More Data and Better Keywords Imply Better Educational Transcript Classification?

Theodora Ioana Danciulescu,
Marian Cristian Mihaescu
University of Craiova
Department of Computer Science
theodora_danciulescu@yahoo.com
mihaescu@software.ucv.ro

Stella Heras, Javier Palanca,
Vicente Julian
Universitat Politecnica de Valencia
Sistemas Informaticos y Computacion
stehebar@upv.es, jpalanca@dsic.upv.es
vjulian@upv.es

ABSTRACT

Building and especially improving a classification kernel represents a challenging task. The works presented in this paper continue an already developed semi-supervised classification approach that aimed at labelling transcripts from educational videos. We questioned whether the size of the ground-truth data-set (Wikipedia articles) or the quality of the keywords used in the semi-supervised labelling have a significant impact on the accuracy metrics of the final obtained data model. Experimental results took into consideration three Wikipedia data-sets of *Small*, *Medium* and *Large* sizes. For each data-set there were used three sets of keywords: offered by video authors, determined by *rake-nltk* on available transcripts and determined by *rake-nltk* on Wikipedia articles that serve as training and testing data for the LDA model that determine keywords on the transcripts. Experiments show that the size of the data-set has little importance, while the quality of the keywords has a more significant impact. Therefore, an improved version of the previously developed classifier has been obtained by improving the quality of the keywords involved in semi-supervised training. This result paves the way towards further improvements that may finally be deployed as within a recommender system of educational videos at the Universitat Politècnica de València.

Keywords

classification, educational transcripts, keywords, data-set size

1. INTRODUCTION

Over the last few years, the quantity of online learning objects (LO) [6] and Massive Online Learning Courses (MOOCs) have increased dramatically representing a real boom in online learning. This boom of online learning resources has caused a problem for students, as they have hundreds of thousands of online documentation. At the same time, different approaches to discover topics and hidden semantic structures in text have been proposed with the goal of go forward on topic modelling which has been a challenging and critical issue for information retrieval. Therefore, taking into account all of this, topic modelling has become in a trending topic for the e-learning research community. Following that trend, the *Universitat Politècnica de València* (UPV) in Spain launched a video lectures sharing website, called *Polimedia*[1], and a MOOC platform, called *UPV[X]*[2], which is powered by the edX MOOC platform[3].

Theodora Danciulescu, Stella Heras, Javier Palanca, Vicente Julian and Cristian Mihaescu "More Data and Better Keywords Imply Better Educational Transcript Classification?" In: *Proceedings of The 13th International Conference on Educational Data Mining (EDM 2020)*, Anna N. Rafferty, Jacob Whitehill, Violetta Cavalli-Sforza, and Cristobal Romero (eds.) 2020, pp. 381 - 387

Both proposals have a basic search engine allowing students to search for videos (learning objects) by simply using a set of keywords. Current solutions compare these keywords with some typical metadata of the videos (title, authors, ...) and returns the set of videos that match with this data. Obviously, this basic retrieval solution overlooks any semantics, which produces incomplete results that do not take into account some videos that are relevant for the student but that do not include any of the keywords in their titles.

The MOOCs we are using in this work consists of a set of educational videos that have an automatic transcription of the lectures that is going to be used as part of the input data for this proposal. The motivation of this work is to use this information to help students to find more suited learning objects, personalized to their interests, in these massive online platforms where the number of learning objects grows quickly and they usually are not tagged correctly.

According to this, this paper focuses on the improvement of this search engine proposing a new retrieval method that uses a dataset extracted from Wikipedia articles and that is trained to classify keywords based on the topic of the available educational videos. This proposed model is an improvement of a previous work presented in [14], where pre-tagged wikipedia articles were used as ground-truth. In this work we improve this semi-supervised method by: 1) automatically tagging Wikipedia articles and using them to create an extended dataset for training the semi-supervised method, and 2) proposing an improved pipeline for cleaning the data, extract keywords and obtain a better classification model that improves the precision of the student's searches.

The rest of the paper is structured as follows: Section 2 presents some works related to the topic of this paper; Sec-

[1] UPV Media, https://media.upv.es

[2] UPV[X], https://www.upvx.es/

[3] edX MOOC platform, https://www.edx.org/

tion 3 details the approach proposed by the authors; Section 4 presents some experimental results; and finally, Section 5 shows the conclusions of this work.

2. RELATED WORK

The problem of the correct keyword extraction is a recurrent problem over the last few years. Different works have appeared trying to solve this problem using different approaches. At the end, the idea is to have a solid set of words that concisely represent the content of a text (in this case the content of a learning object).

Most of the last approaches on document-oriented methods of keyword extraction use natural language processing (NLP) techniques mainly based on machine learning algorithms and statistical methods. One of the most well-known approaches is the work presented in [17] where authors propose the use of Support Vector Machines as a way to extract the most important keywords.

On the other hand, the work in [9] presents a solution based on the graph-based syntactic representation of text and web documents that combines supervised and unsupervised learning. In a similar way, the work presented in [7] proposes an unsupervised keyword extraction technique including several different ways of the conventional TF-IDF model with reasonable heuristics. Other approaches, like the work presented in [12] called Rapid Automatic Keyword Extraction (RAKE), employ unsupervised methods for extracting keywords which are domain-independent, and also, language-independent.

The latent Dirichlet allocation (LDA) model is one of the most used techniques to classify documents according to a set of topics. One example is the work presented in [1] that automatically captures the thematic patterns and identifies emerging topics using a non-Markov on-line LDA Gibbs sampler topic model. In the online educational field, the LDA model has been used in works such us the presented in [16] where the authors use topic detection for the analysis of the feedback submitted by students in online courses. The work in [10] tries to solve the problem of topic detection by identifying words that appear with high frequency in the topic and low frequency in other topics.

Some works face the keyword extraction problem in learning objects through the use of other approaches such as ontologies like the work presented in [8] that aims to improve the effectiveness of retrieval and accessibility of learning objects integrating semantic knowledge through domain-specific ontologies. In [4] authors use Wikipedia to associate learning objects to Wikipedia pages, specifically with the topics of those pages, trying to find relationships among learning objects.

Finally, recent work also uses intelligent algorithms and method to face other challenges of efficient videolecures management, such as video shots skimming [15] and supervised multi-class classification [5].

Opposite to most related works, our method is fully semi-supervised, with no need for a previously tagged database nor an ontology, that can act as ground truth to train the models. Also, to the best of our knowledge, there are no other intelligent systems trained to automatically classify a Spanish database of educational videos.

3. PROPOSED APPROACH

From a classification perspective, the first issue is to clearly state the actual number of topics (i.e., labels) that exist in available transcripts. Since all transcripts come from educational videos from UPV, it certainly means that the number of topics is represented by the domains from which videos come from, that is *biology & sciences* (BS), *engineering*(E) and *humanities & arts* (HA). BS topic considers aspects of bacteria, diseases, bio-engineering, bio-medicine, E topic considers aspects of computers, electrical, architecture, civil, aerospace. In contrast, HA considers aspects of laws, arts, social and economic.

The proposed approach extends the semi-supervised method described in a previous paper-work[14]. It improves the data analysis pipeline in terms of accuracy of classification on the videos currently available in the database. As in the initial approach, the training on Wikipedia articles uses the SVM[3] classification algorithm, which used a Radial Basis Function (RBF) kernel from the *sklearn* library[11]. The validation approach uses the same two steps: 1) train on 70% of Wikipedia articles and cross-validate with 15%, 2) train on labelled transcripts and validate on remaining unseen 15% of Wikipedia articles.

Internally, the semi-supervised training has been performed on a set of labelled Wikipedia articles by building a data model that has been used for classifying educational transcripts and their associated keywords. The transcripts which had the same label as the keywords were considered correctly labelled and therefore were added to the initial training dataset. The newly obtained dataset is used in an iterative semi-supervised set up for training in an attempt to tag as many educational transcripts as possible.

One limitation of previous works is that HA items were mislabeled as E. This flaw may be caused by the fact that videos about HA reach more various subjects, that are not so domain-specific. Mathematics videos with proofs demonstration and analysis are also not correctly labelled as there is a large number of words that are not mathematics domain-specific. Many videos about the economy and economic environments tend to be categorised as E, as many explanations heavily use mathematics and calculus. A positive aspect is that the classification for BS items is acquiring excellent results, there are no confusions made for this domain. This behaviour is expected as this domain has many specific terms and principles, so videos from this area are easily classifiable and do not create confusions.

As a first step to improve the previous work[14] was to extend the Wikipedia articles data-set for training the semi-supervised method. This was done progressively, as we compared results with the previous ones and checked manually if the videos that were badly classified have been classified correctly. The decision about the amount and about which Wikipedia articles categories should be downloaded was made by manually analysing the clustering results from previous work. By doing so, we obtained best results with three versions of datasets: a *Small* data-set (3747 Wikipedia articles), a *Medium* dataset(6373 Wikipedia articles) and a *Large* dataset (18527 Wikipedia articles).

Secondly, we focused on the importance of relevant keywords to obtain a good classification result. There were provided three sets of keywords supplied by three different methods. The first set was obtained using the same process from the previous paper[14] by using the keywords provided by the videos' authors. However, we observed inconsistencies as some videos do not offer keywords in their metadata. The second set of keywords was obtained by using *rake-nltk*[13] tool for extracting the keywords directly from the transcripts' text. Finally, the third set was obtained by using *rake-nltk* tool for getting keywords from the *Large* data-set of Wikipedia articles to use them as training and testing data for an LDA (Latent Dirichlet Allocation)[2] model that will extract domain-specific tags from the transcripts.

3.1 Training on more Wikipedia articles

Intuitively, more data should help to improve the accuracy, but in practical situation this may not happen. An issue that currently occurs in machine learning systems is whether or not the size of the data-set is too small for the classification problem. Proper debugging of the data analysis pipeline should clearly point out if current accuracy results may be improved by using a larger data-set or other leverages should be taken into consideration.

As a first approach, we tried to detect a pattern in the classification errors and download the appropriate Wikipedia articles to cover the subjects in the videos that were mistakenly classified. Consequently, when choosing the Wikipedia articles, not only the covering of the topics was taken into consideration but also the quantity of the articles about that subject was an important factor.

In response to this, additionally to the initial *Small* dataset used in the previous work[14] we obtained two new datasets: ***Medium*** dataset with a total of 6373 articles (i.e., 1219 BS articles, 2737 HA articles, and 1626 E articles), and a *Large* dataset with 18526 Wikipedia articles (including 5830 BS articles, 5882 E articles and 6814 HA articles).

3.2 Determining better keywords

The transcripts' keywords represent a key-point for the classification algorithm, as the quality of the classification may be directly influenced by the relevance and quality of the keywords.

A second solution was represented by the *rake-nltk* tool, as it supports the Spanish language and it provides good results for this language, too. *Rake-nltk* tool is a domain-independent keyword extraction algorithm which tries to determine key phrases in a body of text by analyzing the frequency of word appearance and its co-occurrence with other words in the text.

After trying to classify the videos in 3 clusters (BS, E and HA) using three different sized data-sets (i.e., *Small*, *Medium* and *Large*) for training and two different methods for assigning keywords to each transcript (the manually provided keywords by authors and the keywords extracted with *rake-nltk*), we finally use the third method of providing more domain-specific keywords for every transcript: we used LDA as business logic for the implementation of transcript keywords recommendation system and used *rake-nltk* for providing keywords for Wikipedia articles to obtain training and testing data.

As the transcripts and the keywords from the metadata (i.e. authors' keywords) do not represent a valid data-set (the words used as keywords are either ambiguous, either too name specific and they often induce classification errors).

The limitation of the second method consists from the fact that the keywords provided by *rake-nltk* from transcripts were large and with numerous phrases without a focus on the essential subject of the video, also causing classification errors in some cases. So, a third solution was needed: there were used Wikipedia articles and keywords extracted with *rake-nltk* as training and testing data set for the LDA model to extract domain-specific keywords from the transcripts. The third solution is combining the *rake-nltk* tool with the LDA model. *Rake-nltk* will be used to extract keywords from the Wikipedia articles resulting in a labelled dataset that will serve later as training and testing dataset for the LDA model to extract domain-specific keywords from the transcripts.

The second approach provides new keywords for every transcript by using *rake-nltk*. The keywords extracted with this tool were also pre-processed by eliminating stop words and lowering all the letters. However, there still is one disadvantage for this method: the keywords extracted are large phrases that are not necessarily very domain-specific. Moreover, the extracted sentences are ambiguous in some cases, lacking the essential subject of the transcript. This error is most likely to be caused by the fact that the transcripts are not always subject-focused, they usually have an introduction about the teacher, the subject in general, many examples are provided. Hence, there is a broad set of words that may induce errors.

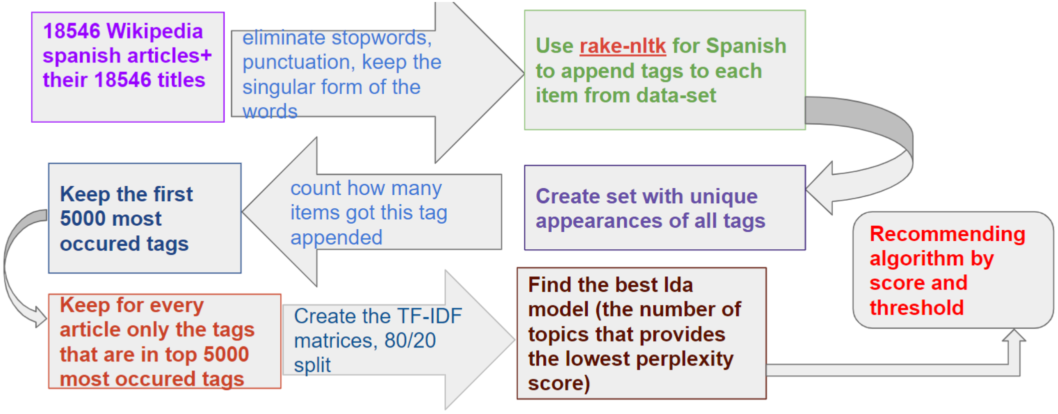
The third approach used *rake-nltk* tool, not for extracting keywords directly for our transcripts, but for extracting keywords for each article from the Wikipedia articles *Large* data-set (18526 articles). The tagged Wikipedia articles using *rake-nltk* will be used as training data for assigning keywords to the video transcripts employing LDA.

Figure 1 presents in detail the data analysis pipeline for the third method of providing keywords. This method is being described in this section in particular.

The following steps were followed for obtaining the domain-specific transcript tags recommendation algorithm utilizing LDA:

Create a balanced and large data-set of Wikipedia articles in Spanish. By saying to have a balanced data-set, there are supposed to be enough BS articles to obtain a set of keywords for BS, enough E articles to get a set of keywords for this domain, and most important enough HA articles to form a set of tags for this domain, too. The difficult part was to get a good set of keywords for HA domain, as this cluster covers a wide range of fields like Economy, Law,

Figure 1: Pipeline for extracting keywords from Wikipedia articles


18546 Wikipedia spanish articles+ their 18546 titles
eliminate stopwords, punctuation, keep the singular form of the words
Use rake-nltk for Spanish to append tags to each item from data-set
Create set with unique appearances of all tags
count how many items got this tag appended
Keep the first 5000 most occured tags
Keep for every article only the tags that are in top 5000 most occured tags
Create the TF-IDF matrices, 80/20 split
Find the best lda model (the number of topics that provides the lowest perplexity score)
Recommending algorithm by score and threshold

Arts, Architecture, Language learning, Politics, Social Sciences, Philosophy, Psychology and basically anything that does not fit in the other two clusters.
Clean the text from the downloaded Wikipedia articles by lowering text, removing undesirable marks and stop words, using the singular form of the word. Append each Wikipedia article tags using *rake-nltk* tool and also *clean* (lower text, remove undesirable marks, remove stop words, use the singular form of the word) these tags. For better results, there are also tags extracted from the titles of the Wikipedia articles. That means that we pull tags for 18526 x 2 items.
Add all these tags in a set to have only unique appearances of the extracted tags.
Count how many Wikipedia articles were assigned to each tag from the set.
Get top 5000 most occurred tags (having less tags, it means that only the most occurred tags from each domain will be kept, and in this way, a classification with the semi-supervised method will be simpler to perform with a smaller training data-set)
Keep only the top 5000 occurring tags for each Wikipedia article.
Keep only the articles that are still labelled. After these operations, we end up with 21743 labelled items out of 37092 items.
Create the TF-IDF matrices by splitting our obtained data set in 80%/20%.
We try to **train various LDA models** using *sklearn*[4] implementation [18], by assigning each of them a different topic number, then the different models are evaluated on the test set using the metric perplexity. By definition, the lower the perplexity, the better the model.
Showing the perplexity score for several LDA models with different values for n_components parameter, and printing the top words for the best LDA model (the one with the lowest perplexity).
Now that we have designed the workflow, we focus on the keywords recommendation algorithm for the transcripts, which is based on two main aspects:

- **Score** = probability that document is assigned to a specific topic, represents the topic's probability of generating the word.

- A word is considered as a relevant tag, when its score is superior to a defined threshold. After testing different values for the threshold, we decided to choose the threshold to 0.008, that is because, for this value, because with a threshold equals to 0.008 more than 95 percents of the transcripts have recommended tags.

Also, an advantage for obtaining keywords for every transcript employing *rake-nltk* combined with LDA would be that all the videos will be classified. In the original method, only the videos that were provided keywords by authors could have been taken into consideration. Now, as we offer keywords to every transcript, all the videos with an available transcript may be taken into consideration. An even bigger advantage is the fact that the training set contains articles about well-defined domains, their subject is focused on a small range of ideas, so the set of most frequently used tags will be very domain-specific, a fact that will be helpful for the classification algorithm.

4. EXPERIMENTAL RESULTS

After running the semi-supervised learning method for the *Small*, *Medium* and *Large* data-sets, and also with the three sets of keywords, the best results were obtained by training the semi-supervised method with the *Small* data-set of Wikipedia articles and the keywords provided employing *rake-nltk* for obtaining training and testing data and LDA to obtain the proper transcript's tags. The results are presented in Table 1. This table also provides a detailed insight of the semi-supervised training process results along with the number of transcripts added to the model in every iteration and with the classification accuracy obtained for each label. The computation of the classification accuracy metrics is done on the validation data-set, which contains only unseen data in the training step.

Analysis of the iterative semi-supervised training process in all nine scenarios (i.e., for three data-set sizes and for three methods of obtaining the keywords) revealed several pat-

[4] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.LatentDirichletAllocation.html

Iteration (valid /available)	Accuracy	Class	Precision	Recall	F1-score
#1 (8487 / 14395)	0.92 (+/- 0.01)	Biology&Sciences	0.95	0.93	0.94
		Engineering	0.88	0.92	0.90
		Humanities&Arts	0.95	0.93	0.94
#2 (2375 / 5908)	0.96 (+/- 0.02)	Biology&Sciences	0.92	0.94	0.93
		Engineering	0.87	0.88	0.87
		Humanities&Arts	0.95	0.93	0.94
...	...	...	...	...	...
#8 (9 / 1940)	0.94 (+/- 0.05)	Biology&Sciences	0.90	0.94	0.92
		Engineering	0.85	0.86	0.85
		Humanities&Arts	0.93	0.90	0.92

Table 1: Validation scores for each iteration in the pipeline with the *Small* Wikipedia articles data-set and the keywords provided by means of *rake-nltk* for obtaining training and testing data and LDA to obtain the proper transcript's tags.

terns. The first observation regards the fact that the number of iterations has low variance. So, irrespective of the size of Wikipedia data-set and the method for obtaining the keywords the number of iterations is in the range from six to twelve. This observation represents a clear indication that the size of the training data-set of Wikipedia articles does not highly influence the semi-supervised learning. Another observation is that each step in the semi-supervised training keeps unchanged or slightly decreases the F1 score, while slightly increasing the accuracy of the 10-fold cross-validation on the Wikipedia test data-set. This observation shows that all experiments are consistent and produce similar behavioural patterns in terms of accuracy, precision, recall and F1-score measures evolution in terms of evolution during semi-supervised training.

Table 2 presents the validation scores for all the three data-sets (i.e., *Small*, *Medium* and *Large*) and all three keywords data-sets.

The first observation regarding the validation results from table 2 regards the fact that there are no big differences in terms of overall accuracy and F1-scores for the three data-sets of keywords and for each training data-set. Still, the method with *rake-nltk* for Wikipedia articles keywords and LDA for obtaining transcript keywords generally has better scores than the other two methods for cluster 2. Still, it has usually lower scores for cluster 1. This pattern shows an indication that improvements in classification metrics should focus on classes where poorly results occur.

We further observe that scores tend to slightly decrease as the data-set is getting larger. Therefore for the *Medium* data-set, only the method with *rake-nltk* for extracting transcript keywords provides better results than it does with the *Small* data-set. A particular result consists in major score decreases for cluster 1 for the *Medium* data-set. This is mainly due to the unbalance of this data-set regarding the items from labelled in class 1. The imbalance of class 1 is also signalled by the excellent results for classes 0 and 1 in the experiment with *Large* data-set and the method with *rake-nltk* for Wikipedia articles keywords and LDA for transcript keywords.

Despite the *Large* data-set used for training the model, comparing the time required to train the model with the *Small* data-set and the time necessary to train the model with the *Large* data-set with all three sets of keywords, we have noticed that the time has doubled in the worst case, even though the data-set used is 6 times larger than the initial one.

Besides, the method to obtain keywords employing *rake-nltk* and LDA transcript keywords provide a better running-time execution for the *Small* and *Large* data-sets than the original keywords set as the number of iterations is also smaller.

The method with *rake-nltk* and LDA transcript keywords provides best result for the *Small* data-set, though the *rake-nltk* transcript keywords methods has the best results for the *Medium* and *Large* data-sets. For the method to obtain domain-specific keywords for transcripts employing *rake-nltk* to extract Wikipedia articles keywords and LDA to extract the proper keywords for transcripts, the tags distribution per the 10 topics of the model is presented in Table 3. We also notice that the 10 topics do not mix the three domains that we are interested about: E tags are found only in topics that do not contain tags from the other two domains, and the same for BS tags and HA tags. There can be easily noticed the domain that each topic covers: the topics with indexes 1, 2, 5, 9 and 10 are focused on HA domain, the topics with indexes 4, 6 and 8 are focused on E domain, and finally, the topics 3 and 7 are focused on BS domain.

Furthermore, the topic order shows that the first three most important topics are 4, 3 and 9, where 4 is focused on the E domain, 3 is concentrated in BS tags, and 9 is focused on HA tags. Considering that the first three most important topics contain one topic for each of the three domains that we are interested in, ultimately confirms that the model is suitable for our purpose. In addition, the following 3 topics in the topic order are also distributed equally across the three domains.

We can notice that the original keywords provided by authors are provided in different styles: some of them are too specific(tool names that are not so common), some of them too ambiguous to be categorised to a domain, and some of them provide domain-specific terms, but those terms may not be so standard in that domain in such a way to be correctly categorised by put semi-supervised method that is not trained on a massive data-set.

Table 2: Validation scores for all data-sets and keywords sets

Data-set	Keywords	Accuracy/Avg F1	Class	Precision	Recall	F1-score
Small	Original keywords	0.94/0.88	0	0.96	0.86	0.91
			1	0.84	0.86	0.85
			2	0.86	0.90	0.88
	rake-nltk transcript keywords	0.94/0.88	0	0.96	0.86	0.91
			1	0.84	0.87	0.86
			2	0.86	0.90	0.88
	rake-nltk and LDA transcript keywords	0.95/0.89	0	0.91	0.92	0.91
			1	0.81	0.91	0.86
			2	0.93	0.84	0.88
Medium	Original keywords	0.94/0.86	0	0.93	0.83	0.88
			1	0.75	0.92	0.83
			2	0.93	0.82	0.87
	rake-nltk transcript keywords	0.96/0.88	0	0.93	0.85	0.89
			1	0.80	0.90	0.85
			2	0.93	0.87	0.90
	rake-nltk and LDA transcript keywords	0.94/0.85	0	0.93	0.84	0.88
			1	0.72	0.91	0.80
			2	0.93	0.79	0.85
Large	Original keywords	0.95/0.86	0	0.95	0.82	0.88
			1	0.80	0.90	0.85
			2	0.86	0.85	0.85
	rake-nltk transcript keywords	0.96/0.86	0	0.95	0.81	0.88
			1	0.82	0.88	0.85
			2	0.83	0.87	0.85
	rake-nltk and LDA transcript keywords	0.95/0.85	0	0.93	0.82	0.87
			1	0.77	0.90	0.83
			2	0.86	0.82	0.84

Table 3: Highest score tags per topics in the LDA model

T 1	derecho / social / sociedad / política / cultura
T 2	dato / software / aplicación / versión / código
T 3	célula / proteína / agua / animal / forma / celular
T 4	algoritmo / error / programa / memoria / ejecución
T 5	mercado / precio / economía / financiero / empresa
T 6	displaystyle / teoría / lógica / matemática
T 7	tratamiento / cirugía / médico / paciente / sindrome
T 8	ecuación / ingeniería / inteligencia / artificial
T 9	política / análisi / marketing / rama / arteria
T 10	industrial / industria / plano / internacional
Order	[4, 3, 9, 8, 7, 5, 10, 6, 2, 1]

The third method, the one that uses *rake-nltk* for providing keywords to the Wikipedia articles used for training and LDA for extracting transcript tags, provides a few labels, but they are very domain-specific. The tags that can be resulted from this method come from a relatively small set of possible tags (this set is formed by the most commonly used terms in the 3 domains of our clusters), so the most relevant tags from this set will be chosen.

This is an advantage for our semi-supervised method as we can provide good results with a relatively small data-set for training. The words used for tags by this method are very likely to be well categorised by the semi-supervised method as they are very common only in the are of one of the three domains.

5. CONCLUSIONS

This paper has presented a method which combines the extraction of keywords from a Wikipedia data-set with the automatic classification of learning objects using LDA to obtain better keywords for searching educational videos. This will allow students to find more accurate resources for videos that have not been appropriately tagged by authors.

Using Wikipedia for creating a labelled data-set has allowed us to build a balanced set of articles that have been used to train a model for extracting keywords from educational video transcripts. However, in future works, it would be interesting to provide an automatic mechanism for building balanced training data-sets.

The proposed has been tested using a real environment, concretely the video lectures sharing website of the *Universitat Politècnica de València*, which has more than 55.000 short videos mainly in Spanish. Results have shown the benefits of this proposal for classifying learning objects into categories (specifically Biology&Sciences, Engineering and Humanities&Arts), which will help students in their search of appropriated learning resources.

Future works should focus on improving accuracy of the classification especially for the classes with poorer results, that is *Engineering* and *Humanities & arts* as *Biology* transcripts are correctly classified. The obtained classifier may be further used for labeling new videos that may be added into UPV Media site.

6. REFERENCES

[1] L. AlSumait, D. Barbará, and C. Domeniconi. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *2008 eighth IEEE international conference on data mining*, pages 3–12. IEEE, 2008.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Lda(latent dirichlet allocation). In *Advances in neural information processing systems*, pages 601–608, 2002.

[3] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[4] C. De Medio, F. Gasparetti, C. Limongelli, F. Sciarrone, and M. Temperini. Automatic extraction of prerequisites among learning objects using wikipedia-based content analysis. In *International conference on intelligent tutoring systems*, pages 375–381. Springer, 2016.

[5] D. Dessì, G. Fenu, M. Marras, and D. R. Recupero. Leveraging cognitive computing for multi-class classification of e-learning videos. In *European Semantic Web Conference*, pages 21–25. Springer, 2017.

[6] S. Downes. Learning objects: resources for distance education worldwide. *The International Review of Research in Open and Distributed Learning*, 2(1), 2001.

[7] S. Lee and H.-j. Kim. News keyword extraction for topic tracking. In *2008 Fourth International Conference on Networked Computing and Advanced Information Management*, volume 2, pages 554–559. IEEE, 2008.

[8] L. Lemnitzer, C. Vertan, A. Killing, K. Simov, D. Evans, D. Cristea, and P. Monachesi. Improving the search for learning objects with keywords and ontologies. In *European Conference on Technology Enhanced Learning*, pages 202–216. Springer, 2007.

[9] M. Litvak and M. Last. Graph-based keyword extraction for single-document summarization. In *Proceedings of the workshop on Multi-source Multilingual Information Extraction and Summarization*, pages 17–24. Association for Computational Linguistics, 2008.

[10] T. Liu, N. L. Zhang, and P. Chen. Hierarchical latent tree analysis for topic detection. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 256–272. Springer, 2014.

[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. sklearn. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[12] S. Rose, D. Engel, N. Cramer, and W. Cowley. Automatic keyword extraction from individual documents. *Text mining: applications and theory*, 1:1–20, 2010.

[13] V. B. Sharma. Rapid automatic keyword extraction algorithm using nltk. https://pypi.org/project/rake-nltk, 2019.

[14] A. S. Stoica, S. Heras, J. Palanca, V. Julian, and M. C. Mihaescu. A semi-supervised method to classify educational videos. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 218–228. Springer, 2019.

[15] B. N. Subudhi, T. Veerakumar, S. Esakkirajan, and S. Chaudhury. Automatic lecture video skimming using shot categorization and contrast based features. *Expert Systems with Applications*, page 113341, 2020.

[16] S. Unankard and W. Nadee. Topic detection for online course feedback using lda. In *International Symposium on Emerging Technologies for Education*, pages 133–142. Springer, 2019.

[17] K. Zhang, H. Xu, J. Tang, and J. Li. Keyword extraction using support vector machine. In *international conference on web-age information management*, pages 85–96. Springer, 2006.

[18] G. Zhao, Y. Liu, W. Zhang, and Y. Wang. sklearn.decomposition.latentdirichletallocation. In *Proceedings of the 2018 2nd International Conference on Management Engineering, Software Engineering and Service Sciences*, pages 188–191, 2018.