X Modal
X Cultural
X Lingual
X Domain
X Site
Global OER Network

Grant Agreement Number:	761758
Project Acronym:	X5GON
Project title:	Cross Modal, Cross Cultural, Cross Lingual, Cross Domain, and Cross Site Global OER Network
Project Date:	2017-09-01 to 2020-08-31
Project Duration:	36 months
Deliverable Title:	D1.5 – Evaluation Methodologies for Content Representation Models and Release of Datasets for Measuring Quality of OERs
Lead beneficiary:	UCL
Type:	Report
Dissemination level:	Public
Due Date (in months):	36 (August 2020)
Date:	31-December-2020
Status (Draft/Final):	Final
Authors:	Sahan Bulathwela, Maria Perez-Ortiz, E. S. V. Ranawaka, R. I. P. B. B. Siriwardana, G. A. K. Y. Ganepola, Emine Yilmaz and John Shawe-Taylor
Contact persons:	Sahan Bulathwela, Maria Perez-Ortiz, Emine Yilmaz and John Shawe-Taylor

Revision

Date	Lead author(s)	Commments
01/12/2020	Sahan Bulathwela Maria Perez-Ortiz, Emine Yilmaz and John Shawe-Taylor	Initial Draft
08/12/2020	E. S. V. Ranawaka R. I. P. B. B. Siriwardana and G. A. K. Y. Ganepola	Added Chapters on Language Detection and Duplicate Detection
14/12/2020	Alfons Juan	Internal Review
28/12/2020	Sahan Bulathwela	Final Version

Contents

List of Figures

List of Tables

Abstract

Multiple content representation models have been proposed in deliverable D1.4 – Advanced Content Representations [1] towards improving automatic language detection, duplicate detection and improved personalisation of educational materials, all tasks which are aimed towards improving the quality of the X5GON Open Educational Resource (OER) database and its services. Most suitable evaluation methodologies and metrics should be identified in order to carry out effective model comparison. This report outlines the relevant evaluation methodologies identified and enforced towards selecting the most suitable content representation model for language detection, duplicate detection and personalisation. Multiple novel datasets that were constructed for evaluating the content representation models are also described. Furthermore, VLEngagement, a novel dataset that consits of over 4,000 video lectures were constructed and published to advance the research front of automatic, scalable quality assurance of OERs.

[1] https://www.x5gon.org/science/deliverables/

1 Introduction

As the Internet gets filled with more and more powerful educational resources by the day, it is greatly imperative that the scientific community takes actions to improve how this plethora of educational resources can be matched to the relevant learner without hindering their learning experience. This boom in availability of educational resources is further fuelled by the OER movement that aims to democratise high quality educational resources to all. In order to achieve this goal, platforms such as X5GON [1] and X5Learn[2] has stood to the role in unifying openly available educational resources that are currently scattered all over the world. There are major challenges in achieving such tasks relating to the sheer scale of materials that need to be processed [3].

In order to address some of the key pain points in aggregating all of OERs to one index, X5GON project has proposed several content representation models that allows tasks such as automatic language detection, duplication detection and personalisation of educational materials. Deliverable *D1.4 – Advanced Content Representations* report outlines these models. This report describes the evaluation methodologies that are formulated for assessing the usefulness of multiple models that have been developed to improve the content representations of OERs outlined in the aforementioned deliverable.

Furthermore, it reports details on multiple datasets that have been constructed and published in order to improve the quality of representations that can be maintained in a large educational resource database.

1.1 Chapter Overview

This report mainly revolves around two main themes.

1. Evaluation Methodologies: Topics relating to the evaluation methodologies for advanced content representation models built by X5GON.

2. Published Datasets: Proposal of multiple datasets that have been constructed and made available publicly for improving quality of OERs.

In chapter 2 of the report, we discuss the nature of the task of recommending personalised learning materials to users. Then we propose a set of evaluation methodologies that can be used to evaluate the content representation models objectively.

In chapter 3, we identify two quality related tasks (automatic language detection and duplication detection) that are elemental in improving the user experience and quality of the collection of OERs that are maintained in the X5GON database. The evaluation methodologies for assessing the language and duplicate detection models are proposed along with two datasets that contribute towards quantitatively assessing the performance.

Chapter 4 of the report proposes a novel supervised dataset for predicting context-free engagement of scientific video lectures. Engagement is a main component of quality of an educational material. This dataset will enable the research community to push the research landscape to new horizons.

Finally, chapter 5 summarises the details of the report and concludes it.

2 Evaluation Methodologies for Advanced Content Representations for Educational Recommendation

The advancement of identifying suitable content representations lies in the heart of an intelligent learning platform. In the context of building educational recommendation systems, the topic lies at the intersection of the research topics, intelligent tutoring systems (ITS) and information retrieval/ recommendations systems (IR).

2.1 Potential Evaluation Criteria

Depending on how the task is formulated, different evaluation methodologies may be more suitable for assessing the content representation models that are developed. Different tasks formulated in ITS and IR communities use different families of evaluation metrics to evaluate performance of content representation models.

2.1.1 Classification

Knowledge Tracing (KT) [4] is one of the foundational models that are used by the ITS community to predict learner future activity. In this model, a representation of the learner's knowledge is inferred and is used to predict if a learner is likely to succeed answering a question or otherwise. Due to the fact that the outcome of the action is binary (+1 if the learner succeeds, -1 otherwise), classification metrics are suitable to assess the performance of such a model. Different classification metrics such as accuracy [5, 6, 7], Area-Under-the-Curve (AUC) [8, 9, 10], Precision, Recall and F1 score [7, 11].

Applications that go beyond KT such as recommendation, matchmaking which are a highly active research areas among the IR researchers use classification metrics such as classification accuracy [12].

2.1.2 Quantifying Real Error

It is also evident that previous works also tend to use metrics that quantify error in a continuous scale. Works in KT also use Root Mean Square Error (RMSE) to quantify error [5, 10, 13]. Although more intuitive metrics such as Mean Absolute Error have been used [14, 15], the impact of its improper nature makes it an undesirable metric [16].

2.1.3 Ranking

Ranking metrics are another family of metrics that are used in similar tasks. Especially, when multiple items are ranked in-front of a learner, ranking metrics are useful. Metrics such as Normalised Cumulative Discounted Gain (NDCG) , Average Reciprocal Hit Rank (ARHR) are used to evaluate recommendation systems and information retrieval applications [17].

2.2 Nature of the Current Task

In this work, we set out to evaluate an advanced content representation model that can predict the likelihood of a learner engaging with a learning resource based on her historical interactions with other learning resources. Engagement in this context is a binary outcome where positive outcomes occur when the engagement criteria is met. The task is similar to the typical task tackled in KT models where the success of the learner to answer a question correctly is predicted.

2.3 Dataset

We use data from a popular OER repository to evaluate the performance of the models. The data source consists of users watching video lectures from VideoLectures.Net[2]. The lectures are also accompanied with transcriptions and multiple translations provided by the TransLectures project[3]. For detailed descriptions of the construction of transcription and translation models, we direct the reader to Deliverables *D3.4 – Early support for cross-lingual OER* [18] and *D3.5 – Final support for cross-lingual OER* [19]. We use the English transcription of the lecture (or the English translation where the resource is non-English) to annotate the lecture with relevant KCs using Wikifier. We divide the lecture text into multiple fragments of approximately 5,000 characters (equivalent roughly to 5 minutes of lecture) [2]. This data is sourced from the same data source that was used to create VLEngagement dataset described in chapter 4.

The choice of video partitioning is motivated by several reasons. The first one is a technical limitation on the number of characters supported by Wikifier[20]. However, we also believe that these partitions allow us to use finer-grain engagement user signals, where our algorithm learns from the specific partitions that the user watched (and the topics covered in those).

Once the fragments are Wikified, we rank the topics using a linear combination of PageRank and cosine similarity (further details in the next section) and use the top k ranked topics along with the associated cosine similarity as our feature set. We define binary engagement e^t_{ℓ,r_i} between a learner ℓ and a resource r_i as 1 if the learner watched at least 75% of the fragment of 5000 characters, and -1 otherwise. This is because we hypothesise that the learner must have consumed approximately the whole fragment to learn significantly from it. Note that user view logs are of learners actively accessing videos, i.e. when engagement is negative the learner has accessed the material but left without spending a significant amount of time on it.

In terms of individual sessions, the dataset includes a collection of view log events. For each user, the videos that they watched and the exact parts that they watched are recorded in the dataset. The timestamps in the data allows sequencing the watch data in the correct order. The source dataset consisted of 25,697 lectures as of February 2018 that were categorised into 21 subjects, e.g. Data Science, Computer Science, Arts, Physics, etc. However, as VideoLectures.net has a heavy presence of Computer Science and Data Science lectures, we restricted the dataset to lectures categorised under Computer Science or Data Science categories only. To create the dataset, we extracted the transcripts of the videos and their viewers' view logs. A total of 402,350 view log entries were found between December 8, 2016 and February 17, 2018. These video lectures are long videos that run for 36 minutes on average and hence discuss a large number of KCs in a single lecture.

2.4 Experimental Design

A learner model is built for every learner independently based on the user interaction data. Given that we aim to build this algorithm for online system, we test the different learner models using a sequential experimental design, where engagement of fragment t is predicted using fragments 1 to $t-1$.

2.4.1 Hyper-parameter Tuning

The learner model we propose contains several hyper-parameters that we train using grid search. We use a hold-out validation approach for hyper-parameter tuning where hyper-parameters are learned on 70% of the learners (validation set) and the model is evaluated (tested) on the remaining 30%

[2] `www.videolectures.net`

[3] `www.translectures.eu`

using the the best hyper-parameter combination from the validation set. Note that we both learn and predict the engagement per fragment.

Regarding initial configurations and hyper-parameters, we initialised the initial mean skill of learners to 0 for all reformulations of TrueLearn. We use grid search to find the suitable hyper-parameters for the initial variance while keeping β constant at 0.5. The search range for the initial variance was [0.1, 2]. For these models, initial hyper parameters are set in the following manner. First, we compute σ_c^2, the variance of the cosine similarity values belonging the educational resources. Initial variance of the learner (σ_ℓ^2) is set as $\sigma_\ell^2 = (\sigma_c^2 \times initial\ variance\ factor)^2$. Then we set initial β^2 as $\beta^2 = (\sigma_\ell * \beta\ factor)^2$

We also tested different combinations of τ $(0.1, 0.05, 0.01)$, the hyper-parameter controlling the dynamic factor [12]. However, the results did not changed for different settings. This suggests that the dataset might still be relatively small and sparse for this factor to have an impact. The algorithms were developed in python, using MapReduce to parallelise the computation per learner. The code for TrueLearn and all the baselines is available online[4].

2.4.2 Model Comparisons

We carry out a two step model comparison process to identify the most suitable model.

KC Ranking Weights with TrueLearn Novel Model (Step 1) Section 2.3 states that the KCs identified for learning resources are ranked using a weighted combination of PageRank and Cosine Similarity scores obtained from Wikifier [20]. However, *Deliverable D1.3 – Initial Content Representations* [21] uses the weight combination that is best performant on the Knowledge Tracing model that was proposed as a baseline. Given that TrueLearn Novel model outperforms Knowledge Tracing model [7], we find the optimal weight combinations for ranking KCs that will perform best with TrueLearn Novel Model. This is done by using combinations $W_{PageRank} = \alpha$ and $W_{Cosine} = 1 - \alpha$ where $\alpha \in \{.0, .2, .4, .6, .8, 1.\}$.

Hyperparameters for Semantic TrueLearn Novel (Step 2) Once the ranking weight combination is identified, we carry out hyper-parameter tuning for Semantic TrueLearn Novel model. We utilise multiple semantic relatedness metrics [22] to train Semantic TrueLearn model and compare those models with TrueLearn Novel model which we treat as the baseline. For a detailed description of the architecture of Semantic TrueLearn model and for an exhaustive list of semantic relatedness metrics used for the experiments, we direct the reader to Deliverable *D1.4 – Advanced Content Representations*.

2.5 Evaluation Metrics

Since engagement is binary, predictions for each fragment can be assembled into a confusion matrix, from which we compute well-known binary classification metrics. For the engagement prediction task, we identify accuracy, precision, recall and F1-score as suitable metrics to look at. A detailed description on how to compute these metrics can be found in Appendix A.1.

As we have multiple learners in the dataset, we compute the weighted average of classification metrics to represent the overall performance of the model with the population of learners in the dataset. We average these metrics per learner and weight each learner according to their amount of activity in the system. The weighted average of each classification metric m_Ω is computed according to equation 1

[4] `https://github.com/sahanbull/semantic_truelearn`

$$m_\Omega = \sum_{\ell \in \Omega} \frac{|\ell|}{\sum_{\ell \in \Omega} |\ell|} \cdot m_\ell, \tag{1}$$

where Ω is the set of all users in the test dataset, $|\ell|$ is the number of events coming from user ℓ and m_ℓ being the classification metric score for user ℓ where m can be accuracy, precision, recall or F1 score.

Note that most learners present an imbalanced setting, where they are mostly engaged or disengaged. Because of this, we do not use Accuracy as the main metric, but rather focus on Precision, Recall and F1. In the context of predicting if a learner is going to engage with an educational resource, both precision and recall are important metrics to consider. Therefore we use *F1 score*, the harmonic mean of precision and recall as the ultimate metric that is used for model comparison.

2.6 Summary of Results

Once the models were trained and compared according to the 2 step protocol described in section 2.4, The most suitable model is identified.

It was determined from step 1 ranking weights tuning that the optimal weight combination for ranking KCs is $W_{PageRank} = 0.8$ and $W_{Cos} = 0.2$. Step 2 model comparison between baseline TrueLearn algorithm and Semantic TrueLearn implementations shows that Semantic TrueLearn can improve predictive performance in terms of recall and F1 score across multiple semantic relatedness metrics. The most performant Semantic TrueLearn Novel model utilises Word2Vec based semantic relatedness and outperforms baseline TrueLearn Novel algorithm in precision and F1 Score. For a detailed report of full evaluation results and discussion of the chosen model, we direct the reader to Deliverable *D1.6 – Report on Selected Models and Content Representations*.

3 Evaluation Methodologies and Datasets for Quality Related Tasks

This chapter details the evaluation methodologies and datasets constructed and published in order to improve the quality of the materials stored in X5GON database. There are two main components that were identified and developed.

1. **Language Detection:** Detecting the language of materials beforehand is fundamental to AI-powered enrichment tasks that are carried out by X5GON processing pipeline (such as Wikification [20], translation services etc.).
2. **Duplicate Detection:** Detecting duplicates of the same educational resource is useful in compiling information retrieval results (search and recommendation results) that are pleasing to the learners in terms of user experience.

3.1 Language Detection

X5GON database consists of OERs that come from many different European Languages [1] and continues to discover new materials [23]. Due to the scale of documents coming in, a language detector that is accurate and fast is suitable.

3.1.1 Source Dataset

We utilise WiLI-2018, a popular dataset for monolingual written natural language identification [24] to benchmark the candidate models. WiLI-2018 is a publicly available and available free of charge with short text extracts from Wikipedia. It contains 1,000 paragraphs of 235 languages, totalling in 235000 paragraphs. This dataset is a classification dataset where the train-test splits are well defined.

We utilise a subset of this dataset as X5GON database is composed with materials that are authored in a subset of European Languages. We only consider *eight* European languages, namely, German(`de`), Dutch(`nl`), English(`en`), Slovene(`sl`), Slovak(`sk`), French(`fr`), Italian(`it`) and Spanish(`es`) as the majority of X5GON database consists of these languages [1]. From the selected data, we construct two datasets.

1. *Monolingual Dataset* that only consists of observations that have one language in them
2. *Bilingual Dataset* where each observation consists of a pair of aforementioned languages.

All the created datasets are available publicly [5].

Monolingual Dataset The Monolingual Dataset samples the subset of languages from the WiLI dataset. The dataset can be used as it is to evaluate performance of language detection models in the scenario where documents are monolingual.

Bilingual Dataset Constructing a bilingual Document dataset is important as one document may contain more than one language. There are many examples for such instances such as web pages with extracts from other languages and European Union documents [25] that motivates understanding multi-language detection.

[5] `https://github.com/X5GON/X5_langdetect`

Proportion of Language Presence To evaluate how the language detection models behave with changing proportions of languages is also important. In order to evaluate this, we synthesise observations where we control the proportion of text that is included in the synthesised bi-lingual observations. For each language pair, text extracts from WiLI dataset is randomly selected until the desired proportion of the two languages is obtained. The proportions selected were 20:80, 30:70, 40:60 and 50:50 case where two equal sized documents were concatenated together.

3.1.2 Evaluation Metrics

Automatic Language Detection task is a multi-label classification task [24] (multi class in monolingual case [26])that can be evaluated using classification metrics. We utilise classification accuracy to measure the performance of the the language detection models. As the datasets are free of effects such as label imbalance, classification accuracy is a sensible metric to represent model performance. The exact definition of accuracy score can be found in appendix A.1.

3.1.3 Computational Cost

The language detection service is built with the aim of running it in a production setting. The ability to scale with data velocity is essential to make sure that it can cope with a collection of documents that gets ingested to X5GON. Execution time is the most realistic metric for measuring time cost. The average time taken per observation for iteratively (thrice) classifying the dataset is calculated as per equation 2.

$$Time(\gamma) = \frac{1}{3 \cdot |\Omega|} \sum_{i \in \Omega} \sum_{j=1}^{3} t(\gamma, i) \tag{2}$$

where Ω is the set of observations in the dataset, $|\cdot|$ returns the cardinality of dataset and $t(\gamma, i)$ is a function that returns the time taken to classify observation i using language detection model γ.

3.1.4 Experimental Methodology

Two experiments are setup with the two datasets synthesised in section 3.1.1. The classification accuracy and the computational cost is recorded for both mono-lingual and bilingual datasets to identify the suitable model. There is no necessity of a train/test split as there is no language model training that takes place. Experimenting with the monolingual dataset is straightforward as every observation has only one label. Therefore, classification accuracy can be calculated without difficulties.

Contrary to the monolingual dataset, the bilingual dataset is not straightforward. We calculate classification accuracy subject to three distinct phenomena.

1. *Pair-wise Accuracy*: Observation considered to be predicted accurately if,
   (a) Both languages (full pair) are predicted correctly
   (b) The order of dominance is predicted correctly
2. *Dominant Language*: Observation considered to be predicted accurately if the dominant language in the observation is predicted correctly. The second dominant language may be mis-classified or not predicted at all.
3. *Both Languages*: Observations considered to be predicted accurately if the two languages are predicted correctly. The order may be mis-classified.

3.1.5 Summary of Results

The accuracy of detecting the dominant language in each library seems to decrease as the split percentages reaches 50-50 while the pairwise accuracy and the accuracy in detecting both languages increases. As the original dataset of the problem may include documents written both a single language and multiple languages, using a single library for language detection may result in loss of accuracy.

A two step ensemble strategy is proposed where the dominant language is predicted in the first step with higher accuracy and then determining if the document other multiple languages. The proposed model has an overall accuracy of 95%. Due to the computational efficiency of the models, running two models sequentially is technically feasible. For an extensive study of the results, we direct the reader to deliverable *D1.6 – Report on Selected Models and Content Representations.*

3.2 Automatic Duplicate Detection

The derivation of duplicate detection model entails an exploratory analysis based approach where different content representation techniques are evaluated for their ability to identify duplicate documents.

3.2.1 Content Representation Models

Two content representations are used as candidates for duplication detection system.

1. **Term-Frequency Feature Set (TF)**: In this representation, the words of the document are represented using the Bag-of-Words representation where the frequency of terms is used as the numeric representation for each token. Bag-of-Words representation is widely used in information retrieval domain (eg: Vector Space Model [27], Measuring Document Similarity [28])

2. **Wikification Representation**: The document is represented as a Bag-of-Wikipedia Concepts. The Wikipedia can be extracted via entity linking [20]. The cosine similarity between the Wikipedia Topic and the textual content of the document is treated as a proxy for topic coverage [7]. Wikification associates Wikipedia concepts to documents which is similar to the process of document tagging. Topic similarity is a higher level approach that can be used to detect document similarity that is not sensitive to slight deviation of tokens in the documents. Wikification representations allows us to do this.

We devise cosine similarity according to equation 3 to compute pairwise similarity between documents in both content representations.

$$cos(d_1, d_2) = \frac{\phi_{d_1} \cdot \phi_{d_2}}{\|\phi_{d_1}\| \times \|\phi_{d_2}\|} \tag{3}$$

where d is an OER in X5GON and ϕ_d is vector representation of document d when transformed to content representation ϕ and $\|\cdot\|$ is the norm of the vector.

3.2.2 Data Source

The main data source for this work is the X5GON database and the educational resources that have been ingested in it. The document processing pipeline extracts the text representation of the OERs which is used for creating content representations described in section 3.2.1. For more information about how and what data is extracted from OERs while content processing, we direct the reader to deliverable *D2.2 – Final Server-side Platform*[29]. We randomly sample 10,000 documents for ease of data processing from this database which has over 100,000 documents.

Final Datasets For every different non-deterministic content representation model, we use the model for detecting document similarity. Then we randomly sample a subset of document pairs that the model classifies to be duplicate materials. Then we use human annotators to obtain gold standard labels for these materials. The human annotators use predefined definitions to annotate documents pairs as duplicates or otherwise. We direct the reader to deliverable *D1.4 – Advanced Content Representations* to find additional information about the two different definitions of duplicate materials. The human annotated duplication detection dataset is publicly available [6]

3.2.3 Methodology

We attempt to identify duplicate documents using multiple approaches while attempting to answer multiple questions through this work.

- What are the deterministic ways to identify duplicates?
- Is there a connection between document word counts and similarity score?
- How to incorporate TF and Wikification representations to identify similar documents?
- Does the best model identify duplicates effectively?

At first, we evaluate the feasibility of using deterministic techniques such as URL redirects, document hash, filenames to resolve duplicate documents.

Then, we introduce approximate approaches that depend on alternative content representations (outlined in section 3.2.1) to evaluate the feasibility of detecting near duplicate documents. The two content representations utilised has different strengths where the TF representation is good at capturing token level similarities while the Wikification representations are better at capturing topic level similarities. We run a series of experiments to investigate which approach is more suitable.

3.2.4 Evaluating Duplicate Document Detector

As described in section 3.2.2, we use model predictions to sample observations that are then labelled by human annotators. We use the human labelling to verify the fraction of correctly labelled duplicates. The fraction value we obtain is the definition of *precision* as per appendix A.1. We use precision for model comparison.

3.2.5 Summary of Results

From experimenting with the deterministic approaches for finding similar documents, we managed to identify 5,312 documents that are duplicates of other OERs in X5GON. We also identified two repository domains where URL redirects lead to indexing duplicate materials.

When evaluating non-deterministic similarity based models, the results indicated that both content representations have strengths in detecting duplicate documents. Therefore, a content representation model that uses both term frequency and Wikification representation based similarity was identified as the best model.

A detailed description of the model evaluation results can be found in deliverable *D1.6 – Report on Selected Models and Content Representations.*

[6] `https://github.com/X5GON/dupe_detect`

4 VLEngagement: A Dataset of Scientific Video Lectures for Evaluating Population-based Engagement

The VLEngagement dataset is constructed using the aggregated video lectures consumption data coming from a popular scientific OER repository, VideoLectures.Net[7]. These videos are recorded when researchers are presenting their work at peer-reviewed conferences. Lectures are thus reviewed and material is controlled for correctness of knowledge. It is noteworthy that the dataset consists of *scientific video lectures* that explain novel scientific work geared more towards postgraduate, PhD level learners and the scientific research community. Therefore, the learner audience of the video lectures in this dataset may significantly differ from one of a conventional MOOC platform.

The dataset provides a set of statistics aimed at studying population based engagement in video lectures, together with other conventional metrics in subjective assessment such as average star ratings and number of views. We believe the dataset will serve the community applying AI in Education to further understand what are the features of educational material that makes it engaging for learners.

4.1 Feature Extraction

The dataset provides three types of features as outlined in Table 2: i) content-based textual features, ii) Wikipedia entity linking features and iii) video-based features. Although our dataset is composed of video lectures data, the majority of our features (with exception of some of the features in the video-based category) can be used across different modalities of educational material (e.g. books) as they are computed only considering the text transcription. The transcriptions for the English lectures and the English translations of the non-English lectures are provided by the TransLectures project[8].

In this section, we define how different features are calculated from the lecture transcription. These features have been identified from the related work and are categorised under different verticals of quality assurance in text articles [30, 31, 32, 33] and engagement with video lectures [34]. The verticals are for example understandability, topic coverage, presentation, freshness and authority [35]. The code for computing some of these features is available together with the dataset.

4.1.1 Content-based Features

For explaining the features based on content transcripts, several functions need to be introduced: i) $\texttt{count}(s)$ is a function that returns the number of tokens in string s, ii) $\texttt{count}(t, s)$ is a function that returns the number of occurrences of tokens in token set t in string s and iii) $\texttt{u\_count}(t, s)$ returns the frequency of unique tokens from token set t in string s. String s can be the transcript text s_{tr} or the lecture title s_{title}. *Stop-word Presence Rate* and *Stop-word Coverage Rate* are calculated using Eq. 8 and 9 based on the work of Ntoulas et al [32]. Textual features defined by Eq. 10 through Eq. 15 are based on the work of Dalip et al [36]. All definitions utilised the token sets provided in Table A.3. More specifically, the content-based features extracted are the following:

- *Word Count* of lecture transcript s_{tr}:

$$\texttt{Word Count} = \texttt{count}(s_{tr}) \tag{4}$$

- *Title Word Count* of lecture s_{title}:

$$\texttt{Title Word Count} = \texttt{count}(s_{title}) \tag{5}$$

[7] www.videolectures.net
[8] www.translectures.eu

- *Document Entropy*, based on the work of Bendersky2011, is calculated over every word w in transcript s_{tr} as:

$$\texttt{Document Entropy} = \sum_{w \in s_{tr}} p_{s_{tr}}(w) \log p_{s_{tr}}(w), \tag{6}$$

  where $p_{s_{tr}}(w_i) = \frac{\texttt{count}(w_i, s_{tr})}{\texttt{Word Count}}$.

- *FK Easiness* is computed using `textatistic` [37] for transcript s_{tr} using:

$$\texttt{FK Easiness} = 206.835 - 1.015 \left(\frac{\texttt{Word Count}}{\texttt{sen\_count}(s_{tr})} \right) - 84.6 \left(\frac{\texttt{syll\_count}(s_{tr})}{\texttt{Word Count}} \right) \tag{7}$$

  where $\texttt{sen\_count}(s_{tr})$ and $\texttt{syll\_count}(s_{tr})$ returns the number of sentences and syllables in transcript s_{tr} respectively. FK Easiness proxies complexity of the language used giving a low score for complex language and vice versa.

- *Stop-word Presence Rate* of lecture transcript s_{tr}:

$$\texttt{Stop-word Presence Rate} = \frac{\texttt{count}(sw, s_{tr})}{\texttt{Word Count}} \tag{8}$$

- *Stop-word Coverage Rate* of lecture transcript s_{tr}:

$$\texttt{Stop-word Coverage Rate} = \frac{\texttt{u\_count}(sw, s_{tr})}{\texttt{count}(sw)} \tag{9}$$

- *Preposition Rate* of the lecture transcript s_{tr}:

$$\texttt{Preposition Rate} = \frac{\texttt{count}(prep, s_{tr})}{\texttt{Word Count}} \tag{10}$$

- *Auxiliary Rate* of the lecture transcript s_{tr}:

$$\texttt{Preposition Rate} = \frac{\texttt{count}(auxi, s_{tr})}{\texttt{Word Count}} \tag{11}$$

- *To Be Rate* of lecture transcript s_{tr}:

$$\texttt{To Be Rate} = \frac{\texttt{count}(tobe, s_{tr})}{\texttt{Word Count}} \tag{12}$$

- *Conjunction Rate* of lecture transcript s_{tr}:

$$\texttt{Conjunction Rate} = \frac{\texttt{count}(conj, s_{tr})}{\texttt{Word Count}} \tag{13}$$

- *Normalisation Rate* of lecture transcript s_{tr}:

$$\texttt{Normalisation Rate} = \frac{\texttt{count}(norm, s_{tr})}{\texttt{Word Count}} \tag{14}$$

- *Pronoun Rate* of lecture transcript s_{tr}:

$$\texttt{Pronoun Rate} = \frac{\texttt{count}(pron, s_{tr})}{\texttt{Word Count}} \tag{15}$$

- *Published Date* of video lecture ℓ calculates the epoch time of publication date of the lecture in days [38]:

$$\texttt{Published Date} = \texttt{days}(\ell_{pub\_date} - 1970/01/01) \tag{16}$$

Various prior works provide the rationale behind the suitability of these features [35, 34, 36].

4.1.2 Wikipedia-based Features

The Wikipedia topics most connected to the lectures are identified using Wikification [20], an entity linking approach. Using the identified Wiki topics, four different feature groups are introduced with the dataset. They fall under the *Authority* and *Topic Coverage* verticals.

The *top-5 authoritative topic URLs* and *top-5 PageRank scores* features represent the Topic Authority feature vertical. Figure 1 (left) shows the summary of Wikipedia topics that are most authoritative (top 1 topic) in the lectures found in the dataset. When PageRank score [39] is computed, Wikipedia topics heavily connected to other topics (i.e. more semantically related) within the lecture will emerge. Hence, the top-ranking topics are the more authoritative topics within the context of topics in the lecture. During Wikification [20], a semantic graph is constructed where semantic relatedness ($SR(c, c')$) between each Wikipedia topic pair c and c' in the graph are calculated using:

$$SR(c, c') = \frac{\log(max(|L_c|, |L_{c'}|) - \log(|L_c \cap L_{c'}|)}{\log |W| - log(min(|L_c|, |L_{c'}|)} \tag{17}$$

where L_c represents the set of topics with inwards links to Wikipedia topic c, $|\cdot|$ represents the cardinality of the set and W represents the set of all Wikipedia topics. This semantic relatedness graph is used for computing PageRank scores. It is noteworthy that "authority" of a learning resource entails author, organisation and content authority [35]. These features represent content authority. The top 5 topic URLs and their relative PageRank Score are included as two feature groups providing 10 distinct features for each video lecture.

The *top-5 covered topic URLs* and *top-5 cosine similarity scores* features represent *Topic Coverage* feature vertical. The cosine similarity score $cos(s_{tr}, c)$ between the *Term Frequency-Inverse Document Frequency (TF-IDF)* representations of the lecture transcript s_{tr} and the Wikipedia page c is calculated using:

$$cos(s_{tr}, c) = \frac{\texttt{TFIDF}(s_{tr}) \cdot \texttt{TFIDF}(c)}{\|\texttt{TFIDF}(s_{tr})\| \times \|\texttt{TFIDF}(c)\|} \tag{18}$$

where $\texttt{TFIDF}(s)$ returns the TF-IDF vector of string s. Topics in the lecture are then ranked using this score. Figure 1 (right) shows the summary of Wikipedia Topics that are most covered (top 1 topic) in the lectures found in the dataset. The top 5 covered topic URLs and their cosine similarity scores are included as two additional feature groups providing 10 distinct features.

Topic authority and topic coverage features represent two different aspects of the content of a video lecture. Authoritative topics are the ones highly connected and dominant within the range of topics that are discussed in the lecture. An authoritative topic needs to have high semantic relatedness to other topics in the lecture. On the contrary, covered topics represent the heavy overlap between individual Wikipedia topics and the lecture transcript. Figure 1 gives further evidence of how these two feature groups are different from each other. The most emerging Wikipedia topics that are authoritative (left) in the lecture dataset are very different from the covered topics (right). The figure also shows that the authoritative topics are narrowly focused concepts (e.g. Machine Learning, Algorithm, Ontology, etc.) whereas the most covered topics tend to be more general topics (e.g. Time, Scientific Method, Unit, etc.).

4.1.3 Video-specific Features

A set of easily automatable features that are video specific are also included in the VLEngagement dataset. Features *Lecture Duration, In Chunked, Lecture Type* and *Speaker Speed* are calculated based on prior work [34]. *Lecture Duration* feature reports the duration of the video in seconds. *Is Chunked* is a binary feature which reports*True* if the lecture consists of multiple videos, and *False* otherwise.

Most Authoritative Topic (PageRank – based) *Most Covered Topic (Cosine – based)*

Figure 1: WordClouds summarising the distribution of the most authoritative (left) and most covered (right) Wikipedia topics in the dataset. Note that Computer Science and Data Science are the two dominant knowledge areas in our dataset.

Table 1: 14 types of lectures in the VLEngagement dataset and their abbriviation (Abbr.) and frequency (Freq).

Abbr.	Description	Freq.	Abbr.	Description	Freq.
`vbp`	Best Paper	16	`vdb`	Debate	30
`vdm`	Demonstration	124	`viv`	Interview	52
`vid`	Introduction	15	`vit`	Invited Talk	300
`vkn`	Keynote	115	`vl`	Lecture	2956
`vop`	Opening	31	`oth`	Other	15
`vpa`	Panel	44	`vps`	Poster	56
`vpr`	Promotional Video	23	`vtt`	Tutorial	269

Lecture type value is derived from the metadata. The possible values for this feature are described in Table 1.

A novel feature *Silence Period Rate (SPR)* is introduced using the "silence" tags that are present in the video lecture transcript. The feature is defined as:

$$\texttt{SPR}(\ell) = \frac{1}{D(\ell)} \sum_{t \in T(\ell)} D(t) \cdot \mathcal{I}(N(t) = \text{"}silence\text{"}) \tag{19}$$

where t is a tag in the collection of tags $T(\ell)$ that belong to lecture ℓ, N returns the type of tag t and D returns the duration of tag t or lecture ℓ and $\mathcal{I}(\cdot)$ is the indicator function (returning 1 when the condition is verified, 0 otherwise).

4.2 Labels

There are several target labels available in the VLEngagement dataset. These target labels are created by aggregating available explicit and implicit feedback measures in the repository. Mainly, the labels can be constructed as three different types of quantification's of learner subjective assessment of a video lecture. The relationship between these different subjective assessments metrics can be investigated with the VLengagement dataset.

4.2.1 Explicit Rating

In terms of rating labels, *Mean Star Rating* is provided for the video lecture using a star rating scale from 1 to 5 stars. As expected, explicit ratings are scarce and thus only populated in a subset of

Table 2: Features extracted and available in the VLEngagement dataset with their variable type (Continuous vs. Categorical) and their quality vertical.

Type	Feature	Quality Vertical
	Metadata features	
cat.	Language (English, non-English)	—
cat.	Domain (STEM, Miscellaneous)	—
	Content-based features	
con.	Word Count	Topic Coverage
con.	Title Word Count	Topic Coverage
con.	Document Entropy	Topic Coverage
con.	Easiness (FK Easiness)	Understandability
con.	Stop-word Presence Rate	Understandability
con.	Stop-word Coverage Rate	Understandability
con.	Preposition Rate	Presentation
con.	Auxiliary Rate	Presentation
con.	To Be Rate	Presentation
con.	Conjunction Rate	Presentation
con.	Normalisation Rate	Presentation
con.	Pronoun Rate	Presentation
con.	Published Date	Freshness
	Wikipedia-based features	
cat.	Top-5 Authoritative Topic URLs	Authority
con.	Top-5 PageRank Scores	Authority
cat.	Top-5 Covered Topic URLs	Topic Coverage
con.	Top-5 Cosine Similarities	Topic Coverage
	Video-based features	
con.	Lecture Duration	Topic Coverage
cat.	Is Chunked	Presentation
cat.	Lecture Type	Presentation
con.	Speaker speed	Presentation
con.	Silence Period Rate (SPR)	Presentation

Table 3: Labels included in the VLEngagement dataset with their variable type, value interval and category.

Type	Label	Interval	Category
cont.	Mean Star Rating	$[1, 5)$	Explicit Rating
cont.	View Count	$(5, \infty)$	Popularity
cont.	SMNET (Eq. 20)	$(0, 1)$	Watch Time
cont.	SANET (Eq. 21)	$[0, 1)$	Watch Time
cont.	Std. of NET	$(0, 1)$	Watch Time
cont.	Number of User Sessions	$(5, \infty)$	Watch Time
cont.	Engagement Times (NET)	$[0, 1)$	Watch Time

resources (1250 lectures). Lecture records are labelled with `-1` where star rating labels are missing. The data source does not provide access to ratings from individual users. Instead, only the aggregated average rating is available.

4.2.2 Popularity

A popularity-based target label is created by extracting the *View Count* of the lectures. The total number of views for each video lecture as of February 17, 2018 is extracted from the metadata and provided with the dataset.

4.2.3 Watch Time/Engagement

The majority of learner engagement labels in the VLEngagement dataset are based on watch time. We aggregate the user view logs and use the Normalised Engagement Time (NET) to compute the **Median of Normalised Engagement (MNET)**, as it has been proposed as the gold standard for engagement with educational materials in previous work [34]. We also calculate the **Average of Normalised Engagement (ANET)**. To have the MNET and ANET labels in the range $[0, 1]$, we set the upper bound to 1 and derive Saturated MNET (`SMNET`) and Saturated ANET (`SANET`) respectively. Final `SMNET` (*Median Engagement*) for lecture ℓ is computed as:

$$\texttt{SMNET}(\ell) = \max(\texttt{MNET}(\ell), 1) \tag{20}$$

Similarly, *Average Engagement* is calculated using:

$$\texttt{SANET}(\ell) = \max(\texttt{ANET}(\ell), 1). \tag{21}$$

The standard deviation of `NET` for each lecture (*Std of Engagement*) is reported, together with the *Number of User Sessions* used for calculating `MNET`. These additional features allow future studies to incorporate the degree of uncertainly and statistical confidence in the engagement labels (e.g. in their loss functions or performance metrics). Furthermore, the individual NET values for each lecture are also provided with the dataset. This allows having much more insight into the true distribution of NET for individual lectures rather than summary statistics. This data will allow future studies to refine engagement labels or use more sophisticated methods to predict engagement.

4.3 Anonymity

We restrict the final dataset to lectures that have been viewed by at least 5 unique users to have reliable engagement measurements. Additionally, a regime of techniques are used for preserving the anonymity of the lectures in order to preserve the identities of the authors/lecturers. The motivation

behind this decision is to avoid authors of the video lectures having unanticipated effects on their reputation by associating implicit learner engagement values to their content.

Rarely occurring values in *Lecture Type* feature were grouped together to create the *other* category found in Table 1. *Language* feature is grouped into `en` and `non-en` categories. Similarly, Domain category groups Life Sciences, Physics, Technology, Mathematics, Computer Science, Data Science and Computers subjects to `stem` category and the other subjects to `misc` category. Rounding is used with *Published Date*, rounding to the nearest 10 days. *Lecture Duration* is rounded to the nearest 10 seconds. Gaussian white noise (10%) is added to *Title Word Count* feature and rounded to the nearest integer.

4.4 Final Dataset

The final dataset includes lectures that are published between September 1, 1999 and October 1, 2017. The engagement labels are created from 155,850 user views logged between December 8, 2016 and February 17, 2018. The final dataset consists of 4,046 lectures across 21 subjects (eg. Computer Science, Philosophy, etc.) that are categorised into STEM and Miscellaneous domains. The dataset, helper tools and example code snippets are available publicly[9]

4.5 Supported Tasks

This section introduces the reader to the tasks that the dataset could be used for. The main application areas of these tasks are quality assurance in open education and scientific content recommenders and understanding and predicting population engagement in an online learning setting. Tasks 1 and 2 are demonstrated in this paper. Tasks 3-6 have been partially tackled in our prior work [38]. Tasks 7-8 are novel.

We establish two main tasks, which we mainly focus on in this paper, that can be objectively addressed using the VLEngagement dataset using a supervised learning approach. These are:

1. **Task 1: Predicting context-agnostic (population-based) engagement of video lectures**: The dataset provides a set of relevant features and labels to construct machine learning models to predict context-agnostic engagement in video lectures. The task can be treated as a regression problem to predict the different engagement labels.

2. **Task 2: Ranking of video lectures based on engagement**: Building predictive models that could rank lectures based on their context-agnostic engagement could be useful in the setting of an educational recommendation system, including tackling the cold-start problem associated to new video lectures. The task can be treated as a ranking problem to predict the global/relative ranking of video lectures.

We further identify several auxiliary tasks that can also be addressed with this dataset:

- **Task 3: Features influencing engagement**: Uncovering the role of different textual and video-specific features involved in several statistics of population-based engagement.

- **Task 4: Influence of topics in engagement**: Understand the role that the topical content in the lecture play on population based engagement (with link to the Wikipedia pages of these topics).

- **Task 5: Disentangle different factors from engagement**: Compare features involved in engagement for different video lecture types, language and knowledge areas (e.g. STEM vs non-STEM lectures).

[9] `https://github.com/sahanbull/context-agnostic-engagement`

- **Task 6: Comparing different measures of implicit and explicit subjective assessment**: Analyse the differences between engagement vs mean star ratings and number of views to identify the strengths and weaknesses of the different feedback types.
- **Task 7: Unsupervised learning to understand the distribution of video lectures**: Cluster video lectures according to the provided features to understand their distribution. Identification of formal patterns that depict similarities and differences between lectures could be insightful.
- **Task 8: Deducing the structure of knowledge**: The co-occurrence patterns of topics within the video lectures provide a great source of data to understand inter-topic relationships and how knowledge is structured. Work in this direction can be used in identifying related materials and accounting for novelty in educational recommendation [7].
- **Task 9: Contrasting to other educational datasets**: The lectures in the VLEngagement dataset are scientific videos, thus it may be meaningful to study if similar patterns for engagement hold across other educational datasets that come from other settings (e.g.: MOOCs).

We propose two baseline models addressing the main tasks (1 and 2) in section 4.6.

4.5.1 Evaluation Metrics

We identify *Root Mean Squared Error (RMSE)* as a suitable metric for Task 1. Measuring RMSE against the original labels published with the datasets will allow different works to be compared fairly. With reference to Task 2, we identify *Spearman's Rank Order Correlation Coefficient (SROCC)* and *Pairwise Ranking Accuracy (Pairwise)*. SROCC can be used to measure the ranking correlation between two global rankings. This metric is suitable for comparing between ranking models that create global rankings (e.g. point-wise ranking algorithms). We outline the exact calculation of RMSE and SROCC in Appendix A.2

However, pairwise ranking accuracy is more intuitive for this task as it represents the fraction of pairwise comparisons where the model could predict the more engaging lecture. There is more than one unique solution for this problem, especially when there is error associated with the ranking model [40].

4.5.2 Hyper-parameter Tuning

We use 5-fold cross validation to evaluate model performance with tasks 1 and 2. We release the folds together with the dataset, to allow for fair comparisons to the baselines. The five folds can be identified using the `fold` column in the dataset. This will allow future work that attempts to improve on the results to compare their work consistently against other works that build on top of this dataset. 5-fold cross validation also allows reporting the *standard error (1.96 × Standard Deviation)* of the performance estimate, which we include in our results.

4.6 Experiments and Baselines

Prior work on similar tasks identify ensemble models [38, 33] to be the best performing models with the main tasks described in section 4.5. We use *Random Forests Regressor (RF)* and *Gradient Boosting Machines (GBM) for constructing baselines.* We use *SMNET* labels as the target variable for both engagement prediction and video lecture ranking tasks. No pre-processing or cleaning steps are necessary.

4.6.1 Features and Labels for Baseline Models

All the features outlined in the content-based and video-based sections in Table 2 are included in the baseline models. However, due to the large amount of topics available in the Wikipedia-based feature groups, we restrict the feature set by adding only the *most authoritative topic URL* and *most covered topic URL*, where both the features are added to the baseline models as categorical variables. Practitioners are encouraged to try further encodings of these variables, as it will likely have a great impact in the performance.

The models are trained with three different feature sets in an incremental fashion:

1. *Content-based*: Features extracted from lecture metadata and the textual features extracted from the lecture transcript.

2. *+ Wiki-based*: In addition to the content-based features, two Wikipedia based features (most authoritative topic URL and most covered topic URL) are added to the feature set. The distribution of topics that are present in the dataset are presented in figure 1

3. *+ Video-based*: In addition to both content-based and Wikipedia-based features, video specific features are added.

This allows identifying the performance gain achieved through adding each new group of features.

Our preliminary investigations indicated that SMNET label follows a Log-Normal distribution, motivating us to use a log transformation on the SMNET values before training the models. Empirical results further confirmed that this step improves the final performance of the models. We undo this transformation for computing $RMSE$.

4.7 Summary of Results

VLEngagement dataset has several limitations that are noteworthy. For example, as the topics in Figure 1 indicate, this dataset is dominated with Computer Science and Data Science related lectures that are mainly delivered in English. In addition, the majority of lectures in the dataset are research talks, narrowing down the style and type of data. These limitations cast significant uncertainty regarding the generalisation of the prediction models to more diverse types of educational video lectures. Although VLEngagement dataset is large compared to the rest of educational engagement datasets available, it still suffers from a limitation in the variety of its data.

We run a series of experiments that will validate the predictive performance of the models with respect to tasks 1 and 2 outlined in section 4.5. As pointed in section 4.6, we evaluate predicting context-free engagement on two ensemble models, GBM and RF models. The results for task 1 shows that the RF model categorically outperforms the GBM model across different feature sets. However, in the context of task 2, GBM models tend to outperform RF model in majority of the cases. The results from both tasks demonstrate that incorporating additional features that include topic coverage signals and modality specific signals lead to performance gains. For an extensive report of the results, we direct the reader to deliverable *D1.6 – Report on Selected Models and Content Representations.*

5 Discussion and Conclusions

Throughout this report, we have looked at the nature of different tasks that are related to maintaining a high quality learning ecosystem (such as language detection, deduplication and personalisation). Based on the nature of the tasks, suitable evaluation methodologies for the solutions have been identified and proposed.

5.1 Personalising Educational Materials

Personalisation of educational materials can be looked at from different viewpoints where different evaluation metrics are sensible in order to get a good idea about the performance of the proposed models. Literature shows that evaluation metrics that belong to quantifying the classification error (with discrete labels), quantifying the exact probability error (with continuous labels) and ranking error (with ordinal labels) have been used for model comparison. Looking at the task at hand, classification metrics such as Accuracy, Precision, Recall and F1-score are appropriate evaluation metrics that can be used to identify the most suitable model. This decision is further reinforced by the fact that the dataset that is used in this task contains labels. The temporal dynamic that is present in the dataset also led us to choose a experimental design that respects the sequential occurrence of events.

5.2 Evaluation of Content Representations

Automatic language detection is a supervised learning problem which can be categorised as a classification task [41]. This led us to choose accuracy score as the primary evaluation metric for the off the shelf models that were evaluated.

On the other hand, we chose precision to evaluate the duplicate detection model as we found no dataset that has global gold standard information that provides labels for the duplicate detection task. Instead, the model was used to detect duplicates which was followed by human annotation of those predictions.

As the dataset that was constructed for maginal engagement prediction contained percentage values, Root Mean Square Error (RMSE) and Pairwise Ranking Accuracy were proposed to evaluate prediction accuracy and ranking accuracy respectively.

5.3 Novel Datasets

Overall, three novel datasets were constructed and created to fill the gaps of scarcity of datasets to evaluate the proposed models.

A novel dataset consisting of aggregated engagement data for over 4,000 scientific videos were constructed with features. A bilingual document set dataset of 8,000 observations was created for evaluating the predictive performance with multilingual documents for eight popular European languages. On the contrary, duplication annotation is a much more labour intensive task which led us to create a much smaller dataset for duplicate detection.

A Appendix

A.1 Computing Classification Metrics

The first step to computing popular classification metrics is to construct the confusion matrix as per figure 2.

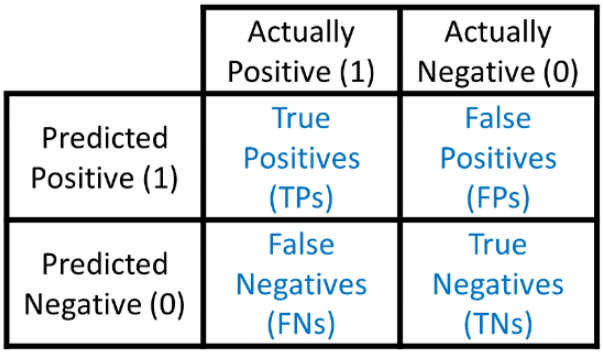

Figure 2: Confusion Matrix, a table that presents how actual labels and predicted labels align with each other. This matrix can be used to identify True Positives (TP), False Positives (FP), False Negatives (FN) and True Negatives (TN) as shown in blue.

From the confusion matrix in figure 2, four statistics can be derived.

1. *True Positives (TPs)*: the number of positive examples that the model correctly classified as positive
2. *True Negatives (TNs)*: the number of negative examples that the model correctly classified as negative
3. *False Positives (FPs)*: the number of negative examples that the model incorrectly classified as positive (i.e. the negative examples that were falsely classified as "positive")
4. *False Negatives (FNs)*: the number of positive examples that the model incorrectly classified as negative (i.e. the positive examples that were falsely classified as "negative")

Using the above statistics, we can compute four classification metrics, namely, accuracy, precision, recall and F1-measure.

A.1.1 Accuracy

Accuracy score quantifies the absolute agreement between the actual labels and the predicted labels in a classification dataset. The definition of Accuracy score is for a dataset of n_ℓ observations for user ℓ is outlined by equation 22.

$$Accuracy(\ell) = \frac{TP_\ell + FP_\ell}{TP_\ell + FP_\ell + TN_\ell + FN_\ell} = \frac{TP_\ell + FP_\ell}{n_\ell} \tag{22}$$

A.1.2 Precision

In the classification context, *precision score* is the fraction of positively classified observations that are truly positive. The definition is outlined by equation 23.

$$Precision(\ell) = \frac{TP_\ell}{TP_\ell + FP_\ell} \tag{23}$$

A.1.3 Recall

In the classification context, *recall score* is the fraction of positively classified observations that are truly positive. The definition is outlined by equation 24.

$$Recall(\ell) = \frac{TP_\ell}{TP_\ell + FN_\ell} \tag{24}$$

A.1.4 F1 Score

F-score or F-measure is a measure that represents the harmonic mean of the precision and recall. The definition of F1-score is depicted in equation 25.

$$F1(\ell) = 2 \times \frac{Precision_\ell \times Recall_\ell}{Precision_\ell + Recall_\ell} \tag{25}$$

A.2 Computing RMSE and SROCC Metrics

A.2.1 Root Mean Square Error (RMSE)

Measuring RMSE against the original labels published with the datasets will allow different works to be compared fairly. The definition of RMSE is found in equation 26.

$$RMSE(y, f(x)) = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\Big(y_i - f(x)_i\Big)^2} \tag{26}$$

where y_i is the actual value, $f(x)_i$ is the predicted value and n is the number of observations in the dataset.

A.2.2 Spearman'r Rank Order Correlation Coefficient (SROCC)

$$SROCC = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \tag{27}$$

where d is the pairwise distances of the ranks of the actual target value y and predicted value $f(x)$ and n is the number of observations in the dataset.

A.3 Tokens used for Feature Extraction in VLEngagement Dataset

Token Set	Description	Tokens
sw	Stopwords	all, show, anyway, fifty, four, go, mill, find, seemed, one, whose, re, herself, whoever, behind, should, to, only, under, herein, do, his, get, very, de, none, cannot, every, during, him, did, cry, beforehand, these, she, thereupon, where, ten, eleven, namely, besides, are, further, sincere, even, what, please, yet, couldnt, enough, above, between, neither, ever, across, thin, we, full, never, however, here, others, hers, along, fifteen, both, last, many, whereafter, wherever, against, etc, s, became, whole, otherwise, among, via, co, afterwards, seems, whatever, alone, moreover, throughout, from, would, two, been, next, few, much, call, therefore, interest, themselves, thr, until, empty, more, fire, latterly, hereby, else, everywhere, former, those, must, me, myself, this, bill, will, while, anywhere, nine, can, of, my, whenever, give, almost, is, thus, it, cant, itself, something, in, ie, if, inc, perhaps, six, amount, same, wherein, beside, how, several, whereas, see, may, after, upon, hereupon, such, a, off, whereby, third, i, well, rather, without, so, the, con, yours, just, less, being, indeed, over, move, front, already, through, yourselves, still, its, before, thence, somewhere, had, except, ours, has, might, thereafter, then, them, someone, around, thereby, five, they, not, now, nor, name, always, whither, t, each, become, side, therein, twelve, because, often, doing, eg, some, back, our, beyond, ourselves, out, for, bottom, since, forty, per, everything, does, three, either, be, amongst, whereupon, nowhere, although, found, sixty, anyhow, by, on, about, anything, theirs, could, put, keep, whence, due, ltd, hence, onto, or, first, own, seeming, formerly, into, within, yourself, down, everyone, done, another, thick, your, her, whom, twenty, top, there, system, least, anyone, their, too, hundred, was, himself, elsewhere, mostly, that, becoming, nobody, but, somehow, part, with, than, he, made, whether, up, us, nevertheless, below, un, were, toward, and, describe, am, mine, an, meanwhile, as, sometime, at, have, seem, any, fill, again, hasnt, no, latter, when, detail, also, other, take, which, becomes, yo, towards, though, who, most, eight, amongst, nothing, why, don, noone, sometimes, together, serious, having, once, hereafter
conj	Conjunctions	and, but, or, yet, nor
norm	Normalizations	-tion, -ment, -ence, -ance
tobe	To-be Verbs	be, being, was, were, been, are, is
prep	Prepositions	aboard, about, above, according to, across from, after, against, alongside, alongside of, along with, amid, among, apart from, around, aside from, at, away from, back of, because of, before, behind, below, beneath, beside, besides, between, beyond, but, by means of, concerning, considering, despite, down, down from, during, except, except for, excepting for, from among, from between, from under, in addition to, in behalf of, in front of, in place of, in regard to, inside of, inside, in spite of, instead of, into, like, near to, off, on account of, on behalf of, onto, on top of, on, opposite, out of, out, outside, outside of, over to, over, owing to, past, prior to, regarding, round about, round, since, subsequent to, together, with, throughout, through, till, toward, under, underneath, until, unto, up, up to, upon, with, within, without, across, long, by, of, in, to, near, of, from
auxi	Auxiliary Verbs	will, shall, cannot, may, need to, would, should, could, might, must, ought, ought to, can't, can
pron	Pronouns	i, me, we, us, you, he, him, she, her, it, they, them, thou, thee, ye, myself, yourself, himself, herself, itself, ourselves, yourselves, themselves, oneself, my, mine, his, hers, yours, ours, theirs, its, our, that, their, these, this, those

References

[1] Erik Novak, Jasna Urbančič, and Miha Jenko. Preparing multi-modal data for natural language processing. In *Proc. of Slovenian KDD Conf. on Data Mining and Data Warehouses (SiKDD)*, 2018.

[2] Sahan Bulathwela, Stefan Kreitmayer, and María Pérez-Ortiz. What's in it for me? augmenting recommended learning resources with navigable annotations. In *Proceedings of the 25th International Conference on Intelligent User Interfaces Companion*, IUI 20, 2020.

[3] Jan M. Pawlowski, Volker Zimmermann, and Imc Ag. Open content: A concept for the future of e-learning and knowledge management?, 2007.

[4] Albert T. Corbett and John R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4), 1994.

[5] Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pages 171–180. Springer, 2013.

[6] Weijie Jiang, Zachary A. Pardos, and Qiang Wei. Goal-based course recommendation. In *Proceedings of International Conference on Learning Analytics & Knowledge*, 2019.

[7] S. Bulathwela, M. Perez-Ortiz, E. Yilmaz, and J. Shawe-Taylor. Truelearn: A family of bayesian algorithms to match lifelong learners to open educational resources. In *AAAI Conference on Artificial Intelligence*, 2020.

[8] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, 2015.

[9] Jill-Jênn Vie and Hisashi Kashima. Knowledge tracing machines: Factorization machines for knowledge tracing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 750–757, 2019.

[10] Theophile Gervet, Ken Koedinger, Jeff Schneider, Tom Mitchell, et al. When is deep learning the best approach to knowledge tracing? *JEDM| Journal of Educational Data Mining*, 12(3):31–54, 2020.

[11] Mojtaba Salehi, Isa Nakhai Kamalabadi, and Mohammad Bagher Ghaznavi Ghoushchi. Personalized recommendation of learning material using sequential pattern mining and attribute based collaborative filtering. *Education and Information Technologies*, 19(4):713–735, 2014.

[12] Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill(tm): A bayesian skill rating system. In *Advances in Neural Information Processing Systems 20*, pages 569–576. MIT Press, January 2007.

[13] Juraj Nižnan, Radek Pelánek, and Jirí Rihák. Student models for prior knowledge estimation. *International Educational Data Mining Society*, 2015.

[14] Hao Cen, Kenneth R. Koedinger, and Brian Junker. Is over practice necessary? –improving learning efficiency with the cognitive tutor through educational data mining. In *Proceedings of the 2007 Conference on Artificial Intelligence in Education: Building Technology Rich Learning Contexts That Work*, page 511–518, NLD, 2007. IOS Press.

[15] Zachary A Pardos and Neil T Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 255–266. Springer, 2010.

[16] Radek Pelánek. Metrics for evaluation of student models. *Journal of Educational Data Mining*, 7(2):1–19, 2015.

[17] Guy Shani and Asela Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, 2011.

[18] Javier Iranzo, Alex P´erez, Jorge Civera, Albert Sanchis, and Alfons Juan. Deliverable 3.4 - early support for cross-lingual oer. `https://www.x5gon.org/wp-content/uploads/2019/10/D3.4_afterColinRev_26Aug19.pdf`. Accessed in: 2020-12-02.

[19] J. Jorge and Alfons Juan. Deliverable 3.5 - final support for cross-lingual oer. `https://www.x5gon.org/wp-content/uploads/2020/06/D3.5_afterErikRev_25Feb2020.pdf`. Accessed in: 2020-12-02.

[20] Janez Brank, Gregor Leban, and Marko Grobelnik. Annotating documents with relevant wikipedia concepts. In *Proc. of Slovenian KDD Conf. on Data Mining and Data Warehouses (SiKDD)*, 2017.

[21] S. Bulathwela, M. Perez-Ortiz, E. Yilmaz, and J. Shawe-Taylor. D1.3 – initial content representations. `https://www.x5gon.org/wp-content/uploads/2019/10/X5GON_Deliverable_D1_3.pdf`. Accessed in: 2020-12-02.

[22] Marco Ponza, Paolo Ferragina, and Soumen Chakrabarti. On computing entity relatedness in wikipedia, with applications. *Knowledge-Based Systems*, 188, 2020.

[23] X5GON. X5gon connect. `https://platform.x5gon.org/products/connect`. Accessed in: 2020-12-02.

[24] Martin Thoma. WiLI-2018 - Wikipedia Language Identification database, January 2018.

[25] Timothy Baldwin and Marco Lui. Multilingual language identification: ALTW 2010 shared task data. In *Proceedings of the Australasian Language Technology Association Workshop 2010*, pages 4–7, Melbourne, Australia, December 2010.

[26] Scikit learn Developers. Multiclass and multilabel algorithms. `https://scikit-learn.org/stable/modules/multiclass.html`. Accessed: 2020-12-01.

[27] Marzieh Oghbaie and Morteza Mohammadi Zanjireh. Pairwise document similarity measure based on present term set. *Journal of Big Data*, 5(1):52, 2018.

[28] Yasunao Takano, Yusuke Iijima, Kou Kobayashi, Hiroshi Sakuta, Hiroki Sakaji, Masaki Kohana, and Akio Kobayashi. Improving document similarity calculation using cosine-similarity graphs. In *International Conference on Advanced Information Networking and Applications*, pages 512–522. Springer, 2019.

[29] N. Erik. Deliverable 2.2 - final server-side platform. `https://www.x5gon.org/wp-content/uploads/2019/10/D2.2-Final-server-side-platform-FINAL.pdf`. Accessed in: 2020-12-02.

[30] Michael Bendersky, W. Bruce Croft, and Yanlei Diao. Quality-biased ranking of web documents. In *Proc. of ACM Int. Conf. on Web Search and Data Mining*, 2011.

[31] Daniel Hasan Dalip, Marcos André Gonçalves, Marco Cristo, and Pável Calado. Automatic assessment of document quality in web collaborative digital libraries. *Journal of Data and Information Quality*, 2(3), December 2011.

[32] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In *Proc. of Int. Conf. on World Wide Web*, 2006.

[33] Morten Warncke-Wang, Dan Cosley, and John Riedl. Tell me more: An actionable quality model for wikipedia. In *Proc. of Int. Symposium on Open Collaboration*, WikiSym '13, 2013.

[34] Philip J. Guo, Juho Kim, and Rob Rubin. How video production affects student engagement: An empirical study of mooc videos. In *Proc. of the First ACM Conf. on Learning @ Scale*, 2014.

[35] Sahan Bulathwela, Emine Yilmaz, and John Shawe-Taylor. Towards Automatic, Scalable Quality Assurance in Open Education. In *Workshop on AI and the United Nations SDGs at Int. Joint Conf. on Artificial Intelligence*, 2019.

[36] Daniel H. Dalip, Marcos André Gonçalves, Marco Cristo, and Pável Calado. A general multiview framework for assessing the quality of collaboratively created content on web 2.0. *Journal of the Association for Information Science and Technology*.

[37] E. Hengel. Publishing while Female. Are women held to higher standards? Evidence from peer review. Cambridge Working Papers in Economics 1753, 2017.

[38] Sahan Bulathwela, Maria Perez-Ortiz, Aldo Lipani, Emine Yilmaz, and John Shawe-Taylor. Predicting engagement in video lectures. In *Proc. of Int. Conf. on Educational Data Mining*, EDM '20, 2020.

[39] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proc. of Int. Conf. on World Wide Web*, 1998.

[40] Johannes Fürnkranz and Eyke Hüllermeier. *Preference Learning and Ranking by Pairwise Comparison*, pages 65–82. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[41] Tae Yano and Moonyoung Kang. Taking advantage of wikipedia in natural language processing. Technical report, Technical report, Carnegie Mellon University Language Technologies Institute, 2016.