

# ConstructED: Constructing Tailored Educational Datasets from Online Courses

Aymen A. Bazouzi<sup>1</sup><sup>a</sup>, Zoltan Miklos<sup>1</sup><sup>b</sup>, Mickaël Foursov<sup>1</sup><sup>c</sup> and Hoël Le Capitaine<sup>2</sup><sup>d</sup>

<sup>1</sup>Univ. Rennes CNRS IRISA, France

<sup>2</sup>Nantes Université, LS2N, UMR 6004, F-44000 Nantes, France

Keywords: Educational Resources, Datasets.


Abstract: Researchers are actively involved in developing various systems to support education, including recommender systems. However, to create and evaluate such systems, they require rich and versatile datasets about educational content. At times, the available data proves insufficient, leading researchers to invest significant time in crafting personalized web scrapers for additional data retrieval. The generated datasets are often task-specific and may be time-consuming to adapt to future tasks. Additionally, researchers may encounter licensing issues when using courses from different providers. Furthermore, researchers prefer evaluating their methods through diverse tests, involving datasets with varying characteristics. However, this diversity is not commonly found in most available datasets, at least not explicitly so. To address these challenges, we introduce ConstructED, a tool built on top of Google APIs, enabling the efficient creation of custom educational datasets from YouTube playlists. This allows datasets to be tailored to specific characteristics such as a predetermined number of courses, coverage of specific topics, or courses from a particular university. ConstructED creates datasets from video course transcripts, providing a ready-to-use solution that significantly shortens the time required to create such datasets. The resulting datasets are versatile and suitable for tasks like classification and learning path creation.


## 1 INTRODUCTION


Educational data has been the fuel for many breakthroughs in the domain of education (Romero and Ventura, 2013). There are different types of educational data available. We can find data about the content, the students' answers, students' behavior, etc (Ferreira-Mello et al., 2019). However, in this article we are interested in educational datasets about the content. Datasets about content regroup educational materials such as articles, books, videos, blogs, or any other medium that can be used to convey educational knowledge. More specifically, we are interested in content that is textual or from which text can be extracted. Using this type of educational content, we can learn new things about the content itself or even about the learners who consume it and their behavior. We can make the learning process more effi-


cient by creating support systems for learners such as recommender systems (Urdaneta-Ponte et al., 2021) and personalize their learning paths (Nabizadeh et al., 2020). It can also help the educators create better courses more efficiently by giving them access to useful resources online and providing them with the necessary tools to query them efficiently. To tackle these interesting challenges, researchers need good quality datasets to develop their support systems and analyze the educational content.

Textual education datasets can be extracted from different sources. For example, online learning platforms are widely used as a data source. Their websites can be scrapped in order to gather data. University libraries and online courses are also another popular data source. However, these data sources can be hard to exploit sometimes. For instance, we might need different web scrapers for different platforms. Furthermore, not all platforms allow the exploitation of their resources which can lead to copyright issues. Another problem researchers face is the heterogeneity of datasets. This can pose challenges for researchers who wish to test their methods on diverse datasets or

<sup>a</sup> <https://orcid.org/0009-0004-5209-6494>

<sup>b</sup> <https://orcid.org/0000-0002-3701-6263>

<sup>c</sup> <https://orcid.org/0009-0002-1048-8663>

<sup>d</sup> <https://orcid.org/0000-0002-7399-0012>

combine multiple datasets to create a larger dataset better suited to their needs. YouTube can be viewed as an alternative for these learning platforms as a lot of educational videos can be found on it. Nevertheless, it requires investing a lot of time to manipulate the different APIs to get the data needed from it.

In this article, we present a tool that can be used to create textual education datasets for research purposes from YouTube videos called ConstrucTED. The use of this tool exempts users wanting to create datasets from coding or explicitly manipulating APIs. Our contribution is two-fold :

- The tool itself which takes a list of YouTube playlists, extracts the transcripts from the videos then constructs a dataset from these transcripts as well as other meta-information. The constructed datasets are structured in a manner that allows it to be used for different tasks such as classification and learning path creation.
- Three datasets that were created using this tool using educational content from Stanford, MIT OpenCourseWare, and Khan Academy.

In Section 2 we present the tool and its different components. Next, in Section 3 we present three sample datasets created using this tool. Then, in Section 4 we discuss the use cases in which datasets constructed using this tool can be used. In Section 5, we discuss the tool's limitations as well as the ethical concerns and potential risks. We conclude in Section 6 by recapitulating the work and discussing different possible ways to improve this tool.

## 2 TOOL PRESENTATION

ConstrucTED constructs educational datasets from playlists found on YouTube. Figure 1 illustrates the process of constructing the dataset. It starts by extracting and structuring the information of the YouTube videos found in the playlists using the Google API Client<sup>1</sup>. Then, for every video it extracts the transcripts using the YouTube Transcript API<sup>2</sup>. In the rest of this section, we will present this tool in detail.

### 2.1 Input Format

To create a dataset, an input file must be provided. This file must be a *csv* file containing three columns (Figure 2). The first column should be the playlist

<sup>1</sup><https://developers.google.com/api-client-library>

<sup>2</sup><https://pypi.org/project/youtube-transcript-api/>

ID, the second being the Channel name, and the third one is the category. For the playlist ID, it is only the ID and not the entire URL. This avoids the redundancy and thus reduces the size of the input file. The channel name, despite the possibility to automatically extract it from the link provided, we gave the users the liberty to name it as there are playlists that contain videos from different channels. This second column is not necessarily a channel name, it can be used to distinguish the sources of the different playlists. Finally, the category is used to give the topic of the playlists. This can be helpful to separate the dataset following the topics covered by each playlist which can be helpful to construct a dataset that will be used in a topic classification task.

### 2.2 Transcript Preprocessing

After extracting the video links from the playlist, transcripts are extracted from every video using YouTube Transcript API. However, there is some preprocessing that can be done. For example, we can remove unnecessary words that can be found in some video transcripts such as the presence of onomatopoeic words, words that inform us about the person currently talking, or special characters such as the end of line (EOL) symbol. Examples of these words are : "[PROFESSOR], [Voiceover], [AUDIENCE], [APPLAUSE], [INAUDIBLE], [Music], [SQUEAKING], [RUSTLING], [CLICKING], \n, ...". The list of unwanted words can be updated by the user, she can add, remove, or even choose not to do any preprocessing.

### 2.3 Hierarchy Levels

To create datasets that can be used in diverse tasks, we provide the users with a three-level hierarchical structure as a foundational framework. As the input file consists of a list of playlists, a two-level hierarchical structure emerges naturally. The first one being that of playlists and the second one is that of videos. However, users might wish to further subdivide videos into segments based on chapter markers present in YouTube videos or to address lengthy videos exceeding the scope of their intended task. This flexibility is offered through the three-level hierarchy, allowing users to organize their datasets according to their specific needs.

Figure 3 illustrates an example of how different elements are mapped to three hierarchical levels to create a dataset:

- The first level, called series, represents playlists: Every playlist used to create the dataset is mapped

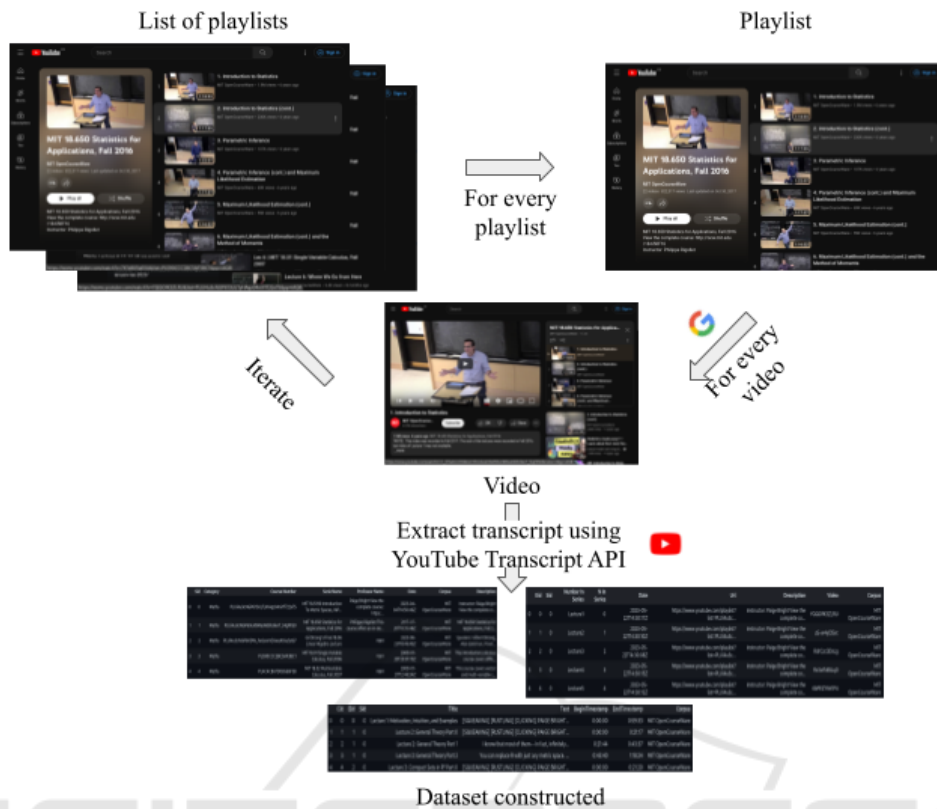


Figure 1: Constructing datasets from YouTube playlists using ConstrucTED.

	Playlist Title	Channel	Category
0	PL7A9646BC5110CF64	Khan Academy	Biology
1	PLSQI0a2vh4HDERCw_ddanXbsDpFWcpL-S	Khan Academy	Economics
2	PLFD0EB975BA0CC1E0	Khan Academy	Maths
3	PLSQI0a2vh4HDCasLssY8bUV2qwDkZeOYL	Khan Academy	Maths
4	PLD6DA74C1DBF770E7	Khan Academy	Maths

Figure 2: Input example used by ConstrucTED.

to a series.

- The second level, called episodes, represents videos: Each video within a playlist is considered an episode.
- The last level, called chapters, represents video segments: If users choose not to segment the videos, the number of episodes and chapters in the dataset will be equal. Otherwise, the number of chapters will be equal to the total number of segments in all the videos combined.

For now, only video segmentation by length is available. The user can specify the max length of a YouTube video and in case a video surpasses it, it gets divide into segments of a length chosen by the user himself. In Figure 3, we can see an example of this segmentation as the video already contains predefined segments.

## 2.4 Meta-Information

While constructing educational datasets, including meta-information about different courses can be valuable. In our tool, we extract certain meta-information, such as the professor who conducted the course. Typically, the name of the professor is included in the video description. We have implemented a functionality based on regular expressions to extract the professor’s name from the description. Despite its utility, this feature has limitations; the professor’s name might not be mentioned, and even when it is, extracting it can be challenging due to variations in formulations.

In addition to the professor’s name, we also extract timestamps for the beginning and ending of each chapter. This allows us to easily navigate back to specific points in the video, facilitating the review of visual content such as figures.

## 2.5 Results

The tool generates three separate *csv* files, each corresponding to one of the previously described hierarchy level. These files contain various attributes further explained in the appendix.

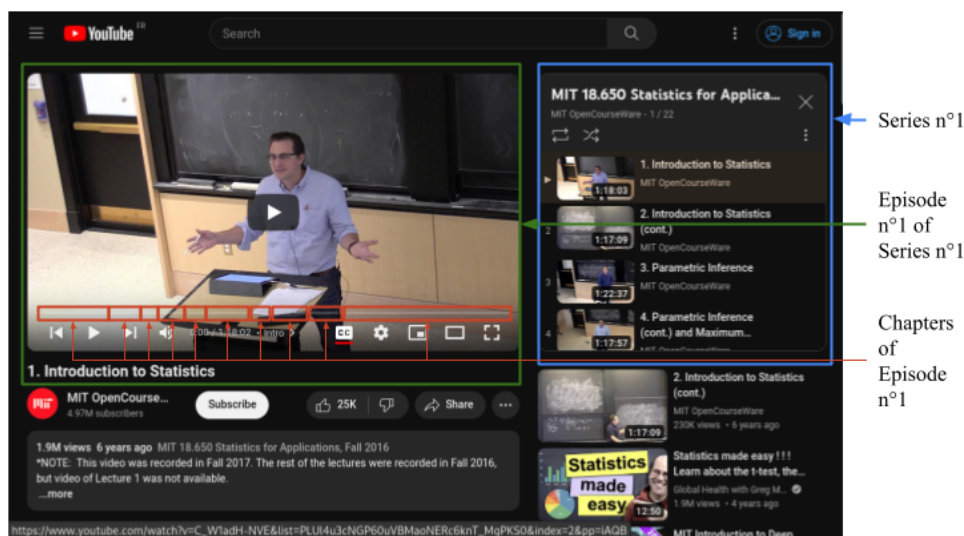


Figure 3: Hierarchy levels example.

We plan to make the tool available as a python package if the paper is accepted. Currently, only the code is available <sup>3</sup>.

## 2.6 Tailored Datasets

One of the main advantages of ConstructED is its ability to empower users in creating tailored datasets. If users wish to test their methods, they can generate diverse datasets, manipulating characteristics like the number of courses, the academic level, the covered topics, and the course provider.

For instance, if a user aims to build a dataset on K-12 courses from Khan Academy, they can either utilize existing playlists available on their YouTube channel that align with the specified criteria or curate new playlists containing hand-picked video courses. These playlists can then be provided as input to ConstructED. Another illustration involves crafting a dataset encompassing Master’s level courses in Computer Science, a task easily achieved by utilizing available Stanford Computer Science YouTube playlists, for example.

In the following section, we will showcase sample datasets crafted using ConstructED. These datasets exhibit diverse characteristics, including variations in course length, covered topics, and course providers.

## 3 DATASET CONSTRUCTION

Naturally, the datasets constructed are dependant on the user’s needs. However, we present in this sec-

<sup>3</sup><https://github.com/AymenRaouf/ConstructED>

tion three sample datasets that were created using this tool. These three datasets have been constructed from YouTube playlists from Khan Academy<sup>4</sup>, Stanford<sup>5</sup>, and MIT OpenCourseWare<sup>6</sup>. A snippet of the input file for the Khan Academy is illustrated in Figure 2 in which we can see the three columns and the first few lines of its content. The characteristics of the created datasets can be found in Table 1. These datasets are available for download in the code repository found at the end of Section 2.5.

Table 1: Characteristics of the created datasets.

Stat	Khan	Stanford	MIT
Series	15	11	42
Episodes	1039	442	1030
Chapters	1039	1010	2277
Avg episodes per series	69.26	40.18	24.52
Avg chapters per episode	1.0	2.28	2.21
Avg words per chapter	1692.97	2383.95	3064.52
N° of categories	10	1	10

From Table 1, it is evident that the datasets exhibit distinct characteristics. For instance, in the Khan dataset, the number of chapters is the same as the number of episodes because the videos used are short and have not been segmented. In contrast, the other two datasets employ university courses, featur-

<sup>4</sup><https://www.youtube.com/@khanacademy>

<sup>5</sup><https://www.youtube.com/@stanfordonline>

<sup>6</sup><https://www.youtube.com/@mitocw>

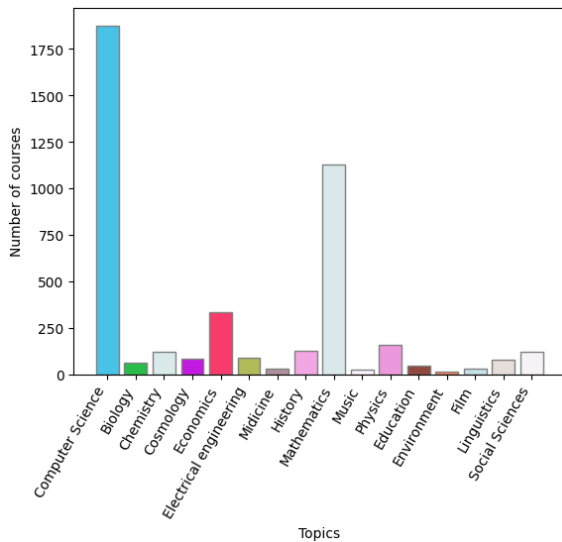


Figure 4: Combined number of courses per topic.

ing longer lecture videos. Furthermore, Stanford is a Computer Science dataset, while Khan and MIT cover different topics. Figure 4 displays the distribution of chapters per topic for the three datasets combined. These highlighted differences (size and number of topics) between the datasets make them valuable for researchers who want to test and analyze their methods.

The time needed to construct various datasets using this tool cannot be estimated beforehand due to various variables that impact it. For example, these datasets are extracted from the internet, so internet speed influences the required time. Furthermore, the number of playlists and the size of the videos also play a role in determining the time needed. Among the three datasets created, the MIT dataset was the largest, requiring 15 minutes and 14 seconds for its creation.

## 4 POTENTIAL USAGE

The datasets created using ConstructED are structured in a format that enables them to be utilized for various tasks. They contain important qualities such as sequentiality and the presence of topics. Some of the use cases for these datasets are :

**Precedence Identification.** Given the sequential nature of the playlists from which the dataset has been constructed, we can deduce a precedence order between the educational resources that constitute the dataset. Therefore, the datasets constructed can be

used as a benchmark for precedence prediction between resources by exploiting the raw text and the ground truth given by the order (Connes et al., 2021).

**Learning Path Creation.** Following the reasoning from the previous potential use case, we can use the order found in the dataset as a ground truth of potential learning paths (Nabizadeh et al., 2020). This makes the datasets constructed useful in a learning path creation scenario with an implicit hypothesis that states that playlists used contain coherent learning paths.

**Recommender Systems.** Another similar use case are recommender systems. Recommender systems suggest content to users that is potentially interesting and/or useful. In education, there are multiple recommender systems that have been presented (Urdaneta-Ponte et al., 2021). The first taxonomy about recommender systems contains three main system classes (Adomavicius and Tuzhilin, 2005). Some are based on recommending the same content for similar users. Others recommend content that is similar to the content previously consumed by the user. We can also find systems that combine both of the previous approaches. The datasets constructed using this tool can be used in the second type which is more formally known as content-based recommender systems.

**Topic Classification.** In the input file, we can specify the category to which playlists belong. This can be used to group the educational texts by categories, which can be topics or domains, then, perform a category classification task by training a model on this data for example (Li and Jain, 1998), (Bazouzi et al., 2023). Additionally, there is the possibility of manually annotating the dataset to perform other classification tasks.

**Text Representation.** LLMs have gained a lot of popularity recently. A lot of these models are trained in a non-supervised manner. Therefore, the educational text found on the constructed datasets can be used to train an LLM if the size of the dataset is large enough. We can even fine tune a pre-trained LLM such as BERT (Devlin et al., 2019), (Sun et al., 2019).

**Concept Graph Learning.** Using Wikification (Hoffart et al., 2011) or other techniques, we can extract concepts for every educational text in the dataset. This can allow us to create a fourth file that contains a list of concepts and the chapters to which they belong. These concepts can be used to construct concept

graphs (Yang et al., 2015). The goal of concept graphs is to create a directed graph that indicates the prerequisite order between different concepts. One simple way to do this with such datasets is to use the hypothesis that states that if a resource A precedes another resource B, it is more likely for concepts related to resource A to be prerequisites of concepts related to resource B (Liang et al., 2015).

## 5 DISCUSSION

The tool that we constructed can be used in different scenarios as explained in the previous section. However, it has some limitations and some constraints that the user must keep in mind while using it.

### 5.1 Limitations

The first limitation that a user can face is related to the transcripts for the video. Although a lot of channels provide official transcripts for their videos, there are others that do not. Fortunately, YouTube has an automatic transcript generation feature. This feature is useful for videos for which no transcripts have been provided but since the transcripts are not official, there is a risk of the transcripts being incorrect.

One more limitation is the nature of the course. Since these texts are constructed from videos, we might find portions of the videos that are non-related to the course such as an out-of-topic discussion professors can have with their students. Although this phenomenon is not common but it remains important to keep in mind.

One last limitation is the quality of the playlists used. The quality of the constructed dataset depends on the quality of the playlists used. If the chosen playlists are not coherent or if the playlists contain intruder videos, this can deteriorate the quality of the generated dataset. To solve this limitation we suggest the creation of personalized playlists on the user's YouTube account then using these created playlists to create the dataset. This process certainly takes more time but it allows for the quality of the dataset to be improved and adapted to the user's needs.

One more thing worth mentioning, despite not being a limitation, is the possibility of using this tool to construct non-educational datasets. Although this tool can construct non-educational datasets, we are unsure of the results produced. It is clear that the tool will be able to extract transcripts and organize them, but we have not tested it on playlists that do not cover educational content.

### 5.2 Ethical Concerns

The videos on YouTube have by default a YouTube standard license. However, content posted on YouTube can be marked with a Creative Commons license<sup>7</sup>. This makes it legal for other users to use this type of content. More about copyrights and licenses can be found on YouTube's terms of service<sup>8</sup>.

After UNSECO's recommendation about Open Educational Resources (OERs)<sup>9</sup>, a lot of universities have joined this initiative that consists of releasing learning, teaching and research materials under an open license. This makes most of the content produced by these universities safe to use.

### 5.3 Potential Risks

As our tool leverages YouTube Data API and Google API Client, which require authentication, users need to obtain a personal API key to extract data. However, it is crucial to emphasize that this key is confidential and tied to the user's account. We strongly advise users to exercise caution and avoid sharing their key, even within collaborative environments. For optimal security, store your key locally. More details about this will be released in the official documentation of the tool.

## 6 CONCLUSIONS

In this article, we introduced ConstructED, a ready-to-use tool for constructing educational datasets from YouTube playlists built on top of Google APIs. The tool's primary objective is to minimize the time required for researchers to create datasets by omitting any coding or explicit API manipulation. Additionally, it provides researchers with the flexibility to tailor their datasets to meet specific requirements, such as covering particular topics, being sourced from specific universities, or having a predefined number of lectures. The structured format of the created datasets allows for their versatile application across various scenarios and facilitates the accomplishment of diverse tasks.

We discussed the limitations of the tool, such as the datasets' quality, which depends on the quality of

<sup>7</sup><https://creativecommons.org/licenses/by/3.0/legalcode>

<sup>8</sup>[https://support.google.com/youtube/topic/2676339?hl=en&ref\\_topic=6151248&sjid=6402530853925201312-EU](https://support.google.com/youtube/topic/2676339?hl=en&ref_topic=6151248&sjid=6402530853925201312-EU)

<sup>9</sup><https://www.unesco.org/en/legal-affairs/recommendation-open-educational-resources-oer>

the playlists used. Additionally, we addressed ethical concerns that users must consider while using this tool, including potential license issues.

For future work, we plan to enhance the tool in various ways. For example, we aim to incorporate an automatic license detection feature that informs the user about the license used for each video, thereby helping to avoid legal issues. The segmentation of chapters can also be significantly improved. Specifically, we can integrate a topic detection layer, enabling the tool to identify changes in topics and segment the transcripts accordingly (Vayansky and Kumar, 2020). Additionally, we intend to enhance text preprocessing and provide users with more options in choosing how the text should be preprocessed to align with their specific needs.

## ACKNOWLEDGEMENTS

This work has received a French government support granted to the Labex Cominlabs excellence laboratory and managed by the National Research Agency in the “Investing for the Future” program under reference ANR-10-LABX-07-01.

We extend our sincere appreciation to Omonliwi Graciela Thoo for her contribution in implementing a first version of the code.

## REFERENCES

- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749.
- Bazouzi, A. A., Foursov, M., Le Capitaine, H., and Miklos, Z. (2023). EMBEDD-ER : EMBEDDing Educational Resources Using Linked Open Data. In Proceedings of the 15th International Conference on Computer Supported Education, page 439–446, Prague, Czech Republic.
- Connes, V., de La Higuera, C., and Le Capitaine, H. (2021). What should i learn next? ranking educational resources. In *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, page 109–114. IEEE.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. (arXiv:1810.04805). arXiv:1810.04805 [cs].
- Ferreira-Mello, R., Andre, M., Pinheiro, A., Costa, E., and Romero, C. (2019). Text mining in education. *WIREs Data Mining and Knowledge Discovery*, 9(6):e1332.
- Hoffart, J., Yosef, M. A., Bordino, I., Furstenuau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., and Weikum, G. (2011). Robust disambiguation of named entities in text. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, page 782–792.
- Li, Y. H. and Jain, A. K. (1998). Classification of text documents. *The Computer Journal*, 41(8):537–546.
- Liang, C., Wu, Z., Huang, W., and Giles, C. L. (2015). Measuring prerequisite relations among concepts. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, page 1668–1674.
- Nabizadeh, A. H., Leal, J. P., Rafsanjani, H. N., and Shah, R. R. (2020). Learning path personalization and recommendation methods: A survey of the state-of-the-art. *Expert Systems with Applications*, 159:113596.
- Romero, C. and Ventura, S. (2013). Data mining in education. *WIREs Data Mining and Knowledge Discovery*, 3(1):12–27.
- Sun, C., Qiu, X., Xu, Y., and Huang, X. (2019). How to fine-tune bert for text classification? In Sun, M., Huang, X., Ji, H., Liu, Z., and Liu, Y., editors, *Chinese Computational Linguistics*, Lecture Notes in Computer Science, page 194–206, Cham. Springer International Publishing.
- Urdaneta-Ponte, M. C., Mendez-Zorrilla, A., and Oleagordia-Ruiz, I. (2021). Recommendation systems for education: Systematic review. *Electronics*, 10(1414):1611.
- Vayansky, I. and Kumar, S. A. P. (2020). A review of topic modeling methods. *Information Systems*, 94:101582.
- Yang, Y., Liu, H., Carbonell, J., and Ma, W. (2015). Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, page 159–168.

## APPENDIX

### 6.1 Dataset Attributes

Tables 2, 3, and 4 present the attributes of the created datasets.

Table 2: Main attributes of the series output file.

Attribute	Description
Sid	Series unique ID in the dataset
Category	The category (topic, domain, ...) specified by the user in the input file
Course	Playlist ID in YouTube
Title	Title of the playlist in YouTube
Created at	Date of the last update of the YouTube playlist
Corpus	Name of the channel specified by the use in the input file
Professor	Name of the professor extracted from the description
Description	The description section extracted from the YouTube playlist

Table 3: Main attributes of the episodes output file.

Attribute	Description
Eid	Episode unique ID in the dataset
Sid	Series to which the episode belongs
Order	Order of the episode in the series
Title	The video title in YouTube
Created at	Publication date of the video
Description	The description section extracted from the YouTube video
Course	Video ID in YouTube

Table 4: Main attributes of the chapters output file.

Attribute	Description
Cid	Chapter unique ID in the dataset
Eid	Episode to which the chapter belongs
Title	Title of the video + the n° of the part if the video was segmented
Text	Text extracted from the video transcript
Begin	Beginning timestamp of the segment on the video
End	Ending timestamp of the segment on the video