Pattern Recognition Letters 128 (2019) 85–92

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Integrating word embeddings and document topics with deep learning in a video classification framework

Zenun Kastrati[a,*], Ali Shariq Imran[b], Arianit Kurti[a]

[a] Linnaeus University, P G Vejdes väg, Växjö 351 95, Sweden
[b] Norwegian University of Science and Technology, Trondheim NO-7491, Norway

ARTICLE INFO

Article history:
Received 28 February 2019
Revised 15 July 2019
Accepted 20 August 2019
Available online 21 August 2019

Keywords:
Deep learning
Video classification
Embedding
Document topics
CNN
DNN

ABSTRACT

The advent of MOOC platforms brought an abundance of video educational content that made the selection of best fitting content for a specific topic a lengthy process. To tackle this challenge in this paper we report our research efforts of using deep learning techniques for managing and classifying educational content for various search and retrieval applications in order to provide a more personalized learning experience. In this regard, we propose a framework which takes advantages of feature representations and deep learning for classifying video lectures in a MOOC setting to aid effective search and retrieval. The framework consists of three main modules. The first module called pre-processing concerns with video-to-text conversion. The second module is transcript representation which represents text in lecture transcripts into vector space by exploiting different representation techniques including bag-of-words, embeddings, transfer learning, and topic modeling. The final module covers classifiers whose aim is to label video lectures into the appropriate categories. Two deep learning models, namely feed-forward deep neural network (DNN) and convolutional neural network (CNN) are examined as part of the classifier module. Multiple simulations are carried out on a large-scale real dataset using various feature representations and classification techniques to test and validate the proposed framework.

1. Introduction

The rapid advances of technologies and overall digitalisation trends, constantly have brought up new demands for new skills and expertise of the professionals in IT industry. The high pace of changes within IT landscape makes rather difficult for university educational program to keep up these demand. Despite the changes, universities to the large extend are still "increasingly stove-piped in highly specialized disciplinary fields" [6] as well as there is a lack of flexibility for the professionals to have their competences developed further. In these settings, the real challenge is how to find the right persons with the right education in an industry where the in-thing yesterday may be out-of-date tomorrow? The ability for establishing personalized learning trajectories and their benefits for education has been also highlighted in the recent work done by FitzGerald et al. [4]. One of the main benefits of personalized learning is the possibility for increased learners effectiveness as in these settings the learning process is steered by learners themselves.

The emergence of massive open online courses (MOOCs) gained a lot of attraction especially since they brought up new possibilities when it comes to processes ability for learners to "pick and choose" the educational content they would like to consume [2]. Despite the large number of benefits that MOOCs brings to the educational institutions [5], there are still issues to be considered regarding their sustainability. In the research conducted by Tirthali [20] is suggested that realizing sustainability of MOOCs also depends on instructional strategies involved and orchestration of the content with the learning activities.

Despite these challenges, still the number of MOOCs, students enrolled, and institutions providing them is increasing steady. The statistics provided by Class Central indicate that the number of MOOCs in the past 4 years has increased exponentially and now counting almost 11,500 courses.[1]

Research conducted by Stöhr et al. [19] indicates that video lectures are the key component of MOOCs. Furthermore, they suggest that increase consumption of video lectures is directly correlated with the performance of the learners. Anyhow in the ocean of video lectures available in MOOC platforms it can be rather tiresome to identify the best fitting content for a specific topic. The

* Corresponding author.
E-mail address: zenun.kastrati@lnu.se (Z. Kastrati).

[1] https://www.class-central.com/report/mooc-stats-2018/.

https://doi.org/10.1016/j.patrec.2019.08.019
0167-8655/

course categories are rather general thus it can be time consuming to identify the best content for personalized learning trajectories. In this aspect, in this paper we report our research efforts of how the use of applied machine learning approaches can be used to support the content categorization of video lectures from MOOC courses. We have created a data set of video lectures and applied different feature representation and machine learning techniques within a proposed classification framework in order to assess the performance and feasibility of these approaches for content classification.

The contribution of this paper is:

1. Collection of a large-scale video lectures dataset consisting of 12,032 videos from 200 courses belonging to 40 fine-grained subject categories, presented in Section 4. The dataset[2] constituted of transcript feature representations is made open and available for the public in order to promote the research in this field.
2. A video classification framework which utilizes various transcript representation techniques including Bag-of-Words (BoW), document topics/themes, and word embeddings, i.e. embeddings generated from our MOOC dataset and transfer learning using state-of-the-art pre-trained word embeddings.
3. Performance analysis of various input feature representations and classification techniques including deep networks and conventional classifiers.

The rest of the paper is structured as following. Section 2 presents the state-of-the-art when it comes to related work. A video classification framework is proposed and described in Section 3 followed by Section 4 that describes the dataset collection procedure and presents the statistics in detail. Results and their analysis are presented in Section 5. Lastly, Section 6 concludes the article.

2. Related work

Open educational video resources has gained popularity in the last decade with a massive growth in eLearing and MOOC platforms. Massive amount of video lectures are uploaded on a daily basis that has created a need for efficient structuring and classification of educational resources into respective categories for easy search and retrieval. Imran and Cheikh [8] first proposed the multimedia learning object framework for video lectures that opened up the niche for utilizing both implicit and explicit metadata obtained from textual content and non-textual cues for content organizing, structuring, and classification of video learning objects. A number of classification approaches as a result with respect to lecture videos have evolved over the years, most of which make use of natural language processing (NLP) either directly on the accompanying audio transcript as in [9], or extracted automatically from the lecture images employing OCR as in [7,11,17,18] or via speech-to-text such as in [3].

Researchers have addressed the video lecture classification problems from three different perspectives, i.e., (a) intended application domain, (b) features exploited such as textual, non-textual, and (c) classification techniques employed. For instance, Dessì et al. [3] studied four feature representations including *tf*idf, concepts, keywords*, and a combination *concepts+keywords* using four conventional machine learning classifiers, namely decision tree, support vector machine (SVM), random forest, and SVM using stochastic gradient descent for classifying lecture videos. All the features are extracted using NLP from speech-to-text generated transcripts. Similar work was carried out by Othman et al. [14]. The authors proposed a framework for classifying 22 web MOOC video metadata instances by extracting the metadata associated with the videos via the XML platform. The authors in [15] later applied two shallow machine learning techniques, namely decision tree, and naive Bayesian.

Chatbri et al. [1] in their paper titled automatic MOOC video classification using transcript features and convolutional neural networks proposed a deep neural network (DNN) classifier based approach consisting of three steps: (i) video transcript is generated using speech recognition, (ii) the transcript is converted into an image representation using a statistical co-occurrence transformation, and (iii) a CNN model is trained on a 2545 videos dataset from Khan Academy.

To the best of our knowledge not much work can be found in the literature with respect to embeddings and document themes/topics representation approach in the MOOC domain for classifying video lectures into predefined categories. The novelty of this paper is a classification system which takes advantage of the strengths of both transcript representation approaches and deep learning to improve the performance of video lectures classification. From classification technique perspective, we employ CNN model and perform a comprehensive comparative evaluation with DNN and shallow machine learning techniques.

3. Proposed framework

Fig. 1 shows the high-level system diagram of the proposed framework depicting the MOOC platform as the data source containing video lectures, corresponding caption (*.VTT*) files, and the general and fine-grained level category labeling. The framework comprises of three main modules which are discussed in the following subsections.

3.1. Pre-processing

Two methods are proposed as part of the pre-processing steps in this study to obtain the lecture transcripts for cases where they are not readily available. Video lectures (**.mp4*) collected from the Coursera MOOC platform, in the first step, are converted into audio files (*.wav*) using *FFmpeg* which are then processed with speech-to-text API to obtain transcripts. Audio files are down-sampled to 16KHz with 16 bits with a single channel. A google speech recognition API implemented in CMU Sphinx library in Python is then used to obtain the audio transcripts. The preliminary experiments have shown a very high accuracy with a word error rate (WER) of less than 5% when compared to *.VTT* files. In the second case, video lectures go through a text analysis process as suggested in [7,11,17,18], where a series of further pre-procesing steps are applied followed by optical character recognition (OCR) to extract all kind of text including handwritten, machine-printed, horizontal and non-horizontal oriented text, etc., from video frames as shown in Fig. 2. For the study carried out in this paper, we assume that a *.VTT* file or a lecture transcript is readily available from which the feature representations can easily be extracted.

Not all the words occurring in a transcript are important in terms of classification. Some words are more discriminating than the others so there is a need to evaluate their discriminative power. This is achieved by assigning a weight to each word. Prior to assigning weights, some pre-processing steps must be taken in order to remove the noise from text in a lecture transcript. These steps primarily include removing punctuation and words that are not purely comprised of alphabetical characters, converting uppercase characters to lower-case, removing of stop words and words with length less than or equal to one character.

[2] Contact the authors via e-mail to providing the dataset.

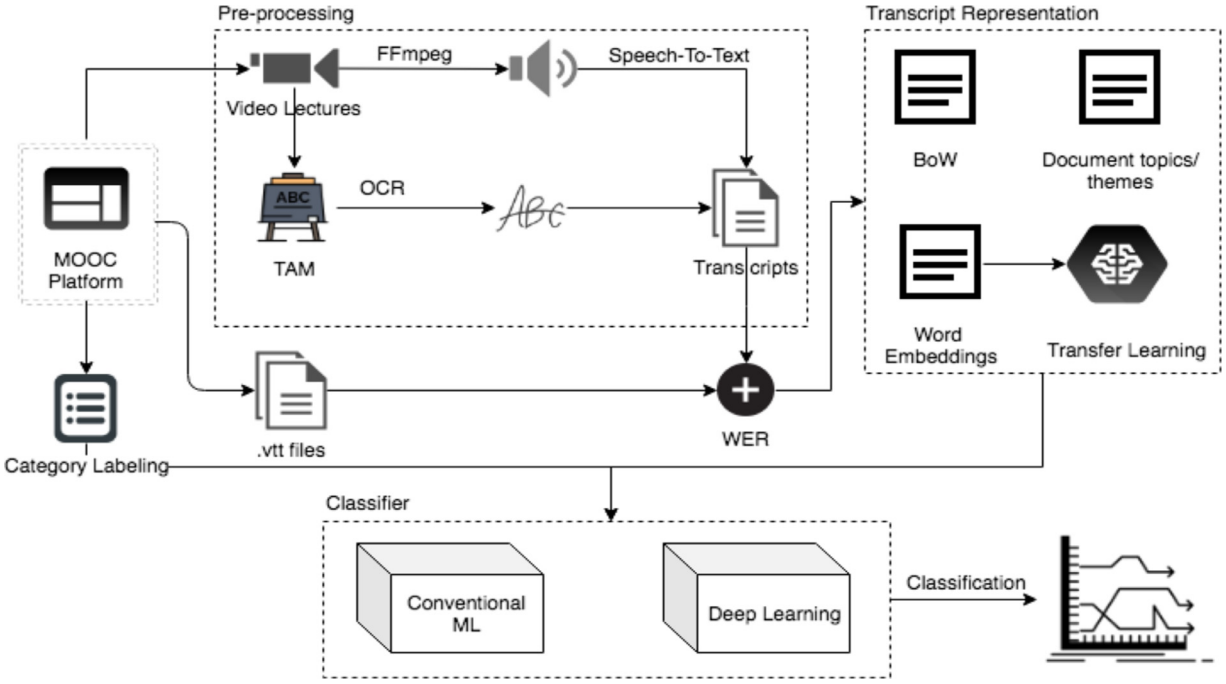
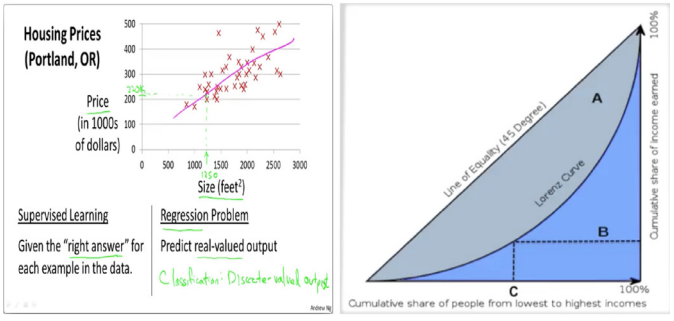


Fig. 1. Proposed video classification framework.



(a) Handwritten & machine-printed (b) Horizontal & non-horizontal text

Fig. 2. Video frame samples taken from MOOC dataset.

3.2. Transcript representation

A vector space model (VSM) is employed for preparing and transforming the text in lecture transcripts to a numerical format, so that it can be processed by machine learning techniques. In a VSM, each text document is represented as a vector composed of words appearing in that document and their corresponding weights. Words are located and extracted from a transcript through a breakdown process. This process known as tokenization splits the text in smaller pieces referred as tokens or words. We studied four feature representation techniques as part of this module.

3.2.1. Bag-of-Words (BoW)

There exists various weighting schemes to computing and assigning weights to words but a so called bag-of-words is the most commonly used. This model relies on distributional feature of words and is very simple to implement. It can be implemented in two ways: count occurrence - *tf* and term frequency inverse document frequency - *tf*idf*. The former relies on word occurrences to show the importance of words in a document while the later measures the relevance of words using two components: *tf* that reflects the importance of words in a document, and *idf* that shows the distribution of those words among the collection of documents. Representing texts using both BoW implementations is a transformation that typically produces large sparse vectors comprised mostly of zero values.

3.2.2. Embedding

A word embedding is a word (text document) representation technique which uses dense vector representations. These vectors are comprised of continuous real values learned from text corpora and are of fixed sizes. Each word is associated with a position (value) in the vector space. The position of the word is defined by its surrounding words and this allows to capture context in which words occur and makes word embeddings more expressive representation technique. Additionally, syntactic and semantic relationship between words can be captured using contextual similarity defined by cosine similarity distance between word embeddings.

3.2.3. Transfer learning

Transfer learning is a machine learning technique in which a model trained on a completely different task is used for the new task of interest. Transfer learning can be used as feature extractor by simply training a deep network architecture with no output layer in a very large dataset. In the case when transfer learning is used as an initialization, a model is primarily trained on large readily available datasets for discovering and learning patterns in the data appearing in these datasets. The learned patterns are then used as input features to train smaller network architectures to learn the relations for the new applied problem. In this research work, transfer learning is used as an initialization where embedings of three state-of-the art pre-trained models, namely Word2Vec [13], GloVe [16], and fastText [12], are employed as an input to our proposed deep learning models.

3.2.4. Topic modeling

A Latent Dirichlet allocation (LDA) topic modeling approach is also used in this study. It is a generative statistical model which considers each document as a mixture of a small number of topics/themes and that each word's presence is attributable to one

Table 1
Size of the dataset covering general-level categories.

No	Category	# of courses	# of videos	Duration	# of tokens	FG
1	Art and Humanities	15	915	4d 17h 23m 11s	6,277,240	
2	Physical Sciences and Engineering	29	2208	12d 19h 01m 27s	14,695,682	
3	Computer Science	25	1591	7d 19h 17m 15s	10,986,475	
4	Data Science	18	1037	5d 04h 26m 06s	6,110,894	
5	Business	30	1569	8d 02h 40m 28s	9,742,055	
6	Information Technology	23	1048	5d 00h 45m 02s	5,187,491	
7	Health	40	2191	15d 13h 06m 32s	16,197,086	
8	Social Sciences	20	1473	8d 02h 33m 48s	10,483,221	
	Total	200	12032	67d 07h 04m 49s	79,680,144	

of the document's topics. Document topics generated from topic modeling are used as input feature representations to feed and train the machine learning techniques in the classifier module.

3.3. Classifier

The last module of the proposed framework is a classifier that aims at assigning a given video lecture to the most appropriate category. Classifier is a mapping from transcript feature representations $f_r(t(x_i))$ to a finite set of class labels c_i. This mapping can be formally defined as a function $f_r(t(x_i)) \rightarrow c_i$. This module is classifier-independent because it is not linked to a specific classifier. It handles both conventional machine learning and deep learning classifiers.

4. Dataset

A real-world dataset from the education domain to conduct the experiments and validate the proposed classification framework is collected from scratch. The dataset consists of 12,032 videos collected from 200 courses on a MOOC platform called Coursera.[3] The total duration of the videos is 1615.08h (67d 07h:04m:49s). The shortest video is 18s while the longest one is 1h:09m:53s. The average duration of videos is 08m:03s (std: 325). Each video lecture is accompanied with its corresponding transcript. All collected videos and their corresponding transcripts are in English language. Coursera uses a 2-level hierarchical structure composed of general-level and fine-grained level to categorizing courses. The same course categorization structure is used in our case to creating the dataset. Each downloaded video is assigned to one fine-grained category and one general-level category. The dataset is comprised of 8 general-level categories and the distribution statistics of each category including number of courses, number of videos (transcripts), duration of the videos, number of tokens, and their color encoded fine-grained categories (FG), are depicted in Table 1. As can be seen in Table 1, the number of video transcripts in each category varies widely, ranging from the Physical Sciences and Engineering category that contains 2208 video transcripts to the Art and Humanities category that covers only 915 video transcripts. The total number of tokens occurring in this dataset is 79,680,144.

The length of video transcripts constituting our dataset varies greatly from 228 words to 32,767 words, with an average of 6622 words per transcript. Transcripts length variation is illustrated in Fig. 3, in which the box plots show the number of words per transcript distributed among general-level categories of the entire corpus. It is Health category which characterizes with the longest

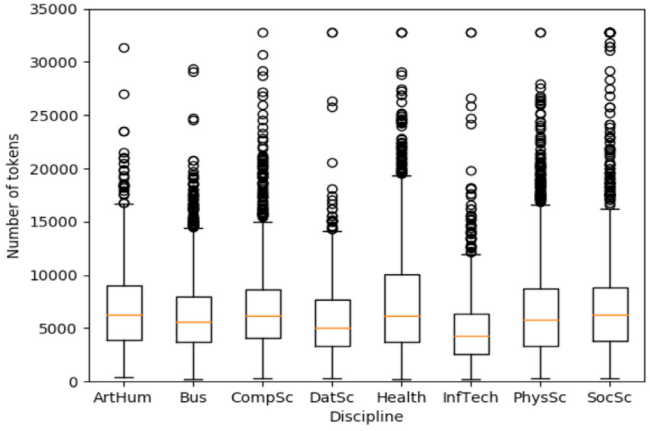


Fig. 3. Length of video transcripts among general-level categories.

transcripts. 25% of transcripts in this discipline are comprised of more than 20,000 words. On the other side, Information Technology (InfTech) category is the most compact discipline in terms of transcripts length in our corpus covering more than 75% of transcripts constituted of less than 6000 words.

In line with the course structure catalog of Coursera, each downloaded video is assigned to one or more specific categories denoted as fine-grained subject categories. Our dataset is comprised of 40 such categories which along with the video transcripts distribution statistic are illustrated in Fig. 4. As illustrated in Fig. 4, fine-grained categories are grouped into 8-color bars corresponding to the 8 general-level subject categories given in Table 1.

5. Results and analysis

In this section, we investigate the performance of our proposed framework on the collected MOOC dataset. For training the classifiers of the framework, we divide the dataset arbitrary into three subsets: training 70%, testing 15% and validation 15%. To evaluate and validate classifiers' performance, measures like macro-averaged (macro) and weighted-averaged (weight) precision, recall, and F1 score, are used.

5.1. Network configurations exploration

In order to investigate which CNN architecture performs better on classifying videos from our dataset, we examined the effect of the depth and width of architecture with respect to accuracy. In particular, we run several simulations using different depth (layers)

[3] http://www.coursera.org.

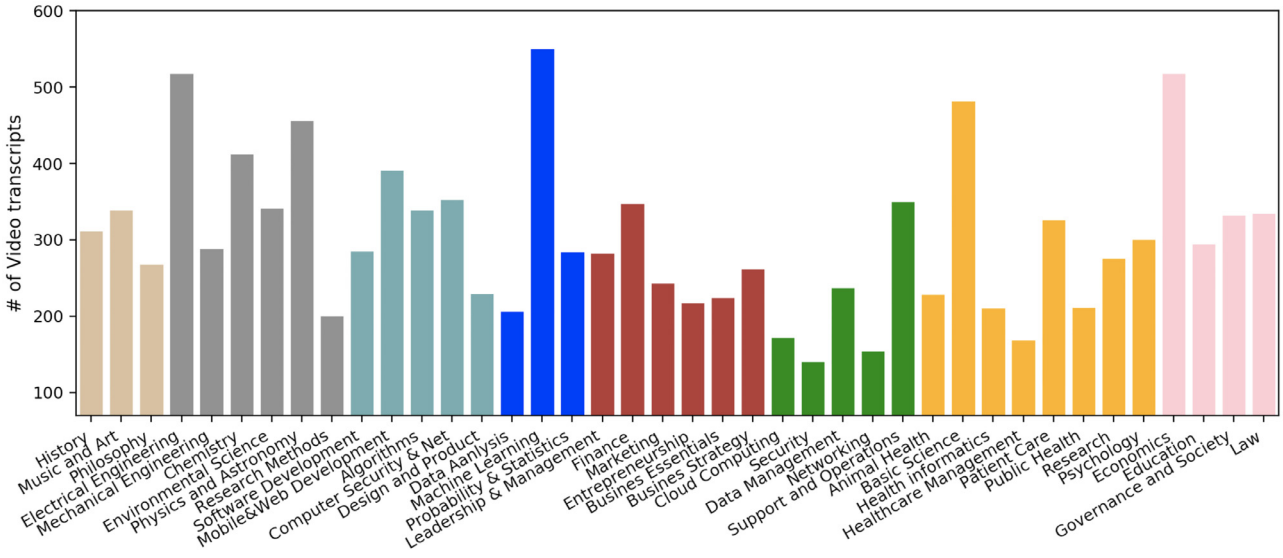


Fig. 4. Distribution of video transcripts among fine-grained categories.

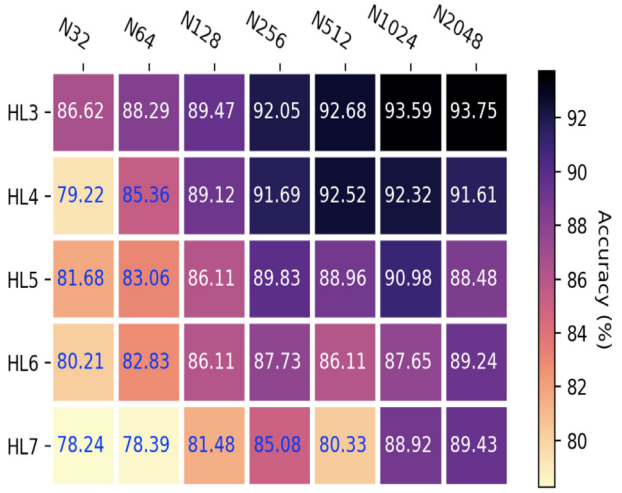


Fig. 5. Heat map of CNN validation accuracy with respect to hidden layers and nodes per layer.

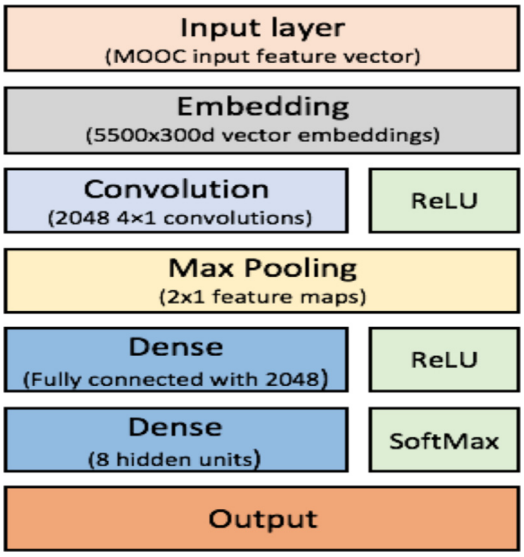


Fig. 6. Architecture design of CNN.

and width (nodes) configurations and the obtained experimental results are illustrated in Fig. 5.

It is apparent from the heat map shown in Fig. 5 that depth and width have the opposite effects on the classification performance of the CNN. More precisely, widening consistently improves performance across CNN architectures of different layers while depthening continuously decreases performance. The best classification accuracy nevertheless is achieved by an architecture design comprising of an input, an output and five hidden layers, as shown in Fig. 6. The input is the first layer composed of transcripts word vectors followed by an embedding layer which generates corresponding word embeddings for the input vocabulary. A complex one-dimensional convolution layer with 2048 units and filters of size four is applied followed by a global max pooling layer that selects the maximum value of its respective convolutional layer outputs. The next two hidden layers are fully connected dense layers containing 2048 and ReLU, and 8 units and SoftMax activation functions, respectively. The final layer of the network is the output which contains 8 units corresponding to 8 general-level categories of the MOOC dataset. Due to its performance and the better applicability to real-world scenarios of CNN [21], we chose this architecture for running other simulations in the paper.

In the same fashion, we conducted simulations with DNN to optimize the network architecture. In contrast to CNN, we observed that the performance of DNN consistently improves as the number of hidden layers increases while increasing the number of nodes per layer causes its performance to decrease. Based on the simulations, we found that an architecture design with a basic structure consisting of an input, an output and seven hidden layers yields the best classification accuracy for DNN. The input is fed with transcripts word vectors followed by an embedding layer containing corresponding word embeddings for the input vocabulary. The next layer applies flattening to convert data into 1D for feeding the following layer. The next four layers are fully connected dense layers with 256 units and ReLU activation functions. The output is preceded by a fully connected dense layer containing 8 units corresponding to the 8 classes of the dataset and SoftMax.

Table 2
Performance of DNN using BoW representation.

Model	(%)	General-level		Fine-grained	
		macro	weight	macro	weight
tf	Pr	94.39	94.56	91.13	92.40
	R	94.32	94.50	90.88	91.93
	F1	94.35	94.53	91.00	92.16
tf*idf	Pr	94.81	95.04	91.83	92.71
	R	94.83	95.01	91.45	92.60
	F1	94.82	95.02	91.64	92.65

Table 3
Performance of CNN and DNN using MOOC embeddings.

Model	(%)	General-level		Fine-grained	
		macro	weight	macro	weight
CNN	Pr	93.52	93.77	86.41	88.00
	R	93.50	93.75	86.18	87.73
	F1	93.51	93.76	86.29	87.86
DNN	Pr	85.26	87.06	63.47	67.50
	R	86.02	86.31	60.57	67.00
	F1	85.64	86.68	61.99	67.25

5.2. Classification using BoW

We conducted the first simulations using BoW representation. In particular, two BoW implementations, namely count occurrence *tf* and *tf*idf* are employed as feature representations to feed the DNN model and the obtained results are summarized in Table 2.

5.3. Classification using MOOC embeddings

In this section we initially generated a package of word embeddings from education domain. To achieve this, we trained and learned the embeddings on our MOOC corpus comprised of 79 million tokens (words) with a vocabulary of 68,175 words. Vocabulary consists of unique words that are obtained after preprocessing steps which primarily include removing punctuation and words that are not purely comprised of alphabetical characters, converting upper-case characters to lower-case, removing of stop words and words with length less than or equal to one character. We generated word embeddings of different vector sizes, including 50, 100, 200, and 300 dimensions. For reasons of space, we use word embeddings with 300 dimensions to train our CNN and DNN models in this paper. The experimental results given in Table 3 show that CNN model significantly outperforms DNN model in both cases of testing, i.e., general-level and fine-grained categories.

5.4. Classification using transfer learning

We extended the experiments with embeddings by using transfer learning, that is, pre-trained models trained on corpora comprised of billions of words. In this paper, we have used pre-trained word embeddings generated by three well known state-of-the-art pretrained models, namely Word2Vec (W2V), GloVe, and fastText (fText). Word2Vec comprises of word embeddings for a vocabulary of 3 million words trained on 100 billion tokens from a Google news dataset. GloVe contains word embeddings for a vocabulary of 400K words trained on 42 billion words from Wikipedia pages and newswire, and fastText includes word embeddings for a vocabulary of 2 million words trained on 600 billion tokens from Common Crawl. Word embeddings with 300 dimensions of all three models are used to train CNN and DNN models and the obtained results are given in Table 4.

We observe from Table 4 that CNN performs significantly better than DNN when using pre-trained word embeddings generated by all three models. The performance gap between these two classifiers is reflected even more when testing on the fine-grained categories.

Table 4
Performance of CNN and DNN using pre-trained embeddings.

Mod	Emb	(%)	General-level		Fine-grained	
			macro	weight	macro	weight
CNN	W2V	Pr	87.40	87.90	77.72	80.10
		R	87.48	87.77	77.27	80.09
		F1	87.44	87.83	77.49	80.09
	GloVe	Pr	88.67	89.08	84.76	86.00
		R	88.56	88.92	83.78	85.79
		F1	88.61	89.00	84.27	85.89
	FText	Pr	91.46	91.84	87.47	88.56
		R	91.64	91.81	86.85	88.41
		F1	91.55	91.82	87.16	88.48
DNN	W2V	Pr	46.55	46.64	21.44	23.14
		R	40.56	43.29	18.37	21.13
		F1	43.35	44.90	19.79	22.09
	GloVe	Pr	66.37	65.12	36.09	39.35
		R	60.04	61.89	26.19	30.27
		F1	63.05	63.46	30.64	34.22
	FText	Pr	71.80	73.67	48.42	50.21
		R	70.45	70.76	40.11	46.38
		F1	71.12	72.19	43.87	48.22

Table 5
Performance of CNN and DNN using document topics/themes.

Model	(%)	General-level		Fine-grained	
		macro	weight	macro	weight
CNN	Pr	81.97	82.19	72.88	75.14
	R	81.41	81.92	72.34	74.48
	F1	81.69	82.05	72.61	74.81
DNN	Pr	84.15	84.00	75.92	78.32
	R	82.94	83.56	75.96	77.56
	F1	83.54	83.78	75.94	77.94

5.5. Classification using topic modeling

One approach proposed in this paper is an LDA topic model using different number of topics. We started with a LDA model with eight document topics corresponding to eight general-level subject categories and continued up to the 300 topics which is an analogue to the dimensions of word embeddings generated from MOOC and pre-trained methods. Document topics generated from the LDA model for both general-level subject categories and fine-grained categories are used as input feature vectors to train our CNN and DNN models. For reasons of space, we have shown in Table 5 the results of CNN and DNN models achieved by using only 300 document topics.

As can be seen in Table 5, DNN and CNN using document topics as input feature representations perform almost the same, with a slight advantage of DNN. These findings suggest that document topics, unlike other feature representations, fit and work pretty well with different classifiers.

5.6. Performance of conventional ML on MOOC dataset

Lastly, we investigated the performance of some of the most common conventional machine learning techniques on our collected MOOC dataset. Four different supervised classifiers including support vector machine (SVM), decision tree (DT), naive Byes (NB), and XGBoost (Boost) are used to conduct experiments. Results summarized in Table 6 show that SVM performs pretty well on MOOC dataset achieving high accuracy on general-level and

Table 6
Performance of conventional ML on MOOC dataset.

Model	(%)	General-level		Fine-grained	
		macro	weight	macro	weight
SVM	Pr	94.55	94.50	92.76	92.94
	R	94.39	94.49	91.92	92.83
	F1	94.47	94.49	92.34	92.88
DT	Pr	64.73	65.75	55.04	57.44
	R	64.63	65.60	54.56	57.45
	F1	64.68	65.67	54.80	57.44
NB	Pr	85.35	81.92	79.95	78.39
	R	72.55	77.84	47.03	56.07
	F1	78.43	78.43	59.22	65.38
Boost	Pr	88.20	87.53	87.36	87.75
	R	86.77	87.42	86.11	87.62
	F1	87.48	87.47	86.73	87.68

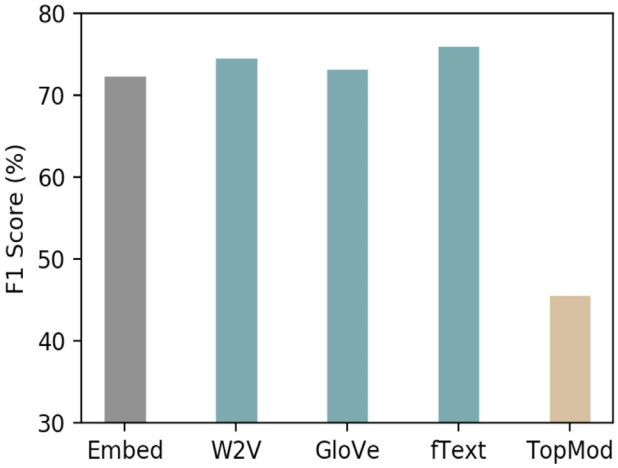


Fig. 7. Performance of CNN on the micro-learning video dataset.

fine-grained categories. Boost also works well with our dataset and it is interesting to note that it is the only classifier that yields almost the same performance on both cases of testing, with a very slight advantage on fine-grained.

5.7. Classification performance on micro-videos dataset

The findings provided above validate the usefulness of proposed classification framework as an effective means to classifying videos but to emphasize its validity, we extended our simulations using a public available dataset [10] comprised of micro-learning videos (1–3 min) collected from Coursera. The dataset contains 3739 videos and their corresponding transcripts of sentence lengths extracted from 128 video lectures from six different disciplines. For comparative analysis of performance, all the simulations are conducted under the same conditions in terms of feature representation techniques and architecture design of the classifier as given in Section 5.1. Results with respect to weighted average F1 score obtained by CNN fed with embeddings generated by using solely the dataset (Embed), transfer learning (W2V, GloVe, fText) and topic/theme modeling (TopMod), are shown in Fig. 7.

As can be seen from the bar chart the proposed classification framework performs well also in classifying micro-videos when word embeddings are learned using either an input text corpus, or pre-trained word embedding models. On the contrary, a significant performance degradation is shown when document topics are used as input feature representations. This can be explained by the fact that micro-video transcripts are characterized by very limited co-occurrence of words (sparseness problem) which can not be handled properly by topic modeling technique like LDA.

6. Conclusion

In this paper, we proposed a video classification framework that exploits various transcript feature representations with deep learning for content classification and organization within a MOOC setting. The framework consisted of three main modules including pre-processing, transcript representation, and classifier. The proposed framework is tested and validated on a large scale real-world dataset collected from Coursera platform for this purpose. The dataset is comprised of videos transcripts categorized into two levels including general-level categories and fined-grained ones. Experimental results obtained from all classifiers (except Boost) employing various feature representations showed that much better classification performance is achieved when using general-level categories than specific-level one. This could be explained by the fact that specific-level categories have very similar characteristics (class overlap) and thus there is needed subtle details to differentiate between them.

To further our research, we are planning to investigate other transcript representation techniques like cognitive computing which aims to extract high level feature representations i.e. concepts. Furthermore, the proposed framework relies on readily available lecture transcripts to exploit textual features thereby it does not utilize any video/audio to text conversions. Future studies on the current topic are therefore suggested in order to establish a video classification framework in which pre-processing can be extended to consider transcripts provided by converting video/audio and image to text.

Declaration of Competing Interest

None.

References

[1] H. Chatbri, K. McGuinness, S. Little, J. Zhou, K. Kameyama, P. Kwan, N.E. O'Connor, Automatic MOOC video classification using transcript features and convolutional neural networks, in: Proc. of the ACM Workshop on Multimedia-based Educational and Knowledge Technologies for Personalized and Social Online Training, ACM, 2017, pp. 21–26.
[2] B. Dasarathy, K. Sullivan, D.C. Schmidt, D.H. Fisher, A. Porter, The past, present, and future of moocs and their relevance to software engineering, in: Proc. of the on Future of Software Engineering, ACM, 2014, pp. 212–224.
[3] D. Dessì, G. Fenu, M. Marras, D.R. Recupero, Bridging learning analytics and cognitive computing for big data classification in micro-learning video collections, Comput. Hum. Behav. (2018).
[4] E. FitzGerald, A. Jones, N. Kucirkova, E. Scanlon, A literature synthesis of personalised technology-enhanced learning: what works and why, Res. Learning Technol. 26 (2018).
[5] F.M. Hollands, D. Tirthali, Why do institutions offer moocs? Online Learn. 18 (3) (2014) 1–19.
[6] G. Hurlburt, J. Voas, K. Miller, P. Laplante, B. Michael, A nonlinear perspective on higher education, Computer 43 (12) (2010) 90–92.
[7] A.S. Imran, S. Chanda, F.A. Cheikh, K. Franke, U. Pal, Cursive handwritten segmentation and recognition for instructional videos, in: Eighth International Conference on Signal Image Technology and Internet Based Systems (SITIS), IEEE, 2012, pp. 155–160.
[8] A.S. Imran, F.A. Cheikh, Multimedia learning objects framework for e-learning, in: International Conference on e-Learning and e-Technologies in Education (ICEEE), IEEE, 2012, pp. 105–109.
[9] A.S. Imran, L. Rahadianti, F.A. Cheikh, S.Y. Yayilgan, Semantic tags for lecture videos, in: IEEE Sixth International Conference on Semantic Computing (ICSC), IEEE, 2012, pp. 117–120.
[10] Z. Kastrati, A.S. Imran, A. Kurti, Transfer learning to timed text based video classification using cnn, in: Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics, in: WIMS2019, ACM, 2019, pp. 23:1–23:9.
[11] H. Li, D. Doermann, O. Kia, Automatic text detection and tracking in digital video, IEEE Trans. Image Process. 9 (1) (2000) 147–156.
[12] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, A. Joulin, Advances in pre-training distributed word representations, in: Proc. of the International Conference on Language Resources and Evaluation (LREC'18), 2018, pp. 52–55.
[13] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proc. of the 26th International Conference on Neural Information Processing Systems, in: NIPS'13, 2013, pp. 3111–3119.

[14] E.H. Othman, S. Abdelali, E.B. Jaber, Education data mining: mining MOOCs videos using metadata based approach, in: IEEE International Colloquium on Information Science and Technology, IEEE, 2016, pp. 531–534.
[15] E.H. Othman, G. Abderrahim, E.B. Jaber, Mining MOOCs videos metadata using classification techniques, in: Proc. of the 2nd International Conference on Big Data, Cloud and Applications, ACM, 2017, pp. 94:1–94:6.
[16] J. Pennington, R. Socher, C.D. Manning, Glove: Global vectors for word representation, in: EMNLP, 14, 2014, pp. 1532–1543.
[17] N. Sharma, P. Shivakumara, U. Pal, M. Blumenstein, C.L. Tan, A new method for arbitrarily-oriented text detection in video, in: 2012 10th IAPR International Workshop on Document Analysis Systems, 2012, pp. 74–78.
[18] N. Sharma, P. Shivakumara, U. Pal, M. Blumenstein, C.L. Tan, Piece-wise linearity based method for text frame classification in video, Pattern Recognit. 48 (3) (2015) 862–881.
[19] C. Stöhr, N. Stathakarou, F. Mueller, S. Nifakos, C. McGrath, Videos as learning objects in moocs: a study of specialist and non-specialist participants' video activity in moocs, Br. J. Educ. Technol. 50 (1) (2019) 166–176.
[20] D. Tirthali, Are moocs sustainable? in: From books to MOOCs? Emerging Models of Learning and Teaching in Higher Education, Portland Press Limited, London, UK, 2016, pp. 115–123.
[21] X. Zhang, J.J. Zhao, Y. LeCun, Character-level convolutional networks for text classification, 2015. CoRR arXiv: 1509.01626.