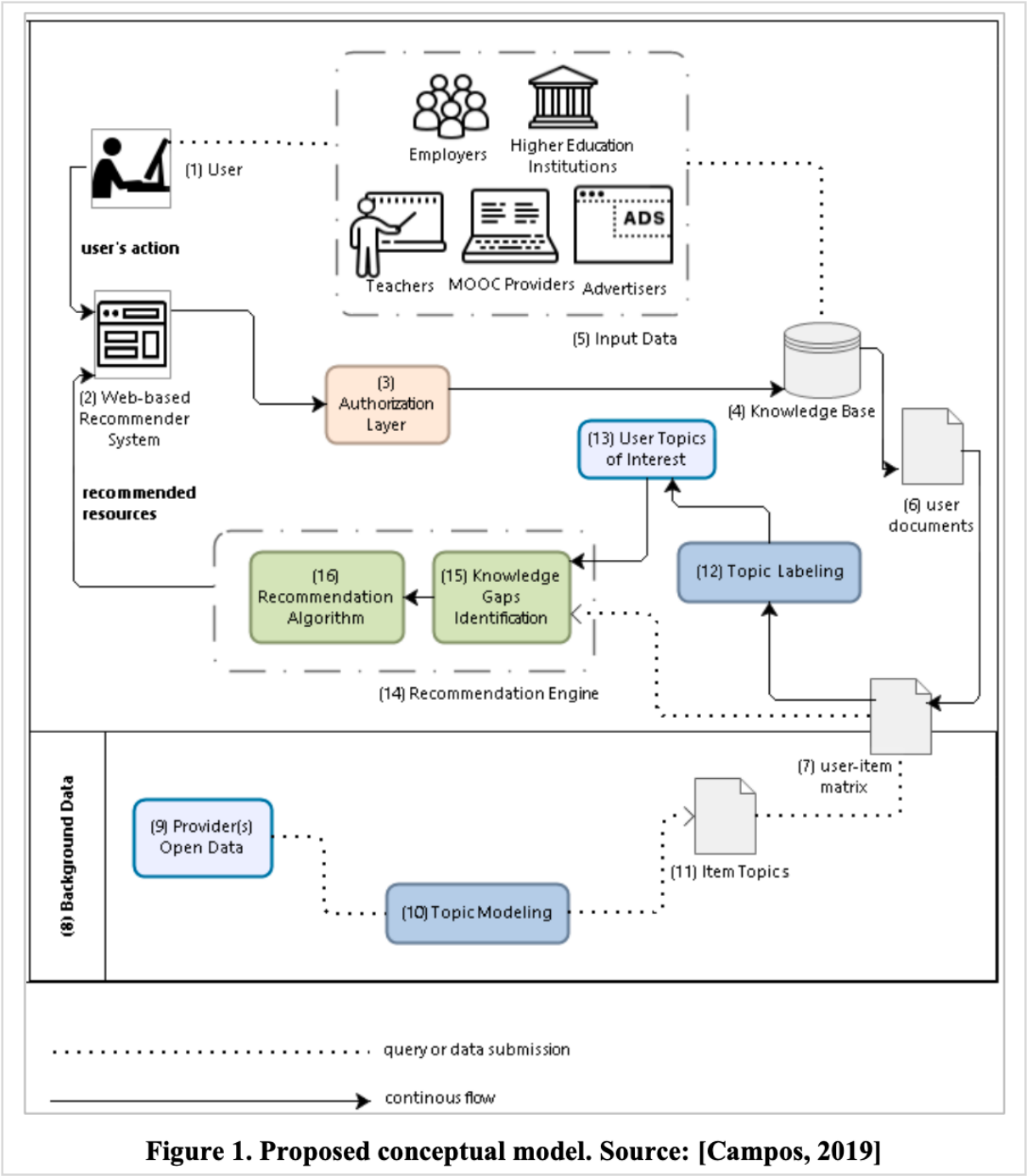

Figure 1. Proposed conceptual model. Source: [Campos, 2019]

With the providers' data selected, the topic modeling method (10) creates the item topics (11).

Using these topics, it is possible to complete the "item" of the user-item matrix (7);

This matrix(7) is input to the techniques responsible for labeling topics (12).

Algorithm 2: MOOCs Ecosystems Automatic topic labeling

Input: Quantity of generated topics k, document-topic matrix W, description (*snippets*) of each document, the generated *model* of topics, the vectorized terms *vec*, and the *approach* selected

Output: *top-1 label and top-3 label*

```
topTerms = getTop10T(model, 10, k, vec)
for topic_i = 1 to k do
    top_D = getTop10D(snippets, W, topic_i, 30)
    top_T = topTerms[topic_i]
    for d = 1 to top_D do
        dt_label = dt_label + getPrimitiveLabels(d, approach)
    if approach is TS do
        for primitive = 1 to dt_label do
            if primitive in top_T do
                list = list + primitive
    else if approach is KS do
        list = dt_label
    candidates = applyTFtoRank(list)
    top-1 = getTopLabel(candidates, 1)
    top-3 = getTopLabel(candidates, 3)
```

TS approach performs text extraction following the fast keyword extract algorithm.
KS uses keyword search

As a first step, it is necessary to select the top-30 documents associated with each topic. Each document is a module of different courses from multiple providers, so a course can have different modules with the same name.

Next, it is necessary to select the top-10 terms for each topic.
For the selection of primitive labels in TS, the fast extraction algorithm provides a wide text of all 30 documents.

After selecting primitive labels, it is necessary to choose the candidate labels.
In the case of TS, we consider all primitive labels that are in the top-10 term set of the topic.
Next, Term Frequency (TF) ranks each term according to the word frequency of the candidates in the primitive label string.

In the case of KS, we consider the same top-30 documents and the top-10 terms associated with each topic. The novelty of the KS approach starts in the selection of

primitive labels. For each of the 30 documents, we select the area in which the course is inserted. While in TS there is a primitive check with the top-10 terms for selecting candidate labels, all primitives are candidates in KS. Then, we apply the TF to order the candidates and, finally, it is possible to select the terms.

Finally, in the selection of top-3, we verified the case of one label being a substring of another (for substitution).
Algorithm 2 has two outputs: one consisting of top-1 labels for each topic and one consisting of top-3 labels for each topic.

Motivation

In our approach, the labels help to define item topics (11) and also to create the user's topics of interest (13);

Topic modeling

Non-negative Matrix Factorization (NMF)

Algorithm 1: Automatic NMF topic modeling integrated with providers. Based on [Greene et al., 2014]

Input: JSON data dt from providers where each row represents a module and list of stopwords sw
Output: W (document-topic matrix) and H (topic-term matrix)

```
list = TransformDataIntoUTF8List(dt)
tokenizer = LemmaTokenizer( )
Vectorizer = TfidfVectorizer(sw, tokenizer)
A = CreateDocumentTermMatrix
vocabulary = Vectorizer(A)
kmin, kmax = SetValues( ) #integer is required
for k = kmin to kmax do
    CalculateCoherence(k)
    coherences = [k]
best_k = GetBestK(coherences)
W = GenerateDocumentTopicMatrix(A, best_k)
H = GenerateTopicTermMatrix(A, best_k)
```

Topic modeling parameters

\

Nr. of topics

The procedures executed to find the value of the ideal number of topics represented by the variable k (lines 6 to 10 of Algorithm 1) are based on the stability analysis approach for automatic calculation of k, proposed by Greene et al. (2014, apud Nolasco, 2016).
As such, tests to find k are based on applying the modeling method to different values of k (given a minimum and maximum k, called respectively kmin and kmax at lines7 of Algorithm 1) until a k that reproduces a topic coherence value higher than the others.

Label

Label selection

In the top-3 labels case, cosine similarity to labels from MOOCs providers (see `label quality evaluation`)

Label quality evaluation

The second experiment focused on evaluating the topic labeling method, mainly the representativeness of labeling technique. Labels for topics are generated from our approach and then compared to labels from MOOCs providers.

Cosine similarity
The evaluation methodology is designed to select the dataset and use our approach to label generation (Text Selection or TS, its variations, and Keywords Selection or KS). Then, we select these labels and compare them with provider labels.
This comparison is made by using the cosine distance of strings (labels).
For this experiment, the automatic labeling technique that obtains the closest proximity to the provider's manual labels is the most appropriate and contains better results.

From the collected results, one way to compare the approaches is to analyze the distance of the automatic labels with those already existing ones in the providers.
Table 3 shows the cosine distance between the TS (top-1), TS (top-3), and KS strings relative to the provider strings.

Table 3. Distance between strings in each approach. Source: [Campos, 2019]

Topic	TS (top-1) Cosine	TS (top-3) Cosine	KS Cosine	Best Approach
0	0.0000	0.5221	0.7957	KS
1	0.0000	0.7512	0.0578	TS (top-3)
2	0.0917	0.8214	0.2380	TS (top-3)
3	0.0000	0.8436	0.0430	TS (top-3)
4	0.0000	0.6642	0.0626	TS (top-3)
5	0.0000	0.6679	0.6180	TS (top-3)
6	0.0000	0.7605	0.3220	TS (top-3)
7	0.2917	0.8674	0.3713	TS (top-3)
8	0.1856	0.8572	0.1856	TS (top-3)
9	0.0000	0.6487	0.6356	TS (top-3)
10	0.0000	0.8211	0.2567	TS (top-3)
11	0.5071	0.8052	0.5738	TS (top-3)
12	0.0000	0.6221	0.6944	KS
13	0.0000	0.7754	0.6203	TS (top-3)

Assessors

\

Domain

Paper: Recommender systems (for MOOCs)
Dataset: MOOCs

Problem statement

The main contribution of this work is the identification and reduction of the students' knowledge gap in MOOCs.
As such, we model and analyze the MOOCs ecosystems and propose a solution for recommending parts of courses.

Corpus

Origin: Various MOOCs providers (Khan Academy, Udemy, and edX)
Nr. of documents: 106,574
Details:

Document

Mooc module content

Pre-processing

Removal of special characters
Removal of stop words
Lemmatisation

```
@inproceedings{campos_2020_recommendation_system_for_knowledge_acquisition_in_moocs_ecosystems,
 author = {Rodrigo Campos and Rodrigo Santos and Jonice Oliveira},
 title = {Recommendation System for Knowledge Acquisition in MOOCs Ecosystems},
 booktitle = {Anais Estendidos do XVI Simpósio Brasileiro de Sistemas de Informação},
 location = {Evento Online},
 year = {2020},
 keywords = {},
 issn = {0000-0000},
 pages = {93--108},
 publisher = {SBC},
 address = {Porto Alegre, RS, Brasil},
 doi = {10.5753/sbsi.2020.13132},
 url = {https://sol.sbc.org.br/index.php/sbsi_estendido/article/view/13132}
}
```

#Thesis/Papers/BS