What should I learn next? Ranking Educational Resources

Victor Connes, Colin de La Higuera, Hoel Le Capitaine

What should I learn next? Ranking Educational Resources

1st Victor Connes
LS2N
Université de Nantes
Nantes, France
victor.connes@univ-nantes.fr

2nd Colin de la Higuera
LS2N
Université de Nantes
Nantes, France
cdlh@univ-nantes.fr

3rd Hoël Le Capitaine
LS2N
Université de Nantes
Nantes, France
hoel.lecapitaine@univ-nantes.fr

***Abstract*—Artificial Intelligence tools can enable a better use of Open Educational Resources (OER) by making it easier to search for OER or to receive high quality recommendations. Ideally, a user should learn from resources suggested in some order consistent with what a teacher may call "progressive". The progress of natural language processing techniques over the recent years allows us to propose a timeline-preserving neural network model whose goal is to be able to rank resources. We experiment with series of lectures organized in courses made by teachers as ground truth. Our method is able to rank correctly 80% of pairs of lectures with a contextual background and 69% in an agnostic setting. Our contribution is completed by the formalization of this task as a machine learning problem, and by the distribution of a new free open data-set.**

***Index Terms*—Open Education, Open Educational Resources, Pedagogic Recommendation, Technologies to Enhance Learning**

I. Introduction

In recent years, we have seen a big increase of available Open Educational Resources (OER). This is in great part due to the success of MOOCs (Massive Online Open Courseware) but also to the political commitment of several governments and of UNESCO's policies in favour of Open Education. Those two factors have led many stakeholders to deploy OER on their own websites, or in joint repositories [**?**], [1]. Project X5-GON has been launched in this context: its goal is to index the different websites (repositories, collections, catalogues) and resources in order to provide user personalized recommendation and navigation through the implicitly created *Global OER Network*. Being able to recommend learning pathways has been identified as a big issue for the future.

Thanks to this emulating context, the global OER network contains millions of OERs, so any system should be designed to interact with millions of requests per second. Highly scalable algorithms for recommendation are today widely used for commercial applications. As an example, YouTube uses a two stages algorithm to provide its recommendation [2], [3]. The first stage consists in using multiple candidates generation algorithms to build a set of interesting resources as candidates for the recommendation: each of these algorithms captures one aspect of similarity between the query video and the candidate video [3], [4]. The second stage consists in ranking the selected candidates using a Multi-gate Mixture-of-Experts deep learning model [5] to efficiently optimize user engagement and user satisfaction objectives. This two stages architecture is largely used by most of the large-scale recommender engines.

The ranking stage of the algorithm raises a problem for our task because it requires being able to define metrics that somehow reflect the positive learning experience. The goal of pedagogic recommendation is to offer the user a satisfying learning experience: it is much harder to translate this into a metric to be maximized. Furthermore, the usual maximized metrics (engagement rate, conversion rate...) require historical interactions of users with the platform (often represented as user-item matrices). However, in order to be computed, these metrics are deeply impacted by the cold start problem. Moreover, some evil side effect of maximizing such metrics are nowadays discussed. In particular, maximizing engagement metrics may lead the algorithm to recommend viral contents such as violence, fake news, and pseudo-scientific [6] or conspiracy theories [7][1]. In commercial applications, it is sufficient to recommend resources one by one on the fly. Rephrasing, we can choose to focus on the short term goal which is to maximize locally our metrics and proceed from one resource to another. One can argue that learning does not have only immediate goals and is more concerned with medium or long-term objectives. Consequently, a challenge is to provide the user with a recommended learning path through several resources. Different distances have been demonstrated as being efficient to recover similar resources [8], [9]. However, long-term recommendation induces to be able to find an order in which the resources are proposed, based on the resource difficulty, but also on the transition coherence between resources. If these distances can label the edges of the Global OER Network, a question remains: how to orient these edges?

In this work we assume that the order of learning defined by teachers in designing a series of lectures is an example of a satisfactory order of pedagogical resource consumption. Unfortunately, the resources in the Global OER network are not always organised in series of lectures. Regarding the amount of resources, the range of domains and the growth speed of this Global OER Network, it would be a huge task to rely on human annotation to direct it. Given the challenges met

[1]https://algotransparency.org/

with the absence of ground truth and the hardness of a labelling task, we suggest to use series of resources built by teachers as ground truth. For this reason, we choose to formalize our problem as a learning problem in which we try to learn a model handling this logical *consumption* order from these sampling data. In order to duplicate the number of training examples we formalize the task as a binary classification of the precedence of resource pairs. In this work, we present a framework to evaluate the ability by a model to capture a logical *consumption* order. Our contribution is three-fold. Firstly we introduce a new evaluation methodology as a proxy for pedagogic recommendation (Section III). Secondly, we provide a new open free data-set (Section V) for pedagogic recommendation. Lastly, we evaluate on this task a standard natural language processing approach as well as a new approach -called TANN - based on an intra-temporal representation of document that we introduce in this paper (Section VII). Finally, we show the validity of this approach by being able to obtain 80% accuracy in the task of predicting the correct order between two unseen resources with contextual information and 69% in a completely agnostic setting (Section VII).

II. Related work

Several papers focus on the case of the recommendation of pedagogical resources. In their meta-review from 2015, Draschler *et al.* present an analysis of recommendation in Technology Enhanced Learning (TEL) over 82 systems from 35 different countries [10]. The majority of related models aim to "support learners by providing new learning content to their current learning process" following the procedure used in the standard fields of application of the recommendation. Nevertheless, some papers were interested in providing long-term recommendations (ie. learning path). But many of them propose approaches that can't scale up, or are specific to a domain or use case, and often suffer from a combination of the three issues [11]. In more recent approaches, long-term recommendation on educational data have been addressed through deep learning approaches [12], [13]. In conclusion of their review, Hernandez-Blanco *et al.* [14] emphasise the recommendation of learning resources in an *informal setting* as a challenge for the future and points out the lack of freely available data-set to address this challenge.

In this paper, we consider that any *resource* (ie. pedagogical content): full courses, course materials, modules, textbooks, streaming videos, etc... can be represented or converted to text using transcription or text extractions techniques. Results in Natural Language Processing (NLP) from the past 10 years brought by machine learning methods have opened up new application areas and provided unprecedented and close to human level results for historical tasks such as automatic transcription and machine translation. This has been the case in a variety of settings including videos from MOOCs or produced as OER [15].

Document representation is also a historical task within NLP, and state of the art approaches use today a latent embedding space to capture the similarities between their textual representations, the latent embedding space are suitable to be used as input of machine learning techniques for addressing high level tasks. One of the most popular approaches -that we are going to use in this paper- Doc2Vec [8], represents an arbitrarily sized piece of text (document, paragraph, sentences) by a dense vector which is trained to predict words distribution in this text fragment.

More recently, alternative methods need more training examples to show up better results. Among them, the most successful ones are the *transformer* approaches [16], directly applicable on raw text by jointly learning the language model with the objective. Nevertheless, these approaches rely on very deep neural network architectures (more than a billion of parameters). Therefore, they require many training examples which are not available typically on this task, and for this reason these are currently not applicable for our problem.

Recurrent Neural Networks (RNN) [17], a natural extension of FNN specifically designed to deal with sequences, represent an interesting neural architecture to use, provided that we can deal with the gradient vanishing problem, which makes them only applicable on small texts (ie. small context size).

The method we propose in this paper (Section IV-B) is based on an RNN architecture and overcomes this issue by cutting the text into chunks (see Section IV-B).

III. Tasks

As explained in Section I the usual approaches for large scale recommendation have several drawbacks. Firstly, and this is the most problematic one, these approaches need a large amount of collected user interactions in order to learn good ranking model for recommendation. In our case, we do not have yet such a history of interactions, so our problem is arguably assimilated to a cold start problem. Secondly, the usual metrics employed in these approaches have been demonstrated to not favor the best pedagogical content but the most viral one. We naturally want to avoid this case, and aim to recommend in the most pedagogical way as possible.

For doing so, we use existing series of lectures made by teachers (data-set details are presented in Section V). More precisely, we aim to predict the logical consumption order of 2 random lectures drawn from a given series. We assume the following hypothesis: the organization of lectures made by teacher is a good example of logical consumption order. Therefore, we used this order as ground truth and a ranking algorithm able to learn this order is a good candidate for pedagogic recommendation, reason why we use this task as proxy for pedagogic recommendation. Finally, we believe that the logical consumption order can be learned outside the scope of a series. Rephrasing, there are some general patterns allowing a better pedagogical continuity. To verify the last assumption, we design three difficulty increasing tasks allowing to measure the quantity of contextual information learned from the given series (Task III-0a and III-0b) and non-contextual information learned from other series seen during the training (Task III-0c). Task III-0c is arguably the most

interesting in our case, because it evaluates the ability of the model to order a completely new series of lectures from a new teacher. Let us detail the three different tasks at hand:

a) Predicting with contextual information about pairs: In the first case, the goal is to predict the order of resources of a known series when a new (unknown) episode is added. Hence, we build a data set of pairs of resources with a ground truth consumption order (i.e.: from the same course), this data set is split into a TRAIN set and a TEST set in such a way that a pair in the TEST set contains **exactly one element** used during training. We call this task an *episode level task*. It is interesting to notice that by itself, this task could be used as a recommendation system in a scenario in which a teacher with a constructed course wants to introduce a new resource into his or her course. The algorithm could then be used to recommend a preferential position for the new resource.

b) Predicting with contextual information about the course: The drawback with the previous task is that an algorithm that completely ignores the characteristics of the new resource in the pair could perform well by just learning the relative positions of the other training) resources. In order to evaluate the difference between such an algorithm and a learner taking advantage of the characteristics of the new resource, we constructed the second task. Once again, we build a data set of pairs of resources with a ground truth consumption order (ie. from the same course); this data set is thus split into a TRAIN set and a TEST set in such a way that a pair in the TEST contains **no element** seen during training. We call this task the *pair level task*.

c) Agnostic task: Finally, we design the last task to be as close as possible to our final goal. In this task we aim to evaluate how well the order generalization can be learned outside the scope of the training series. Rephrasing, we aim to measure the capacity of the model to correctly predict the ranking for episodes belonging to an unseen series. We call this the *agnostic task*. Since we have no knowledge about the tested series, the assumption is that there is enough information in the discourse and the didactic component to infer the order between two episodes from any series.

For this, the TEST and TRAIN sets should be disjoint and no pair in TRAIN shares resources with TEST.

By summarizing in a simplified manner, in the first task, we remove exactly one episode from each series for the test set, and combine this episode with one from the Train set to build a test pair; in the second setting, 2 episodes are removed before training and they constitute the test pair. Finally, in the third setting, an entire series is removed before training and pairs inside this series will be used as test pairs. For practical reasons the 3 protocols above have been relaxed: instead of removing 1 or 2 episodes for our test set, we always remove a fixed portion of 10% of the episodes from each series in protocols 1 and 2. In the same way in each of the tasks we systematically create an additional set for validation purpose. The detailed and exhaustive protocol is described in the Section VI.

IV. THE MODELS

A. Baseline

We choose as baseline a Doc2Vec embedding followed by a Feed-forward Neural Network (FNN). The first step of the baseline approach is to train a global Doc2Vec model on resources. Each resource r^i is represented by a vector $E^i \in \mathcal{R}^{d_e}$, where d_e is the embedding dimension chosen for the Doc2Vec representation.

This new representation directly feeds an FNN containing L layers, followed by a soft-max unit. More formally, for each pair of resources $(r^i,\ r^j)$, an input vector is built by concatenation of their corresponding embeddings E^i, E^j. The overall model maps each pair of resources $(r^i,\ r^j)$ to a floating point value in the unit interval $[0, 1]$. Specifically, the FNN performs the following transformation:

$$h_0^{FNN} = \sigma(W^0[E^i \oplus E^j] + b^0), \quad (1)$$

$$h_l^{FNN} = \sigma(W^l h_{L-1}^{FNN} + b^l), \quad (2)$$

$$\hat{o} = sigmoid(h_L^{FNN}), \quad (3)$$

with $W^0 \in \mathcal{R}^{2d_{inputs} \times d_h^{FNN}}$, $\forall l \in \{1 \dots L-1\} W^l \in \mathcal{R}^{d_h^{FNN} \times d_h^{FNN}}$, $W^L \in \mathcal{R}^{d_h^{FNN} \times 1}$, $\forall l \in \{1 \dots L-1\} b^l \in \mathcal{R}^{d_h^{FNN}}, b^L \in \mathcal{R}$. The operator $\oplus$ denotes the concatenation of the resources embeddings.

B. Our model: The Timeline Aware Neural Network (TANN)

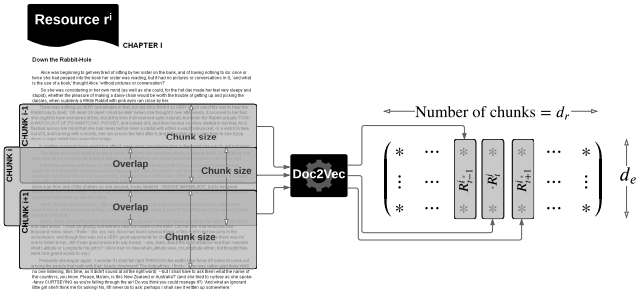
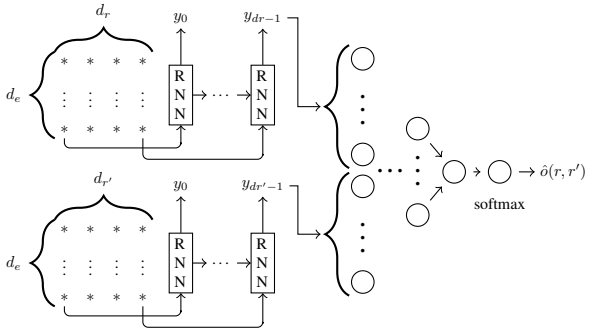


Fig. 1. Procedure for constructing timeline-preserving semantic representation of resources

In order to take into account pedagogical continuity, the construction of discourse and the sequencing of ideas, we propose to build a timeline-preserving semantic representation of resources. This method consists in chunking the raw text into a selected number ($chunk\_size$) of words sized chunks. Each chunk has an overlap of $overlap$ words with the previous chunk. We then train a global Doc2Vec model on all chunks as if they were distinct documents. Each resource r^i is finally represented by a matrix $R^i \in d_e \times d_{r^i}$, where d_e is the embedding size chosen for the semantic Doc2Vec representation (ie. Doc2Vec embedding) and d_r the number of chunks of the resource r^i. This matrix is obtained by concatenation of the semantic representations of each chunk. For this reason, for a given resource, the t-th column of the matrix is the semantic representation of the t-th chunk of the resource. In the sequel we denote R_t^i as the representation of t-th chunk of the resource r^i i.e the t-th column of R^i.



The whole construction of these timeline-preserving semantic representation of resources are summarized in Fig. 1.

The proposed neural architecture described in Fig. IV-B is composed of a Recurrent Neural Network (RNN) followed by a feed-forward Neural Network (FNN). The RNN's role is to project the aforementioned timeline-preserving resource representation into a dense vector capturing the information needed to predict the resource positioning; this vector is expected to completely handle both the semantic and the pedagogic aspects of the resources. The FNN takes as input a pair of RNN outputs and is trained to predict the ground truth consumption order of the pair.

First, for each pair the temporal representation of each pair is independently treated by the same RNN. Given the embedding matrix R^i of a given resource r^i, the RNN computes sequences of outputs $(y_0, \ldots, y_t, \ldots, y_{dr-1})$, $(h_0^{RNN}, \ldots, h_t^{RNN}, \ldots, h_{dr-1}^{RNN})$ by iterating the following equations:

$$h_t^{RNN} = sigmoid(W^{hx} R_t^i + W^{hh} h_{t-1}^{RNN}) \quad (4)$$

$$y_t = W^{yh} h_t^{RNN} \quad (5)$$

with W^{hx}, W^{hh}, W^{yh} being the parameters matrix learned during the training. The last output of the $y_{dr^i-1}^i$ is used as pedagogic embeddings of the resource r^i. To simplify, we introduce the pedagogic embeddings matrix P as $P = \bigoplus_i y_{dr^i-1}^i$ with $\oplus$ the concatenation operator. By definition, $P^i \equiv y_{dr^i-1}^i$.

Second, for each pair of resources $(r^i,\ r^j)$, these corresponding pedagogic embeddings $(P^i,\ P^j)$ are concatenated and directly given as input to the FNN network, followed by a soft-max unit, using the same process as the one described in Section IV-A.

Finally, for both the baseline and the TANN , we train the overall neural model using a stochastic gradient procedure to minimise the sum of squared residuals (SSR) between the estimated order $\hat{o}(r, r')$ and the ground truth order observed in the series $o(r, r')$:

$$\arg\min_{\theta_M} \sum_{\forall r, r' \in TRAIN} [o(r, r') - \hat{o}(r, r' | \theta_M)]^2 \quad (6)$$

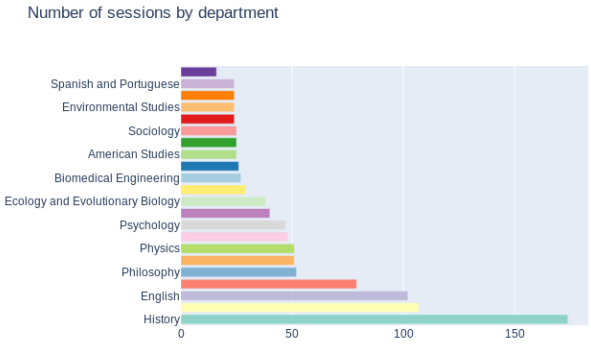


Fig. 2. Distribution of courses by department in YALEOPENCOURSEWARE corpus

V. THE DATA-SET

The YALEOPENCOURSEWARE project [2] provides free and open access to a selection of introductory courses taught in English by distinguished teachers and scholars at Yale University. Each course (we will prefer the term *series*) is composed by a sequence of lectures (we will prefer *episodes*); the handmade transcriptions and chapter division are available for each episode.

The corpus crawled from the project website https://oyc.yale.edu/ contains 40 series from 36 different teachers for a total of 1058 episodes with an average of 26.45 ± 4.8 episodes/series.

The distribution of courses over the different departments is represented in Fig. 2.

VI. EXPERIMENTAL SETUPS

[3] Let us note $\mathcal{X}$ the set of all resources, where the partial order is given by the series of resources built by teachers. Thereby, any two episodes from a series will be comparable, and thus have a ground truth consumption order. If they are from different series they are incomparable.

Next, we build a set $P(\mathcal{X})$ of all the valid pairs of resources. By definition, each comparable pair in this set admits a ground truth consumption ordering in $\mathcal{X}$ (ie. the series of lectures made by teachers).

$$P(\mathcal{X})) = \{(x, y) \in \mathcal{X}^2 \ | x \text{ and } y \text{ are comparable}\}$$

A. Predicting with contextual information

For the two-first tasks (episode level and pair level task), we split our valid pairs $(P(\mathcal{X}))$ as follows. We randomly draw 80% of the episodes for the train set and 10% for each of the validation and of the test set; we denote these sets

[2]Full data-set and additional informations can be found at https://gitlab.univ-nantes.fr/connes-v/yaleocw-corpus

[3]Code and all informations needed for reproduce our experiments can be found at https://gitlab.univ-nantes.fr/connes-v/order_inference

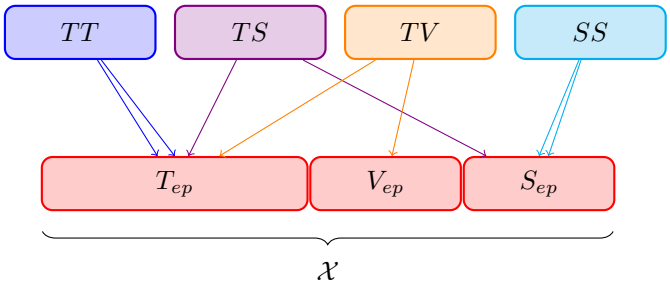


Fig. 3. Graphical representation of intra-series split.

respectively T_{ep}, V_{ep}, S_{ep}. We thus build four disjoint sets TT, SS, TV, TS from pair composed of comparable resources of respectively T_{ep}, S_{ep}, (T_{ep}, V_{ep}) and (T_{ep}, S_{ep}) as illustrated in Fig. 3. Note that all TT, SS, TV, TS are subsets of $P(\mathcal{X})$.

TT is used for the training the model, TS for the episode level task (task a), and SS for the pair level task (task b). We choose to use TV as validation for early-stopping training at the best time.

B. Agnostic case

In this last task (series level), we know nothing about the series from which the tested pair has been taken. For this, we split our dataset as follows. We randomly draw 80% of the series for the train set and 10% for each the validation and the test set. We denote respectively, the set of pairs from the train, the test and the valid set respectively T, V, S.

In order to take into account courses taught by a same teacher, the split ensures that course of a same teacher are always in different subsets.

As with other neural architectures, our model and the baseline inherit several meta-parameters which must be set in order to achieve the best possible learning. In order to fix these, we use cross-validation combined with a grid search approach on the agnostic setup and kept the best set of meta-parameters. The size of chunks and of the overlapping for timeline-preserving representation were also experimentally set and kept coherent for capturing major semantic change in the discourse. We chose $chunk\_size = 1000\ words$ and $overlap = 500\ words$.[4]

In Section VII, the results are presented on the baseline defined in Section IV-A, our TANN defined in Section IV-B. A possible extension to the TANN is to incorporate an attention mechanism. The obtained model is denoted as TANN +Att (a reader interested by this method for adding an attentive mechanism on an RNN architecture may refer to [18]).

VII. Results

For each task, the accuracy of a model M (with parameter θ_M) on a specific subset s is computed as follows:

$$Accuracy(M, s) = \frac{\sum_{\forall (r,r') \in s} \lfloor \hat{o}(r, r' | \theta_M) \rceil = o(r, r')}{|s|}, \quad (7)$$

[4]1000 words approximately correspond to 8-10 min of talk for an average slides presentation.

where $\hat{o}(r, r'|\theta_M) \in [0, 1]$ is an estimate of the binary ground truth order $o(r, r')$ predicted for the input (r, r') by the model M.

Table I summarizes the results obtained on the different tasks: the TANN outperforms the baseline for predicting on pairs from a completely new series of new teacher and often from a completely new domain obtaining an accuracy of 69%, which is arguably the most difficult task.

In the two first tasks, our model has seen contextual information about the series during the training, when the model is used to predict the consumption order of a pair partially seen in the train set (TS), it obtains an average of 80% good predictions and again outperforms the baseline.

We also notice an important gap of performance between the folds of cross validation and we empirically observe that the best test results are also the best validation results: the split operated is possibly the major factor for explaining this variation. As suggested in the literature of recommender systems [2], this makes it advantageous to consider keeping only the best models on the validation for a final exploitation of the model. Finally, we observe, as discussed in the beginning of Section VI that the classic extension of the RNN, here represented by the attention mechanism (TANN + Att) obtains a similar performance to the TANN whilst augmenting the number of parameters. For this reason, this extension appears to be less relevant in the context of our problem.

Task:	With contextual information						Agnostic		
Model:	TS			SS			S		
	Avg	Max	Min	Avg	Max	Min	Avg	Max	Min
Baseline	0.76	0.82	0.69	**0.74**	0.79	0.66	0.63	0.70	0.54
TANN	**0.80**	0.84	0.76	0.71	0.80	0.65	**0.69**	0.77	0.61
TANN + Att	0.78	0.81	0.67	0.72	0.80	0.62	0.67	0.71	0.63

TABLE I
10-FOLDS CROSS VALIDATION ACCURACY FOR EPISODES, PAIR AND SERIES LEVEL TASKS

VIII. Discussion

In this work we choose to build a generic model able to predict regardless of the domain instead of building a specific model by domain. This choice is motivated by two main reasons. The first is that in the case of the global OER network the categorisation of resources into domains is made complex by the nature of the construction of this network. Indeed, the diversity of origins, domains, formats and languages does not usually and systematically allow us to obtain a labelling of resources by domain through meta-data. Of course, there are many categorization algorithms that could compensate for this deficiency. However, the evaluation of their annotation in a context such as the one of the global OER network is a research question in itself, which is beyond the scope of this paper. The second reason is much more pragmatic. From our experimentation on the YaleOpenCourseware corpus we have observed that the generic model performs equivalently to the specific models or even much better in the case of poorly endowed domains. For this reason we believe that the use of a

generic model allows a transfer of learning between domains. Indeed, we think that some semantic information allowing to schedule the resources is independent of the domain. However, we have chosen not to present these experiments in this paper as the lack of resources for specific domains does not allow us to properly evaluate this transfer of learning. In-depth experiences to confirm this result is an interesting research perspective.

Furthermore, the general setting is that of a cold-start: there are not enough user data elements to (1) build a solution based on user activity, and (2) use user activity to perform AB testing.

The ultimate goal of pedagogic recommendation would be to build a learning path for a given user in a variety of situations: her goals may be clear or not, the material she wishes to include in her learning experience may be completely or partially unknown. This learning path should therefore take into account many facets of the learning problem. Being able to rank the resources will certainly not, on its own, give the answer to this more ambitious goal, but we believe it is a necessary condition for success in this task. In project X5-GON we have been addressing the questions of difficulty, randomness, personalization. And the ideas presented in this work have already led to an implementation on platform x5learn.org: a user is encouraged to browse the collections of OER and build a playlist, and then to ask for the elements of this playlist to be reordered in a more comprehensive way.

IX. Conclusion

To conclude, in this work we proposed a new framework to tackle the problem of pedagogic recommendation by focusing on the necessity of satisfactory continuity in the learning path. For this, we designed a new learning task as proxy for this problem which tries to take benefit of the implicit pedagogic continuity embedded in teacher built series of lectures. We then provided a new open free data-set for pedagogic recommendation and evaluated the new timeline-preserving neural network based approach we specifically designed for this task against a state of the art NLP baseline. The evaluation demonstrates that our model outperforms the baseline and this particularly when we request it to predict on completely new pairs of resources, even out of any context seen during learning. In this agnostic setting, 69% of accuracy was obtained.

This result argues for the capacity of the model to predict in real life settings; We expect in the future to replace the current tools allowing to rank OER in the X5-GON platforms by models built following this work, and evaluate it with AB-testing and user experience analysis.

References

[1] P. J. Guo and K. Reinecke, "Demographic differences in how students navigate through MOOCs," in *Proceedings of the first ACM conference on Learning@ scale conference*. ACM, 2014, pp. 21–30.

[2] Z. Zhao, L. Hong, L. Wei, J. Chen, A. Nath, S. Andrews, A. Kumthekar, M. Sathiamoorthy, X. Yi, and E. Chi, "Recommending what video to watch next: a multitask ranking system," in *Proceedings of the 13th ACM Conference on Recommender Systems*. ACM, 2019, pp. 43–51.

[3] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM conference on recommender systems*. ACM, 2016, pp. 191–198.

[4] W. Krichene, N. Mayoraz, S. Rendle, L. Zhang, X. Yi, L. Hong, E. Chi, and J. Anderson, "Efficient training on very large corpora via gramian estimation," in *International Conference on Learning Representations*, 2019.

[5] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi, "Modeling task relationships in multi-task learning with multi-gate mixture-of-experts," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1930–1939.

[6] J. Allgaier, "Science and Environmental Communication via Online Video: Strategically Distorted Communications on Climate Change and Climate Engineering on YouTube," *Frontiers in Communication*, vol. 4, p. 36, 2019.

[7] B. Rieder, A. Matamoros-Fernández, and Ò. Coromina, "From Ranking Algorithms to 'Ranking Cultures' Investigating the Modulation of Visibility in YouTube Search Results," *Convergence*, vol. 24, no. 1, pp. 50–68, 2018.

[8] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'13. Red Hook, NY, USA: Curran Associates Inc., 2013, p. 3111–3119.

[9] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[10] H. Drachsler, K. Verbert, O. C. Santos, and N. Manouselis, "Panorama of recommender systems to support learning," in *Recommender systems handbook*. Springer, 2015, pp. 421–451.

[11] C.-K. Hsu, G.-J. Hwang, and C.-K. Chang, "A personalized recommendation-based mobile learning approach to improving the reading performance of EFL students," *Computers & Education*, vol. 63, pp. 327–336, 2013.

[12] K. Abhinav, V. Subramanian, A. Dubey, P. Bhat, and A. D. Venkat, "Lecore: A framework for modeling learner's preference." in *11th International Conference on Educational Data Mining*, 2018.

[13] C. Wong, "Sequence based course recommender for personalized curriculum planning," in *International Conference on Artificial Intelligence in Education*. Springer, 2018, pp. 531–534.

[14] A. Hernández-Blanco, B. Herrera-Flores, D. Tomás, and B. Navarro-Colorado, "A systematic review of deep learning approaches to educational data mining," *Complexity*, vol. 2019, 2019.

[15] J. D. V. Miró, P. Baquero-Arnal, J. Civera, C. Turró, and A. Juan, "Multilingual videos for moocs and OER," *J. Educ. Technol. Soc.*, vol. 21, no. 2, pp. 1–12, 2018.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[17] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, p. 3104–3112.

[18] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," Jan. 2015, 3rd International Conference on Learning Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015.