Analysis of Video Lessons: a Case for Smart Indexing and Topic Extraction

Marco Arazzi [1], Marco Ferretti [1] and Antonino Nocera [1]

[1] *Department of Electrical, Computer and Biomedical Engineering, University of Pavia, Via Ferrata 5, 27100 Pavia, Italy*

Abstract

On-line teaching activity during the pandemic has generated huge amounts of data, mainly in the form of video. Leveraging this rich source of didactic material requires at least two somewhat complementary facilities: smart indexing into the corpus of lessons, and automatic extraction of a list of topics that best represent the main subject of a course, based on the set of its lessons. This paper shows a preliminary attempt to address both issues, using material produced at the University of Pavia in the year 2020, specifically a prototype bachelor course to develop the tools and the methods to be later applied on the big data repository of the whole set of lessons available.

Keywords

Video indexing, topic modeling, online lessons, university repositories

1. Introduction

One, possibly the only single nice, side effect of the pandemic storm that has recently raged all over the world is the strong thrust by public and private teaching institutions to deliver on their mission by resorting to online teaching in various forms, with different technologies and platforms. The most common means to keep teaching on has been the provision for streaming video lessons, and in many cases for setting up huge archives of recorded video and audio lessons.

As an instance of this pattern, at the University of Pavia online lesson delivery was carried out starting with the spring semester of 2020, which began in almost all tracks at the end of February, and lasted throughout the year, continuing on in the winter semester 2020/2021, even if it was somehow intermingled with on-premise activity. Indeed, even when the pandemic allowed for on-premise delivery of lessons, this was combined with reduced-number classes, so that students attended alternatively from home and from regular university rooms; so, video streaming continued along with lessons recording.

This alternating mode of lessons attendance called for quick access to online video material, so that a student could watch again a whole lesson or, more profitably, just rehearse those sections that required further attention. Online access to the video for sequential reading and for inspection when looking for an a-priori unknown spot containing the interesting material can be a frustrating experience; furthermore, often a subject is treated in many spots, and in many lessons, possibly under different perspectives.

A second correlated issue is obtaining a quick list of the most important topics treated in a lesson and in the set of lessons that make up a course. Obviously, in many cases such a "short list" is contained in the institutional syllabus of the course, but unfortunately there is often a great diversity in the depth and accuracy of the content of official syllabi.

ITADATA2022: The 1st Italian Conference on Big Data and Data Science, September 20-21 2022, Milan, Italy
EMAIL: marco.arazzi01@universitadipavia.it (A. 1); marco.ferretti@unipv.it (A. 2); antonino.nocera@unipv.it (A. 3)
ORCID: 0000-0002-3371-307X (A. 1); 0000-0003-3543-2383 (A. 2); 0000-0003-2120-2341 (A. 3)


The combined availability of these two facilities, namely a “smart video index” and a “smart topics list” can really benefit students, offering them a tool for effective rehearsing learning material attached to lessons that they have attended possibly months earlier, or that they never attended at all.

To appreciate the problem in terms of data to be processed, one has first to collect reliable statistics on the dimension of the archived videos, and secondly to figure out the processing required on some experimental course, on which a-priori knowledge is easily available.

It is however quite evident that this endeavor can face tight limitations in terms of “privacy” and “intellectual property” issues, so that effective collaboration, both at the personal and at the institutional level is required. This is especially true if one wants to consistently apply this approach to the whole corpus of video lessons archived by the institution.

To set the ground, an inquiry on the institutional repository of the university used to store video lessons is severely limited in the detail that can be extracted. The archive is allocated on a contracted Web service, and even system administrators have very limited grants on the metadata of the stored objects. For a limited time span (6 months backwards), it is possible to get a list of file names, filtered by file type (in our case, .mp4 only), but not their dimension, not to mention ownership. Furthermore, the institutional repository can be queried for the overall dimension of the allocated storage of each account, but this is of no real use, because in this case no breakdown can be produced in terms of file type.

Notwithstanding these severe limitations, a fairly reliable estimate of the amount of data available can be obtained by considering that lessons recorded during the pandemic have not been deleted, on the contrary the university governors have requested professors to keep this material available online for one more academic year. So, the system look-up carried out in May 2022 for this paper show a total of 19,556 .mp4 files over six month, that are likely to be under-estimate of the online material actual available.

The work we are reporting here, therefore, has been developed as a test-bed prototype, by choosing a bachelor level course (taught by one of the authors), for which full access is available. The set of lessons have been used both for devising a video indexing procedure and for analyzing the feasibility of extracting “meaningful” topics for the whole course.

This paper is organized as follows: Section 2) briefly summarizes the most relevant scientific literature background. Section 3) covers the development of an indexing procedure which allows a student to look up in the corpus of lessons “elementary concepts” that are described by a few significant words, with a proper set of pointers into the videos, where such significant words are spoken.

The goal is a most practical one, in that the tool targets easy recollection of positions in the huge sequence of spoken utterances, with limited, if no insight into the semantics of such utterances within the single lesson of the course. Some lexical analysis is however mandatory, but the parts-of-speech used for this task can be possibly tailored to the knowledge domain.

Section 4) instead sets the foundations for the automatic construction of a course syllabus, but predominantly addresses the goal of identifying the most relevant topics using NLP approaches.

Section 5) discusses the dimension of the problem in light of the results obtained on the test bed course and envisions a scenario for selecting a big data approach for managing the whole corpus of online material available in the university repository.

2. Related work

Video indexing is a well-established field and, combined with speech analysis and recognition, has been used in many environments. Initial manual annotations have long been abandoned.

Many approaches have been proposed, that are best suited for content based on slides and speech [1], and that also try to leverage existing material with an OCR processing phase carried out on slides extracted from the video track [2]. Indeed, OCR is very developed and precise when applied to typed text; however, this technique is not available if lessons make relevant use of blackboards. This is why audio, though less accurate than OCR, has become a major means to track the most important and detailed info. On the negative sides, there may still be difficulties in recognizing some voices, and some

very specific terms can produce OOV (Out-Of-Vocabulary) words errors. The lack of a strong sentence structure, typical of written material, can also add to the difficulty.

Nevertheless, mature applications exist for voice analysis, whose main goal is to minimize WER (Word Error Rate), especially to avoid OOV. A limited, non-exclusive list includes Gaudi [3], PodCastle [4, 5], NTU Virtual Instructor [6] and MIT Lecture Browser [7]. A critical point for the purpose of this project is the availability of a good Italian language model, a feature not always embedded int these applications.

The actual indexing in a corpus of video/audio recording has been pursued in some special application domains, such as call centers [8] and news in broadcasting [9]. Many advanced features developed in these contexts, such as detection of change in speaker, do not add to the purpose of this research.

As for the second objective of this paper, i.e., the implementation of an indexing strategy based on topic extracted from the video content, several approaches in the related literature focus on the problem of topic modeling from different textual sources.

In particular, the approach of [10] identify the research topics that best describe the scope of a scientific publication. An application called Smart Topic Miner incorporates the solution, enabling editors of Springer Nature journals to annotate publications according to a set of topics drawn from a large ontology of Computer Science-related fields.

The authors of [11] try to answer the question "in what research topics were the academic community of Computers & Education interested?". Their approach consisted of bibliometrically analyzing a structural topic model (STM) to identify the topic hotspots of 3,963 articles published in Computers & Education between 1976 and 2018.

In the context of education, the approach proposed in [12] focused on the difficulty of students to make informed decisions using the content available through online reviews of Academic institutions. The authors of the paper present an ensemble of Latent Dirichlet Allocation (E-LDA) topic models for automatically identifying key features of student discussion and categorizing each review statement into the most relevant topic.

Similarly, the authors of [13] exploit a statistical algorithm (LDA) applied to the complete full-text corpus of one major journal of the field (Biology and Philosophy) to identify the key research topics that span across these 32 years (1986-2017).

Finally, the authors of [14] developed an algorithm that takes URL of a video from user as input and they implemented a summarization process with the help of two algorithms. The model also gives the flexibility to the user to decide what percentage of summary is needed compared to the original lecture. The summarization technique is a subjective process. Two prominent methods were incorporated into the model. One is cosine similarity and the other is ROUGE score.

Human-generated summaries are not needed in the former, whereas the latter requires it. Both TF-IDF and Gensim can obtain greater than 90% efficiency via Cosine similarity. When it comes to ROUGE scores, an efficiency of 40-50% can be obtained.

3. Smart Video Indexing

The test bed used to carry out the project is a bachelor course on data base technology (Introduction to Data Bases), consisting of 21 recorded lessons, each some 1h and 50 m in length. The course has been delivered in the Italian language, but this has little bearings on the developed methodology. The total space for the online material is 2,192 GB (the lessons have been recorded in low resolution), with an average dimension of 104,4 MB per lesson.

As anticipated, the first goal is to build an index that allows to search within the lessons a set of words, obtaining the time frames within each lesson where the words are spoken. As will be soon discussed, the "set of words" that turns out to be really meaningful has extremely low cardinality, and eventually only *binomial* (to be defined later) have been retained. To locate within any lesson binomials, and build the overall index with the list of time frames each binomial is spoken, we devised a procedure that leverages existing Web services and that applies specific transformations, including lexical analysis

and merging, to get the desired final output. Our approach is very simple, because it tries to get the useful information (timestamp) with the least possible effort and simplest tools, to allow for a scale-out approach.

3.1. Indexing Procedure

The indexing procedure consists of the following steps, carried out in each lesson:

1. Speech-to-text conversion, delivering "tokens" and associated timestamp annotation
2. Token lexical analysis and POS (part-of-speech) tagging
3. Tagged token grouping by time proximity, forming tentative "*binomials*" and "*trinomials*"
4. Binomials and trinomials filtering by selecting couples (triples) of significant POS tags
5. Lemmatization of selected binomials (trinomials)

Once this procedure is carried out on each lesson, it is possible to merge the outcomes at the lesson level into a single course-level list of tagged and timestamped binomials (trinomials), which is the basis of the "index". Each index entry actually contains a list of occurrences of the binomial (trinomial) within each lesson and at all timestamps in each lesson. In what follows we give briefly details of each step and keep the description at a fairly high level of abstraction, namely, we disregard practical implementation details, such as underlining DBMS and storage, which are really straightforward, at least when addressing a single course.

3.1.1. Speech-to-text and timestamping

Speech-to-text is a very well-established technology, and the choice is whether to deploy existing packages, or resorting to online services, a classical "buy" or "pay per use" alternative. The main constraint for our case is that the timestamp of each token/word is attached to the transcript. Punctuation is a second feature, that may be a benefit if subsequent text analysis is carried out, as is our case for the "topic" identification second goal of this project. Finally, a reliable Italian model is also necessary.

At the time of writing this paper, several research and industrial solutions of speech-to-text exit. The specific strategy to adopt is orthogonal to our proposal; indeed, any existing approach could be applied in our case. To keep our solution scalable, we leveraged industrial strategies based on Cloud Computing. We exploited the solutions provided by YouTube, by the Google Cloud Platform (GCP) [15] and the one available on the Amazon Web Services (AWS), namely AWS Transcribe [16]. Being this a prototype work, we selected to go for these online "pay per use" services, also leveraging free offers for limited workloads. We performed a WER analysis on the transcripts of the generated texts of three lessons of the course, with all platforms delivering acceptable results, YouTube being slightly more precise, and Google and AWS service providing punctuation.

3.1.2. Lexical analysis and POS tagging

To come up with a "meaningful" search, it is obviously necessary to prune the generated list of tokens/timestamps, and this calls for lexical analysis and POS tagging (Part-of-Speech). This is where the availability of language specific tools and services plays its relevant role. In this experiment, we considered the NLTK (Natural Language Toolkit) Python library [17] and, in particular, its pos-tag module. NLTK provides native support for the English language; however, other libraries could be exploited to support other languages, such as TINT [18] for the Italian language.

On the outcome of the speech-to-text, we set-up an "interesting words list" procedure, namely we selected sequences of time contiguous words. Since the ultimate goal of this part of the project is to offer students a facility to look up for spoken short sentences, we limited the list of time adjacent words to two/three words.

The knowledge domain associated with the testbed course suggests pruning the "two/three word list" with POS filtering that privileges nouns (N) and adjectives (A). Inclusions of verbs seem to offer nonactual advantage. Should this procedure be applied to courses of other knowledge domain

(literature, philosophy, and so forth) a different choice could be meaningful. So, we chose to stay with pairs of items in the form {N,N}, {N,A} and {A,N}, *binomial* in the following. Lemmatization is the final part of this procedure.

Resulting binomials retain the timestamp of the first spoken part, and are collected lesson by lesson, to be later merged for generating the actual course index. Section 5 reports quantitative outcomes.

4. Topics extraction

This section deals with the second objective of the research proposed in this paper, namely: the identification of topics discussed in the video material to enable an advance indexing mechanism considering also the concepts included in a course.

This research objective requires different advanced NLP tasks to be carried out. In the next subsection we will provide a preliminary description of the different strategies adopted to build the complete solution.

4.1. Procedure

Our solution leverages Natural Language Processing strategies specifically designed for textual data. Therefore, starting from the whole repository of a course video lessons, a preliminary step is the application of a *speech-to-text* activity in such a way as to obtain a conversion of the audio content of videos into text documents. To do so, we applied the same solution described in Section 2 exploiting Cloud Computing.

Once the transcribed text is available, our solution proceeds by performing a pre-processing step designed to prepare the input for the subsequent topic modeling task.

We adopted a POS (Part-of-Speech) tagging approach to label each word contained in the textual input with the corresponding grammatical role (e.g., verb, noun, adjective, article, and so forth) by leveraging, once again, the same approach described in Section 2.

Of course, typically, in addition to concepts directly related to a specific topic, a lesson will include parts that are either derived by interactions with students (such, question and answering) or related to examples useful to improve students' understanding of a concept.

Although these parts could be interesting and deserve further investigation, in our preliminary solution we focus only on the general concepts because our objective is to build a solution, based on topic modeling, for enhancing the indexing strategy for video lessons.

To filter out such content from our analysis, we adopted a supervised approach based on the use of advanced word embedding along with a dataset of real syllabi.

At this point, leveraging the pre-processed dataset described above, we trained a Glove model [3] to obtain consistent embedding of words according to the text which they are involved in.

From the syllabi in the training dataset, we extracted the *important* terms, using the same POS-tagging strategy described above, and selected the k-most similar words according to their representation obtained by the Glove model (preliminary we set k=20) from our input textual data.

This step is required to filter-out noisy words from the running text possibly referring to the additional parts mentioned above (e.g., examples, interactions with students, and so forth), which are typically not included in course summaries and syllabi.

Figure 1shows the steps performed to filter-out noisy words from the running text.

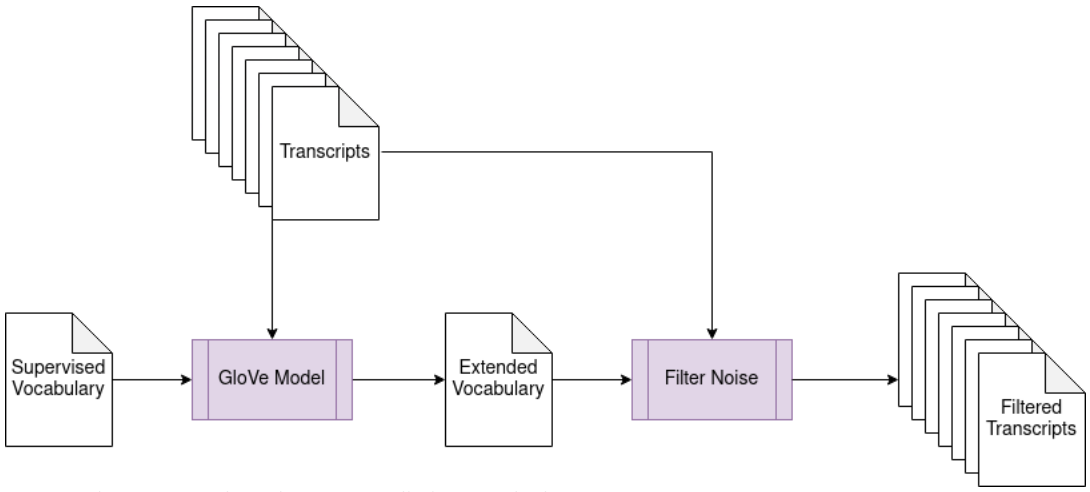

Figure 1: The strategy adopted to create a dictionary of relevant terms

In this way, we obtained a rich dictionary of terms that are inherent to what is "typically" included in a manually written syllabi.

By leveraging the knowledge represented in this dictionary, it was possible to create a *filter* to remove all the portions of text from the lesson recordings that were not related to the concepts "described" in the dictionary (and, hence, in the reference training syllabi).

As a next step, we proceeded by performing a topic extraction task using the BERTopic approach [4]. This approach consists in the following steps:

1. Organize the text in paragraph.
2. Create sentence embeddings using SBERT [5].
3. Dimensionality reduction using UMAP [6].
4. Clustering using HDBSCAN [7].
5. Keyword extraction using a variation of TF/IDF.

As for the first point, we organized our input text corpus into paragraph by adopting a window-based strategy. We imposed a limit to the number of words, say w, of each paragraph and, hence, we organized our text into portions of w words (in our preliminary experiments we set $w=25$).

After that, following the strategy of [4], we applied the SBERT algorithm to the paragraphs obtained in the previous step and we obtained sentence-level embeddings.

Due to the high dimensionality of embeddings, to be able to cluster sentences together, thus obtaining clusters of concepts, we applied a dimensionality reduction strategy. In particular, from the scientific literature, UMAP has been proved to be particularly adequate in contexts similar to ours, therefore, again as done in [4], we selected it as dimensionality reduction algorithm.

At this point, any clustering strategy could be exploited to group together sentences using their reduced embeddings. By leveraging, once again, the recent results in this context reported in the scientific literature, HDBSCAN has been proved to achieve very high performance when applied to cluster text embeddings after the application of a UMAP-based dimensionality reduction.

Finally, we used the modified version of the TF/IDF algorithm, described in [4], to select the most relevant keywords for each cluster (i.e., the most relevant words in the sentences of a cluster).
Figure 2 shows a graphical representation of the steps described above to obtain topics along with their keywords.

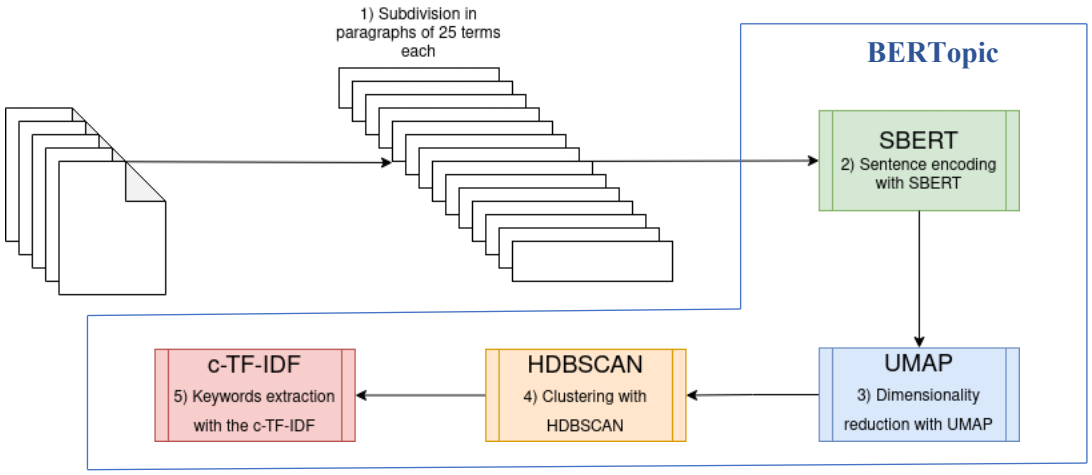

Figure 2: The adopted flow for topic modeling

5. The Big Picture

In what follows, we present the quantitative assessment resulting from processing the 21 "prof-of-concept" lessons of the course "Basi di Dati". Table 1 lists the main measures; other values, such as length of pauses, duration of the speech and other similar quantities are not interesting for the purpose of this project, although they carry other interesting information.

Table 1
Main measures on the testbed 21 lessons

Lesson #	.mp4 (MB)	tokens	tokens (KB)	binomials	words
1	92,4	10491	229	2966	1117
2	108,5	11676	255	3150	991
3	102,9	9555	198	2538	803
4	105,4	9646	212	2624	840
5	114,3	8349	181	2210	703
6	128,7	9281	201	2559	824
7	123,2	9873	211	2613	874
8	72,2	7530	163	2191	828
9	94,2	9290	201	2435	901
10	100	10830	234	2956	1032
11	85,7	10656	233	2945	1063
12	89,5	9796	215	2722	931
13	73,1	6569	144	1641	591
14	124,2	11553	252	2994	1052
15	104,1	11043	241	2971	960
16	96,9	10251	224	2768	882
17	118,3	9541	207	2442	797
18	112,4	8143	179	2097	761
19	113,2	9520	206	2469	776

20	97,6	10927	238	2961	1050
21	135,9	10630	230	2860	902
totals	2192,7	205150,0	4454,0	55112,0	18678,0
avg	104,4	9769,0	212,1	2624,4	889,4

After processing each lesson, getting the tokens, and applying the pruning procedure, one obtains the required binomials. The following phase merges binomials, effectively building the binomials index at the course level.

The aggregated set of binomials account for some 18,920 entries. A snapshot is available in Table 2, which list the first 10 entries (the English wording has been added for the general reader). The table is a logical representation of the index, as already anticipated, since actual DBMS entries depend on the logical model of the DBMS and might be mapped to quite different physical constructs.

Table 2

Binomial entries (course level), first 10 of 18920. English translation added.

Binomial	in lessons	tot count	# lessons
chiave esterna (foreign key)	2 3 4 5 6 8 9 10 17 20	99	10
dipendenze funzionali (functional dependencies)	17 18 19 21	97	4
chiave primaria (primary key)	2 3 4 6 8 9 15 17 18 19 20 21	92	12
basi dati (database)	1 2 4 6 10 11 12 14 15 16 18 20	63	12
modello relazionale (relational model)	1 2 3 4 6 7 8 12 14 15 16 17 18 21	61	14
target list	3 5 6 7 8 9 10 20 21	60	9
vincoli integrità (integrity constraint)	2 3 4 6 8 9 10 12 17 18 21	59	11
punto vista (view point)	1 2 3 5 6 7 8 9 10 11 12 13 14 15 17 18 19 20 21	51	19
legame associativo (associative link)	12 13 14 15 16 17	42	6
codice fiscale (personal ID)	2 4 6 12 13 15 17 18 19 21	41	10

While the topmost entries show a good representation of the most common bi-words spoken, that match very well the knowledge domain of the testbed course, some spurious entries also appear (such as “punto vista”, “codice fiscale”). The wording attitudes of the speaker, as well as recurrent concepts (personal ID as an instance of primary key in many examples used in the lessons) make the index somewhat cluttered.

At this stage it is possible to outline the scenario, should the proof-of-concept be brought into production at the institutional level, that is, should all the online material be analyzed and transformed accordingly.

While it is not possible to have a precise set of figures, since the distribution of file dimension is not known, one can assume that each of the 19,566 files available at the institutional level represents a “lesson” and brings about a number of “tokens” close to the average extracted from the proof-of-concept testbed.

This amounts to producing 2x10^8 tokens in some 2x10^4 x 0,2MB=4x10^3MB files to be processed. These figures, while not in the order of the TB, start requiring a big-data approach, if one

wants to prepare an architecture that is capable of ingesting online material on a semester basis, and producing reliable indexing in almost "real-time", that is while the lessons are being delivered by professors.

This consideration becomes even more relevant if the topic extraction strategy to improve smart indexing of video lessons is also considered.

In fact, the extraction of topic requires the elaboration of tokens on a sentence level and the execution of the whole flows depicted in Figure 1and Figure 2 and described in Section 4.

Therefore, every time a new lesson is added to the repository, overall, the approach will build a dictionary of "relevant" terms by re-training the Glove model.

An example of the obtained dictionary for the 21 "proof-of-concept" lessons of the course "Basi di Dati" is reported in Table 3. This table reports the 10 most common terms of the whole dictionary containing 700 terms.

Table 3

Top frequent terms of the obtained dictionary for the proof-of-concept

Term	Frequency
relazione (relationship)	879
attributo (attribute)	866
chiave (key)	689
schema	649
entità (entity)	583
concetto (concept)	489
proiezione (projection)	375
vincolo (constraint)	359
esterno (foreign)	343
dbms	307

After that, it exploits the flow of Figure 2 to extract topics and to identify the keywords that better represent the concept encoded by each of them.

As an example, the list of topics along with their keywords for the "proof-of-concept" is reported in Table 4.

Table 4

The topics and keywords extracted by our approach for the proof-of-concept

Topic ID	Keywords
Topic 0	{dipendenza (dependency), funzionale (functional), dipendenza funzionale (functional dependency), forma (form), determinante (determinant), relazione (relationship), forma normale (normal form)}
Topic 1	{proiezione (projection), espressione (expression, attributo (attribute), sigma, restrizione (constraint), predicare (predicate), relazione (relationship}
Topic 2	{query, tabella (table), operatore (operator), query query, cartesiano (cartesian), prodotto (product), algebra}
Topic 3	{dbms, applicazione (application), ambiente (environment), cloud, connessione (connection),

	sistema (system), base dato (data base), rete (network)}
Topic 4	{chiave (key), primario (primary), chiave primario (primary key), chiave esterno (foreign key), chiave chiave (key key), vincolo vincolo (constraint constraint), relazione chiave (key relationship)}
Topic 5	{tabella (table), dominio (domain), lista (list), table (table), tabella tabella (table table), esempio (example), dato (datum)}
Topic 6	{entità (entity), concetto (concept), associazione (association), identificatore (identifier), associativo (associative), associazione logica (logic association, logica (logic)}
Topic 7	{modello (model), concettuale (conceptual), progettazione (design), modello relazionare (relational model), schema, fase (phase), logico (logic)}

By inspecting this table, we can, for instance, see that the first identified topic, namely *Topic 0*, contains the keywords {dipendenza (dependency), funzionale (functional), dipendenza funzionale (functional dependency), forma (form), determinante (determinant), relazione (relation), forma normale (normal form)}. By inspecting the set of keywords, for which we also included the English translation in the parentheses, it is possible to associate with this topic the concept of *functional dependency*.

Our strategy can, hence, associate each paragraph included in the content of the video lesson to one of the obtained topics, thus providing an enhanced semantic indexing.

6. Discussion and Conclusion

This paper described a preliminary attempt to build an intelligent system to support the fruition of the huge amount of video data produced during the Covid-19 pandemic.

In particular, we focused on both the definition of a smart indexing mechanism and an approach to extract the main subjects discussed during the lessons of a course. The experimental campaign has been carried out by leveraging the material produced during the "Basi di Dati" course taught by one of the authors. Although the initial objective was to develop the solution to provide immediate support to the students, due to technical and bureaucratic aspects we were not yet able to deploy such support system and make it available to our students at the moment of writing this manuscript.

As a future research direction, we are hence planning to complete the deployment of our strategy and to analyze usage data to study the impact of such a support system to improve the fruition quality and usability of online material.

Also, because we could not deploy our solution in a real live system, the study presented in this paper refers to a limited case of just 21 videos. Of course, due to this limitation, we could not investigate the performance of our solution when applied to a real big data scenario. However, we argue that the application of such a solution to support all the courses taught in a single University would alone require an ad-hoc big-data strategy to allow for scalability.

On another side, our approach has been designed so that, once trained on an initial dataset, it can be used to support also any new course, provided it refers to concepts included in the initial training.

This would allow the easy extension of our solution to improve the fruition of video material also in disparate contexts. Indeed, the knowledge base obtained by training it on data derived from the University domain could be used to build enhanced smart indexing facilities also for streaming context such as online conferences, webinars, video streaming, and so forth.

The application of our approach to these additional scenarios would require the access to its functionalities in pseudo real-time during the live events.

Therefore, we are planning to study suitable big data architectures to complete the preliminary solution described in this paper and further develop and refine it according to the research directions described above.

7. Acknowledgements

The authors wish to thank Marco Prina for contributing to some preliminary explorative analyses related to the study described in this paper.

8. References

[1] H. Yang and C. Meinel, "Content based lecture video retrieval using speech and video text information," *IEEE transactions on learning technologies,* vol. 2, no. 7, pp. 142-154, 241r.

[2] N. Van Nguyen, M. Coustaty and J. M. Ogier, "Multi-modal and cross- modal for lecture videos retrieval," in *2014 22nd International Conference on Pattern Recognition*, 2014.

[3] C. Alberti, M. Bacchiani, A. Bezman, C. Chelba, A. Drofa and H. Liao, "An audio indexing system for election video material," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[4] J. Ogata and M. Goto, "PodCastle: A spoken document retrieval system for podcasts and its performance improvement by anonymous user contributions," in *Proceedings of the third workshop on Searching spontaneous conversational speech*, 2009.

[5] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks.," *arXiv preprint arXiv:1908.10084,* 2019.

[6] L. L. S. Kong, M. Wu, C. Lin, Y. Fu, Y. Chung, Y. Huang and Y. Chen, "NTU Virtual Instructor - A Spoken Language System Offering Services of Learning on Demand Using Video/Audio/Slides of Course Lectures," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009.

[7] C. Chelba, T. J. Hazen and M. Saraclar, "Retrieval and browsing of spoken content," *IEEE Signal Processing Magazine,* vol. 25, no. 3, pp. 39 - 49, 2008.

[8] M. Garnier-Rizet, G. Adda, F. Cailliau, S. Guillemin-Lanne, C. Waast-Richard, L. Lamel and S. Vanni, "CallSurf: Automatic Transcription, Indexing and Structuration of Call Center Conversational Speech for Knowledge Extraction and Query by Content," in *Proceedings of the International Conference on Language Resources and Evaluation*, Marrakech, Morocco, 2008.

[9] J. Makhoul, F. Kubala, T. Leek, D. Liu, L. Nguyen, R. Schwartz and A. Srivastava, "Speech and language technologies for audio indexing and retrieval," *Proceedings of the IEEE,* vol. 88, no. 8, pp. 1338-1353, 2000.

[10] A. Salatino, F. Osborne, A. Birukou and E. Motta, "Improving editorial workflow and metadata quality at springer nature," in *International Semantic Web Conference*, 2019.

[11] C. Xieling, D. Zou, G. Cheng and H. Xie, "Detecting latent topics and trends in educational technologies over four decades using structural topic modeling: A retrospective of all volumes of Computers & Education," *Computers & Education,* vol. 151, p. 103855, 2020.

[12] S. Srinivas and S. Rajendran, "Topic-based knowledge mining of online student reviews for strategic planning in universities," *Computers \& Industrial Engineering,* vol. 128, pp. 974--984, 2019.

[13] C. Malaterre, D. Pulizzotto and F. Lareau, "Revisiting three decades of Biology and Philosophy: A computational topic-modeling perspective," *Biology \& Philosophy,* vol. 35, no. 1, pp. 1--25, 2020.

[14] K. Kulkarni and R. Padaki, "Video Based Transcript Summarizer for Online Courses using Natural Language Processing," in *2021 IEEE International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, 2021.

[15] Google Cloud Platform, "Cloud Speech-to-Text," 13 June 2022. [Online]. Available: https://cloud.google.com/speech-to-text.

[16] Amazon Web Services, "Amazon Transcribe," 13 June 2022. [Online]. Available: https://aws.amazon.com/it/transcribe/.

[17] N. Team, "Natural Language Toolkit," 13 June 2022. [Online]. Available: https://www.nltk.org/.

[18] Fondazione Bruno Kessler, "TINT – THE ITALIAN NLP TOOL," 13 June 2022. [Online]. Available: https://dh.fbk.eu/research/tint/.

[19] G. Maarten, "BERTopic: Neural topic modeling with a class-based TF-IDF procedure.," *arXiv preprint arXiv:2203.05794,* 2022.

[20] J. H. a. S. A. L. McInnes, "hdbscan: Hierarchical density based clustering.," *J. Open Source Softw.,* vol. 2(11), 2017.

[21] L. McInnes, J. Healy and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction.," *arXiv preprint arXiv:1802.03426,* 2018.

[22] J. Pennington, R. Socher and C. D. Manning, "Glove: Global vectors for word representation.," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.