Web Crawler for Indexing Video e-Learning Resources: A YouTube Case Study

Bogdan IANCU
The Bucharest University of Economic Studies, Romania
bogdan.iancu@ie.ase.ro

The main objective of the current paper is to develop and validate an algorithm focused on automatically indexing YouTube e-learning resources about a certain domain of interest. After identifying the keywords specific to the desired domain, a web crawler is developed for evaluating video resources (from the YouTube platform) in terms of relevance for that domain. Once the most relevant video resources are found, they are indexed with the usage of a NER engine applied on their transcripts. In this manner, semantic queries can be used further in order to find exactly the needed information inside these multimedia resources. The crawler will repeat the indexing process daily in order to maintain the repository of semantically indexed videos up to date. The final chapter presents the obtained results together with the validation of the model.

Keywords: *crawler, YouTube, NER, semantic web, e-learning*

1 Introduction

YouTube is arguably one of the largest e-learning resource of humankind. With 300 hours of video content uploaded to the platform every minute [1], no one knows how deep the information about a certain topic can be buried between a wide range of meaningless videos. To make things even worse, most of the information present on the platform is available in video format only, without captions or any other form of searchable content.
The problem that raises from here is how can one search and watch interesting videos about a certain topic only, without being flooded with unwanted or distracting videos. Having in consideration that the platform's declared main scope is to increase the watch time [2], it is uncertain when and how will Google introduce a feature into their platform in the near future that will resolve the problem.
One solution could be the usage of the existing search feature of the platform and the addition of one or more filtering layers that can extract and rank only the desired videos for a specific topic.
In order to do this, this paper proposes a keyword-based search approach for finding the top most interesting videos for each and every key term related to the desired topic. Then, for each video, the captions are extracted (if they are present) or generated (as show in [3]). The DBpedia Spotlight algorithm [4] is used further for extracting the entities from the text and the results are saved into a Microsoft Azure Cosmos DB database. This cloud database was chosen because it is ranked as the second fastest graph database and the optimal option for saving large semantic data [5].
The indexing process presented above is repeated daily, with the search for newly added videos and their semantic indexing. The results of the proposed algorithm are presented either in a classic format as in the YouTube platform, either in a graph format which highlights the relationships between videos and the grade of their relevance to the searched terms or entities.

2 Related Work

A similar approach for e-learning video resources indexing was conducted by the author in his PhD thesis [6]. However, the current work differs by certain stages of data processing from the previous work. Paper [6] had used a manual data input scheme and the developed platform needed an administrator in charge with adding new e-learning resources. The current approach uses a keyword-based web crawler for automatically adding new data into the platform in order to be processed. Another distinction is given by the multi-lan-

DOI: 10.12948/issn14531305/23.2.2019.02

guage support of the current algorithm. Because the captions are generated, if they are not present on the YouTube platform, they can be easily translated into English in order to perform the entity recognition. After this step, as long as the users search for entities that have correspondents in English, the language of the video doesn't matter anymore from the algorithm's point of view. This was a limitation of the previous work, resolved in this version of the algorithm. A short literature review is presented below.

Paper [7] proposes a framework for video semantic recognition with the usage of supervised and semi-supervised machine learning models. However, the paper is focused rather on feature extraction, based on the visual component of the multimedia content, than on the cognitive content per se.

Even though in [8] the authors are using an Web Ontology Language (OWL)-based ontology for semantic content analysis, the main focus remains on feature extraction, not on knowledge extraction.

A research closer to the current work is [9] where unsupervised machine learning algorithms are used to classify different videos from the Dailymotion website coming from 9 different channels with the usage of Named Entity Recognition (NER). Nevertheless, the final scope is to improve the classification of the videos rather than to extract and further use the content.

The closest research paper to the this one is [10] where Natural Language Processing (NLP) techniques are used to extract certain entities from YouTube videos. The main disadvantage of the model presented in [10] is that it can be used by experienced users only via the SPARQL Protocol and RDF Query Language (SPARQL) endpoint or by connecting it to the Linked Open Data (LOD) cloud.

To summarize, the added value of the current paper consists in the overall data flow between the different components, that allows the automatic indexing of the multimedia e-learning resources from the YouTube platform, the final scope being further access to the indexed resources by non-technical users.

3 The web crawler

The first step of the developed algorithm is the automatic crawling of the newly added YouTube videos for a specific domain of interest. For the scope of this paper, entrepreneurship was chosen as the main topic. The YouTube Data API [11] is used to perform search queries based on relevant keywords. The keywords are extracted from the DBpedia ontology [12] based on The Dublin Core Metadata Initiative [13] component, that the ontology links to.

Let us take a more specific case from the previously mentioned entrepreneurship domain. If we focus on the *dct:subject* property of the *http://dbpedia.org/resource/Entrepreneurship* entity, we will find the following entities as values (their English labels in fact): "Entrepreneurship", "Business models", "Business occupations", "Business terms", "Management occupations", and "Small business". By using the relations available in the LOD cloud further we can identify for each and every entity that was found in this step if it is subject of another DBpedia entity by using the inverse property of *dct:subject*. For example, for "Business occupations" we can find the following entities (enumerated by their English labels again): "Consultant", "Decision analyst", "Board of directors", "Chief Scientific Officer", "Chief operating officer", "Chief privacy officer", "Enrolled agent", and the list can continue. For the topic of entrepreneurship, by following these steps, a list of 732 keywords were extracted with the usage of SPARQL queries.

The next step of the algorithm is to feed the identified keywords to the web crawler which will call the YouTube's Data API to obtain the first 50 results (the maximum number of results in the current version of the YouTube Data API) for every keyword. The API has options for result filtering based on a certain geographic locations or specific languages, if this is the desired behavior.

After the video unique identifiers are obtained for all the extracted keywords, the next phase begins. In this stage, the algorithm requests the English captions for the videos. The videos that have English captions are processed

DOI: 10.12948/issn14531305/23.2.2019.02

first because this means just an additional YouTube API call for getting them. For the rest of the videos, the Google Speech-to-Text API is used to obtain an approximate transcription of the video content. If the language of the video is not English, the captions will be translated to English with the help of the Google Translate API. The main reason for doing this is that the DBpedia Spotlight NER algorithm provides the best results for the English language [14]. All the obtained data is saved into a Microsoft Azure Cosmos DB cloud database as mentioned before. Figure 1 shows the overall schema for the current component of the developed algorithm.

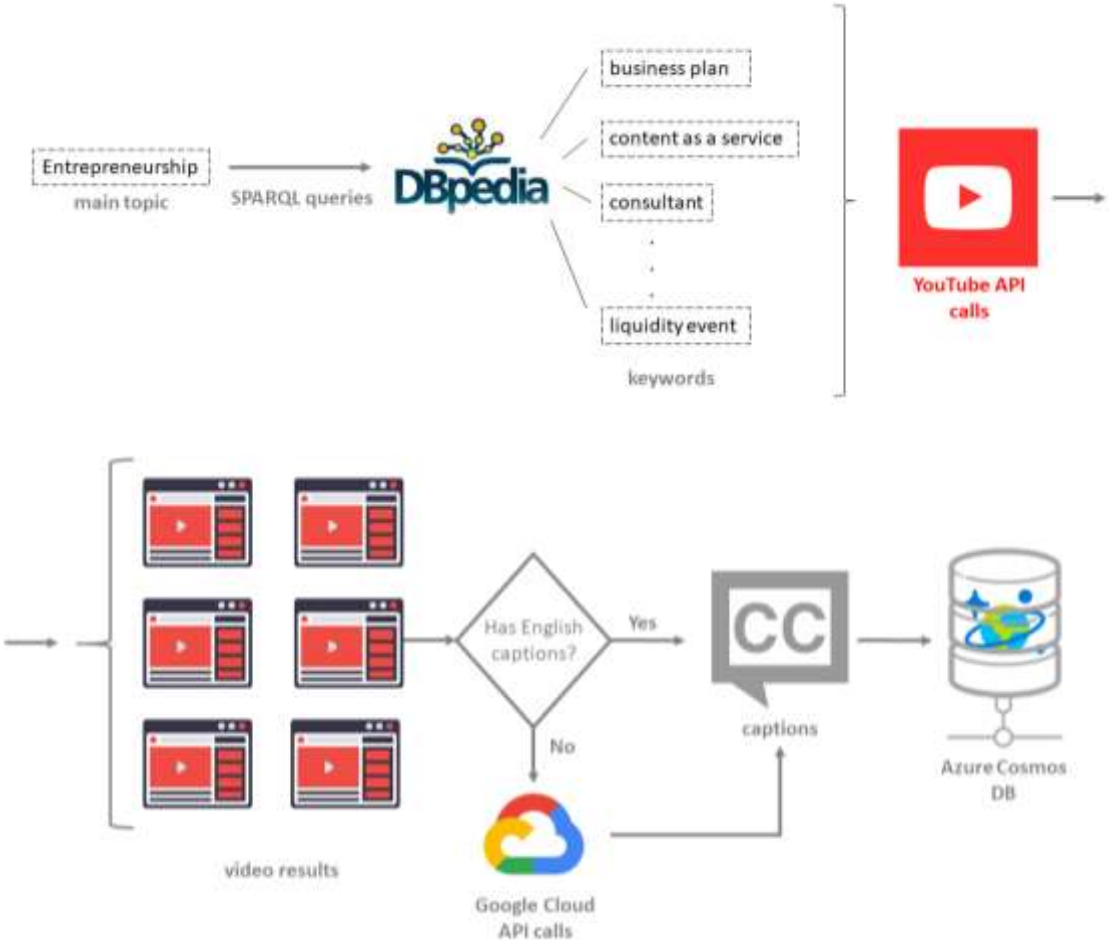

Fig. 1 The crawling component schema of the developed algorithm

4 Extracting the knowledge

In the previous phase, the videos relevant for the specific domain, along with their captions, were saved in the Cosmos DB instance. The next step is the extraction of named entities from the captions of each video. In order to achieve this goal, the DBpedia Spotlight NER engine is used. The best results, when it comes to under-resourced languages (the case of Romanian for example), were obtained for English texts when tuning the confidence to 30% [14]. After performing the NER process for the captions of each video, the database containing the video list will be updated with the list of entities from the DBpedia ontology found in each e-learning resource. The entire process described until this point is repeated daily in order to search and index new videos added on the YouTube platform. The already indexed videos are ignored. The YouTube unique identifier is used in order to detect if a video is already indexed or not.

After the first videos are indexed, the users of the platform are able to perform searches based on their needs. The searching process is

DOI: 10.12948/issn14531305/23.2.2019.02

not a keyword-based classical one. Rather than applying this method, the algorithm applies NER again on the search string in order to identify entities that the user is interested in. This time the n-best candidates version of the DBpedia Spotlight algorithm was chosen with the confidence set again to 30%. If the language of the platform is not English, then the search string is translated into English, the NER is performed and the translation of the identified entities' labels are displayed. Figure 2 shows the search page prototype of the designed platform. In this page the user can type the search string and entities are identified on the go. For every new entity found, a box with a random generated color is used to emphasize it. When the user hovers the text from the box, the description of the entity is displayed. For the entities that have more than one candidate, the box is replaced by a combo box with all the possible candidates ordered by their probability.

Fig. 2 The prototype of the search page with the details of the "Business plan" entity displayed

Once the entities that the user is interested in are found, the searching process starts. The searching algorithm uses these entities along with the relationships between them from the LOD cloud. In the first stage, the algorithm computes two ranks for each video from the database by taking into consideration the entities specified by the user. Below, equations (1) and (2) describe how these ranks are computed [6].

(1) $R_i = \sum n_{i,j}$, where R_i is the rank of the i video, and $n_{i,j}$ represents the number of occurrences of the j entity in the i video resource

(2) $R_i^{max} = \max(n_{i,j})$, where R_i^{max} is the maximum rank of the video i, and $n_{i,j}$ represents the number of occurrences of the j entity in the i video resource

In the second stage, the searching algorithm determines the DBpedia classes that correspond to each and every of the searched entities. A SPARQL query is used to interrogate the DBpedia ontology for the parent classes of every entity:

```
PREFIX rdfs:
<http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbpedia: <http://dbpedia.org/re-
source/>
SELECT ?class
WHERE
{
dbpedia:<entity_i> rdf:type ?class.
?class rdf:type owl:Class .
FILTER(?class NOT IN (owl:Thing))
}
```

Afterwards, the videos that contain entities which share the same parent classes with the identified ones are ranked also. Both ranks are

DOI: 10.12948/issn14531305/23.2.2019.02

recomputed, so the equations (1) and (2) presented above, become (3) and (4) after this phase [6].

(3) $R_i = \sum n_{i,j} + \sum q \cdot m_{i,k}$, where R_i is the rank of the *i* video, and $n_{i,j}$ represents the number of occurrences of the *j* entity in the *i* video resource, $m_{i,k}$ represents the number of occurrences of instances of the *k* class in the *i* video, and q is a configurable significance coefficient (set to 0.2 in our case)

(4) $R_i^{max} = \max(\max(n_{i,j}), \max(q \cdot m_{i,k}))$, where R_i^{max} is the maximum rank of the video *i*, and $n_{i,j}$ represents the number of occurrences of the *j* entity in the *i* video resource, $m_{i,k}$ represents the number of occurrences of instances of the *k* class in the *i* video, and q is a configurable significance coefficient (set to 0.2 in our case)

The rank R_i is used further for sorting the results of the search string and the R_i^{max} to determine the dominant entity for each e-learning video. The dominant entity will be graphically marked in the platform by the use of a specific colour (similar to what is visible in Figure 2). Figure 4 presents the entire logical schema diagram of the searching algorithm.

Because the algorithm needs a large amount of processing power, just grade one parent classes are taken into consideration. This step is necessary because some of the videos might not contain exactly the identified resource. For example, if one of the searched entities is *"Funding"*, we might have the situation when no videos contain this entity. By applying the second phase of the algorithm, the parent class of *"Funding"* is queried, which in this case is *"Band"*. In this way we can find videos that contain entities such as "Lobbying", "Society", "Bestinvest" and other similar terms that might be partially relevant for the person interested in funding.

The results are presented by the platform in two manners: a classical one and a graph-based one which illustrates better the relationships between different video resources. Figure 3 and 5 show the prototypes of the two versions of results displaying page.

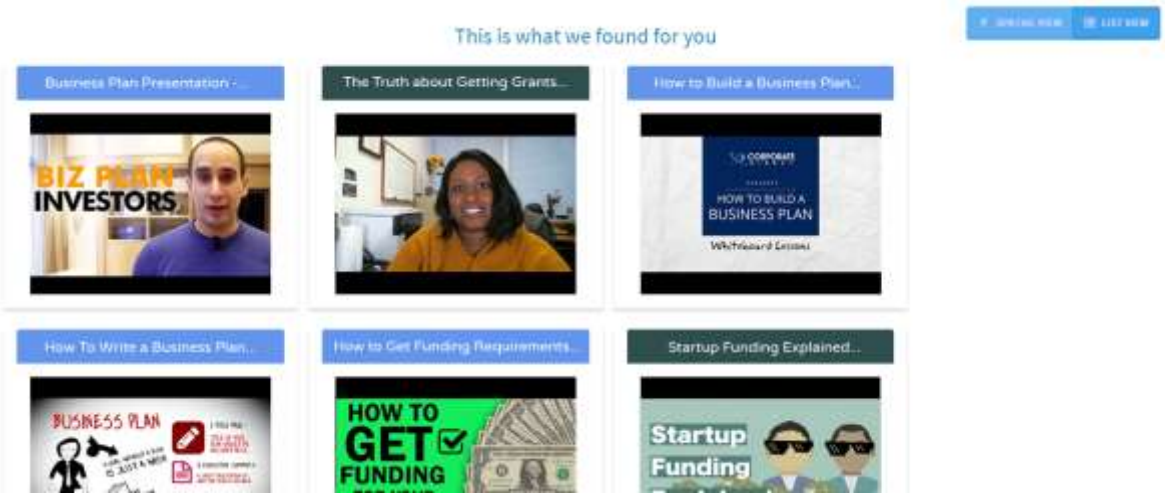

Fig. 3 The prototype of the classical display page. The blue videos are more relevant to "business plan" and the teal ones to "funding"

DOI: 10.12948/issn14531305/23.2.2019.02

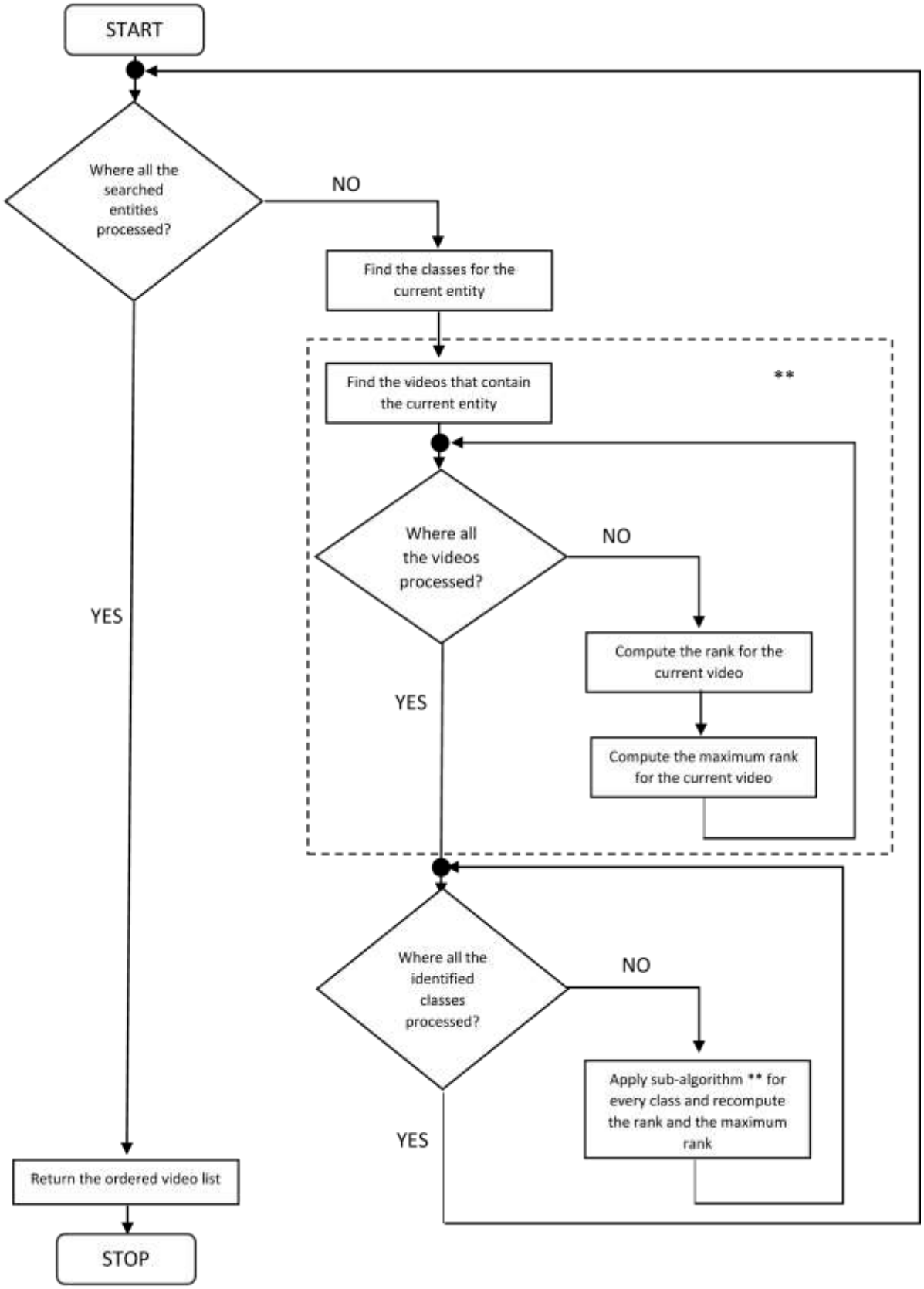

Fig. 4 Logical schema diagram of the e-learning videos searching algorithm

DOI: 10.12948/issn14531305/23.2.2019.02

Fig. 5 The prototype of the graph-based display page. The size of the circle illustrates how relevant is the video to the search query, the colour illustrates the dominant entity and the lines how videos relate to each other

5 Conclusions, validation and future work

The current paper presented the development of a semantic search algorithm applied on YouTube's video e-learning resources. The algorithm is composed of six consecutive processing stages: a) keyword extraction from the DBpedia ontology based on the selected domain of interest; b) YouTube Data API calls in order to obtain the most relevant videos for the identified keywords; c) caption extraction from YouTube or caption generation based on speech-to-text algorithms; d) entities extraction with the usage on the DBpedia Spotlight NER algorithm; e) video indexing and metadata saving into a Cosmos DB database; f) searching based on the identified entities and their relationships with the ones present in the search string.

In order to validate the efficiency of the developed algorithm, compared to YouTube's algorithm, which is based on keyword search and focused on maximizing the watch time, a survey approach was taken.

A sample of 105 persons were chosen among the second and third year students of The Faculty of Cybernetics, Statistics and Economic Informatics, all of them early users of the developed prototype. The chosen methodology was the one of random sampling without replacement with a 95% confidence interval.

The results of the survey were processed with the usage of the SPSS software [15]. The respondents were split into 64,8% third year students and 35,2% second year students, being divided approximately equally per gender.

The first section of the survey was focused on the time spent on the internet, the knowledge of English and the usage of search engines. The questioned students are good English speakers and are using Google and YouTube for searches related to educational resources (98.1% of them are using mainly these two search engines). Even though, 60% of them said that they do not find always the needed information when they use these platforms and 70.5% of them prefer the video e-learning resources over text ones.

When it comes to the developed prototype, the respondents said that they would evaluate the efficiency of the algorithm to 87.7%, compared to YouTube's one of just 76.5%. Table 1 and Table 2 present the descriptive statistics for the two questions in greater detail. Additionally, the students evaluated the efficiency of the graph-based representation of the results (depicted in figure 5) compared to the classical one (depicted in figure 3) to be

DOI: 10.12948/issn14531305/23.2.2019.02

81.9% better.

Table 1. Descriptive statistics for the question related to the developed algorithm's efficiency

			Statistic	Std. Error
On a scale from 1 to 10, how would you rate the efficiency of the search algorithm that you just used?	Mean		8,77	,117
	95% Confidence Interval for Mean	Lower Bound	8,54	
		Upper Bound	9,01	
	5% Trimmed Mean		8,83	
	Median		9,00	
	Variance		1,323	
	Std. Deviation		1,150	
	Minimum		6	
	Maximum		10	
	Range		4	
	Interquartile Range		2	
	Skewness		-,506	,245
	Kurtosis		-,874	,485

Table 2. Descriptive statistics for the question related to the YouTube's search algorithm efficiency

			Statistic	Std. Error
On a scale from 1 to 10, how would you rate the efficiency of the YouTube search algorithm?	Mean		7,65	,137
	95% Confidence Interval for Mean	Lower Bound	7,38	
		Upper Bound	7,92	
	5% Trimmed Mean		7,67	
	Median		8,00	
	Variance		1,889	
	Std. Deviation		1,374	
	Minimum		5	
	Maximum		10	
	Range		5	
	Interquartile Range		2	
	Skewness		-,010	,240
	Kurtosis		-,556	,476

Future work will include a way of allowing the users to rate the relevance of the found videos in such a manner that search results that are not appropriate will be ignored in future searches. Ways of implementing this include, but are not limited to, like and dislike buttons, as the ones from the YouTube platform, or 5-star based ratings.

Another future feature is the addition of a new step to the algorithm, that will make possible the creation of a video containing all the needed information by cropping and linking together parts from the relevant videos found based on the existing algorithm. In this manner, one can choose the time that he wants to invest in learning something and his current knowledge of the domain. The algorithm will generate as a result a single video of the selected length containing relevant information extracted from multiple e-learning materials.

DOI: 10.12948/issn14531305/23.2.2019.02

Acknowledgement
This paper presents results obtained within the PN-III-P1-1.2-PCCDI-2017-0272 ATLAS project ("Hub inovativ pentru tehnologii avansate de securitate cibernetică / Innovative Hub for Advanced Cyber Security Technologies"), financed by UEFISCDI through the PN III – "Dezvoltarea sistemului national de cercetare-dezvoltare", PN-III-P1-1.2-PCCDI-2017-1 program.

References

[1] "YouTube by the Numbers: Stats, Demographics & Fun Facts," Omnicore, 6 January 2019. [Online]. Available: https://www.omnicoreagency.com/youtube-statistics/. [Accessed 19 February 2019].

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

[3] B. Iancu, "Evaluating Google Speech-to-Text API's Performance for Romanian e-Learning Resources," *Informatica Economică,* vol. 23, no. 1, pp. 17-25, 2019.

[4] J. Daiber, M. Jakob, C. Hokamp and P. N. Mendes, "Improving Efficiency and Accuracy in Multilingual Entity Extraction," in *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*, 2013.

[5] B. Iancu and T. M. Georgescu, "Saving Large Semantic Data in Cloud: A Survey of the Main DBaaS Solutions," *Informatica Economică,* vol. 22, no. 1, pp. 5-16, 2018.

[6] B. Iancu and I. Smeaureanu, "The Usage of Ontologies for the Discovery and Coupling of Learning Active Components," ASE, Bucharest, 2015.

[7] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe and X. Zhou, "Semisupervised Feature Selection via Spline Regression for Video Semantic Recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 252-264, 2015.

[8] L. Bai, S. Lao, G. J. F. Jones and A. F. Smeaton, "Video Semantic Content Analysis based on Ontology," in *International Machine Vision and Image Processing Conference*, Kildare, Ireland, 2007.

[9] Y. Li, G. Rizzo, J. L. R. García, R. Troncy, M. Wald and G. Wills, "Enriching Media Fragments with Named Entities for Video Classification," in *Proceedings of the 22nd International Conference on World Wide Web*, Rio de Janeiro, Brazil, 2013.

[10] B. Farhadi and M. B. Ghaznavi-Ghoushchi, "Creating a Novel Semantic Video Search Engine Through Enrichment Textual and Temporal Features of Subtitled YouTube Media Fragments," in *3rd International Conference on Computer and Knowledge Engineering (ICCKE 2013)*, Mashhad, 2013.

[11] YouTube, "YouTube Data API," Google, 2019. [Online]. Available: https://developers.google.com/youtube/v3/. [Accessed 14 May 2019].

[12] DBpedia, "DBpedia," 2019. [Online]. Available: https://wiki.dbpedia.org/. [Accessed 14 June 2019].

[13] T. D. C. M. Initiative, "The Dublin Core Metadata Initiative," 2019. [Online]. Available: http://dublincore.org/. [Accessed 14 May 2019].

[14] B. Iancu, "Comparative Analysis of the Main SaaS Algorithms for Named Entity Recognition Applied for Romanian Language," *Romanian Journal of Information Technology And Automatic Control - Revista Romana de Informatica si Automatica*, vol. 28, no. 1, pp. 25-34, 2018.

[15] "IBM SPSS software," IBM, 2019. [Online]. Available: https://www.ibm.com/analytics/spss-statistics-software. [Accessed 14 May 2019].

DOI: 10.12948/issn14531305/23.2.2019.02

Bogdan IANCU has graduated The Faculty of Cybernetics, Statistics and Economic Informatics from The Bucharest University of Economic Studies in 2010. He has a master's degree in Economic Informatics (2012) and a PhD in Economic Informatics starting from 2015 in the field of Ontologies and eLearning. He is an Assistant Lecturer in The Department of Economic Informatics and Cybernetics from The Bucharest University of Economic Studies. His current research focuses on semantic technologies and ontologies innovations. Other fields of interest include machine learning, cybersecurity, mobile devices, embedded systems and IoT.

DOI: 10.12948/issn14531305/23.2.2019.02