Creating a Semantic Academic Lecture Video Search Engine via Enrichment Textual and Temporal Features of Subtitled YouTube EDU Media Fragments

Babak Farhadi
Department of Computer engineering, University of Tehran
Tehran, Iran

ABSTRACT
In this paper, we propose a new framework to annotating subtitled YouTube EDU media fragments using textual features such as exert all the basic portions extracted from the web-based natural language processors of in relation to subtitles and temporal features such as duration of the media fragments where proper entities are spotted. We've created the SY-E-MFSE (Subtitled YouTube EDU Media Fragment Search Engine) as a framework to cruising on the subtitled YouTube EDU videos resident in the Linked Open Data (LOD) cloud. For realizing this purpose, we propose Unifier Module of Outcomes of Web-Based Natural Language Processors (UM-OWNLP) for extracting the essential portions of the 10 NLP tools that are based on the web, from subtitles associated to YouTube videos in order to generate media fragments annotated with resources from the LOD cloud. Then, we propone Unifier Module of Outcomes of Web-Based Named Entity (NE) Booster Processors (UM-OWNEBP) containing the six web Application Programming Interfaces (API) to boost outcomes of NEs obtained from UM-OWNLP. We've presented 'UM-OWNLP ontology' to support all the 10 NLP web-based tools ontological features and representing them in a steadfast framework.

Keywords
Subtitled YouTube EDU video, textual metadata, semantic web, video annotation, web-based natural language processor.

1. INTRODUCTION
Nowadays, video sharing platforms, especially YouTube shows that video has become the medium of selection for lots of people interchanging via the Internet. On the other hand, the extraordinary increase of on-line video contents particularly subtitled YouTube EDU videos confront most of the students with an indefinite amount of data, which can only be accessed with advanced semantic multimedia search and special management technologies in order to retrieve the few needles from the giant haystack. The majority of lecture video search engines provide a keyword-based search, where lexical ambiguity of natural language often leads to imprecise and defective results.

For example, YouTube EDU supports a keyword-based search within the textual metadata provided by the video users and owners, accepting all the shortcomings caused by e.g. homonyms. However, enriched and meaningful results are possible through the analysis of the available textual and timed text (caption) metadata with web-based natural language processors of in cooperating with NE booster processors and efficient semantic video annotations, especially given the availability of subtitles metadata on YouTube EDU videos. YouTube EDU provides access to high-quality, educational content, including short lessons, full courses, professional development material, and inspiring speeches, from the world's leading universities, educators, and thought leaders.

We must evaluate and analyze subtitle text in order to calculate the relatedness between textual inputs. For this purpose, we propose using web-based natural language processors to discover relatedness between words that possibly represent the same concepts. Processes for analyzing words and sentences within Natural Language Processing (NLP) include part-of-speech tagging (POST) and word sense disambiguation (WSD). These processes locate and discover the sense and concept that the word represents. NLP is not only about comprehension, but also the ability to manipulate data, and in some cases, produce answers. It is as well forcefully connected with Information Retrieval (IR) for searching and retrieving information by using some of these concepts. A word can be classified into several linguistic word types. These can, for example, be homonyms, hyponyms and hypernyms. Homonyms are synonyms or equivalent words. Hyponyms are the opposite, a more particular instance of the given word. A hypernym is a less special instance than the original word. In addition, to maximize the serviceability of NLP, we need to have specific ontology.

In this paper by utilizing a rich data model based on the skilled web-based natural language processors, the NE booster processors, ontologies and RDF, we've developed a web application that called SY-E-MFSE (Proposed). Previous works of near to our work haven't used all the main portions and ideas of realized in this paper yet and evaluation of all of them is limited to general videos. They have used some web-based natural language processors only for detect NEs (in very limited basic types). For instance, Alchemy API provides advanced cloud-based and on-premise text analysis infrastructure that removes the cost and problem of integrating natural language processors. Some of the Alchemy API main portions are included in concept tagging, entity extraction, keyword extraction, relation extraction, sentiment analysis, text categorization and etc. Behavior of previous works against the OpenCalais and other the web-based natural language processors are in the same way. By using the NE booster processors such as Amazon web service, Google map, wolfram alpha, Google Book, Facebook graph API and Internet Movie Database (IMDB), we can have really boosted semantically meaning of NEs. However, none of the preceding works haven't used the NE booster processors in their approaches. In addition, in their web applications,

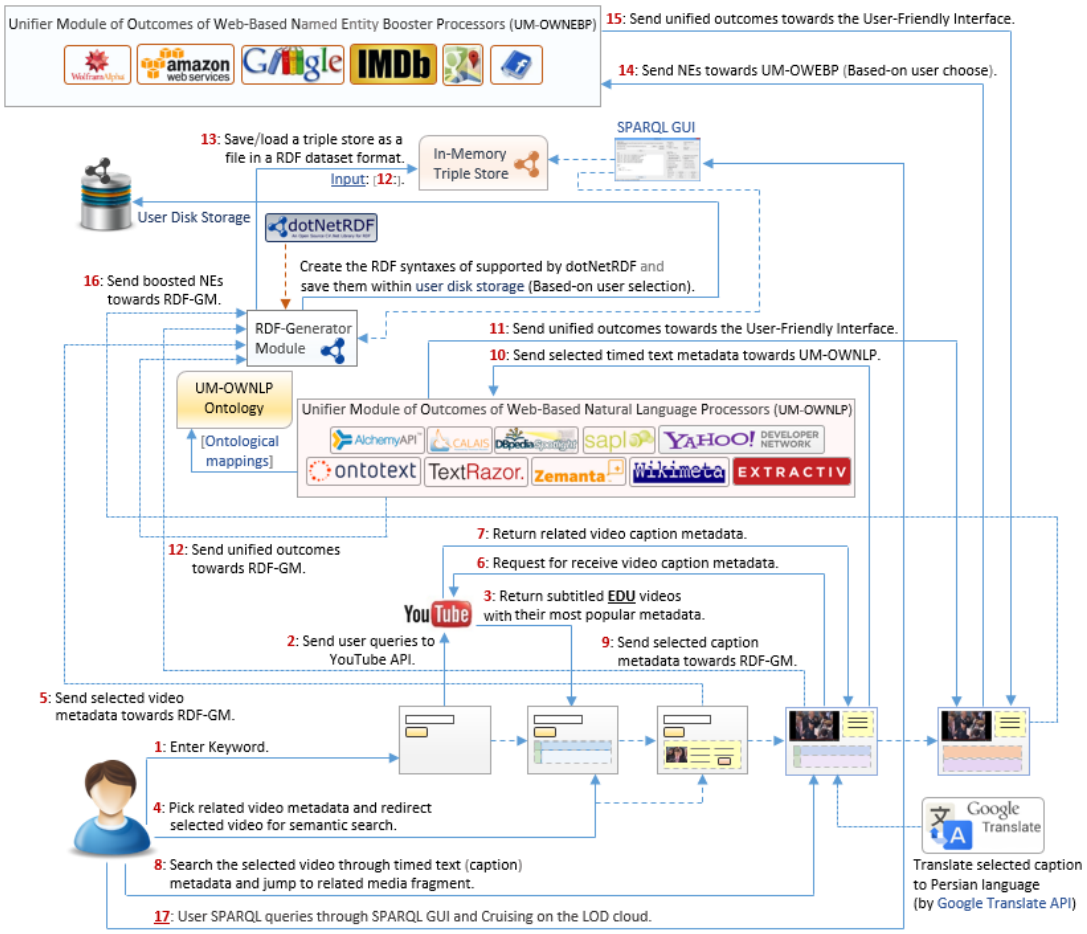


Fig 1: The basic data flow process in proposed SY-E-MFSE.

semantic user can't use SPARQL endpoint for query from RDF datasets. There isn't any SPARQL Graphical user interface (GUI) for run user queries towards triple stores and also there aren't any RDF syntaxes outputs (e.g. RDF/XML, NTriples, Notation 3, Turtle, NQuads, TriG, TriX, RDFa and etc.) for purposes of store into user disk storage, analysis by user and send query to them through SPARQL GUI. SY-E-MFSE (Proposed) framework has eliminated all the difficulties with the previous web applications.

2. RELATED WORK

2.1 Overview on all the previous works related to proposed approach

Many approaches have already presented multimedia and Annotation as LD, which offers experience for us on video resource representing. The LEMO framework provides an integrated model to annotate media fragments while the annotations are enriched with textually related information from the LOD cloud. In LEMO, media fragments are presented using MPEG-21 terminology. LEMO must convert available video files to MPEG suitable version and emanate them from LEMO server. In addition, LEMO derivatives a core Annotation schema from "Annotea Annotation Schema" in order to link annotations to media fragments identifications [1]. Website of "yovisto.com", handle a great amount of recordings of academic lectures and conferences for end clients to search in a content-based method. Yovisto presents an academic video search platform by spread the database containing video and annotations as LD [2] and [9]. It uses Virtuoso server in [3] to propagate the videos and annotations in the database and MPEG-7, Core Ontology of Multimedia (COMM) in [4] to describe multimedia data. Yovisto provides both automatic video annotations based on video analysis and participatory user-generated annotations, which are moreover linked to entities in the LOD cloud with the target to better the search ability of videos.

SemWebVid automatically generates RDF video descriptions using their transcripts. The transcripts are analyzed by three NLP web-based Tools (AlchemyAPI, OpenCalais and Zemanta) but sliced into blocks, which make slack the context for the web-based natural language processors [5]. In [6] has shown how events can be detected on-the-fly through crowd sourcing textual, visual, and behavioral analysis in general YouTube videos, at scale. They defined three types of events: 1) visual events in the sense of shot changes. 2) Occurrence events in the sense of the appearance of an NE and 3) interest-based events in the sense of purposeful in-video navigation by end clients. In occurrence event detection process, they analyzed the available video metadata using NLP techniques, as outlined in [5]. The detected NEs are presented to the user in a list, and upon click via a timeline-like user interface allow for jumping into one of the shots where the NE occurs.

In [10] and [11] we've used the unifier and generator modules for creating an on-line semantic video search engine in the

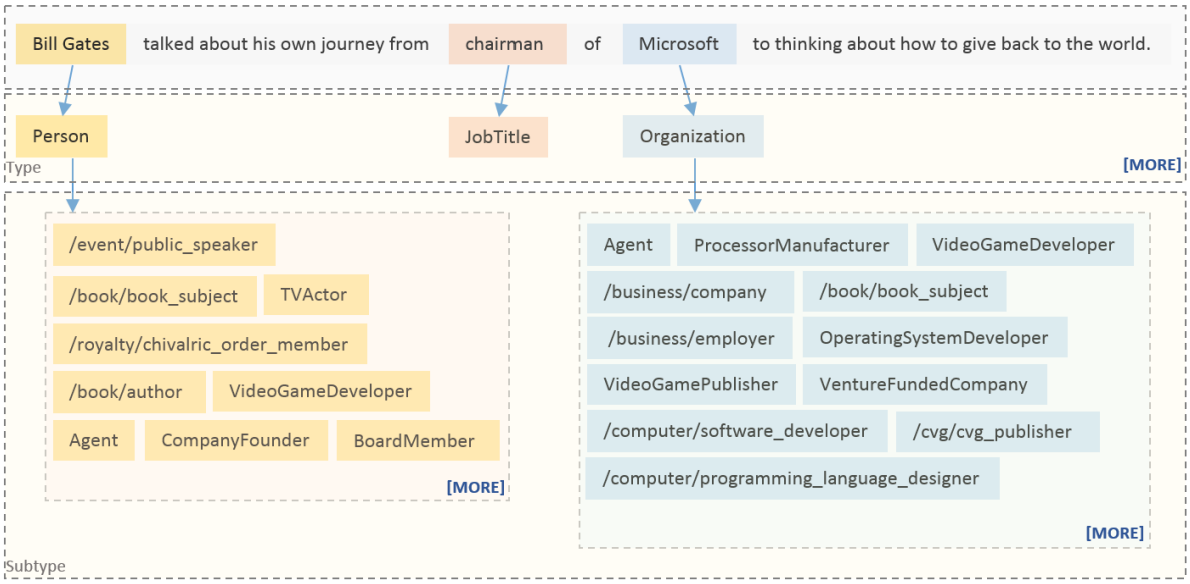
field of subtitled general YouTube media fragments. Therefore, all the experimental results and evaluations have included in subtitled "with-all-categories YouTube" videos area. However, in this paper, we've presented empirical and ontological results, framework, evaluations and search boundary only in subtitled YouTube EDU videos domain.

2.2 Analysis the closest work related to proposed approach

In [7] proposed an approach using the 10 web-based natural language processors based on NERD module in [8]. They only used the portion of NE extraction of these processors and in very restricted types. By having an overview on their demo, end client realizes that he/she can't use advantages RDF syntaxes outputs, SPARQL GUI and triple store module. We found that the most general YouTube videos of returned on their outcomes aren't with subtitle, and they haven't utilized the main keyword-based textual metadata of YouTube data API, up to now. NERD module grabs outcomes of the 10 web-based natural language processors separately, and it uses these processors only for NE recognition about the 10 main types. In NE extraction, it hasn't grabbed any NE subtypes and other derivatives of entity extraction portion. Hence, NERD has answers of chopped and with restricted types of dependent on NEs. It's ignored the most important portions of the web-based natural language processors such as concept tagging, entity extraction, keyword extraction, relation extraction, sentiment analysis, text categorization, fact detection, topic extraction, meaning detection, dependency parse graph and etc. As we mentioned the most challenging problem on mapping textual and temporal features of subtitled YouTube EDU videos to LOD entities is the existence of ambiguous names and therefore, resulting in a set of entities, which have to be disambiguated. Using all the main portions of the web-based natural language processors and efficient ontological mappings, boosting resulted NEs and also utilizing outcomes by RDF-GM and attached modules of related to it, can be eventuated in an enriched entity set. The NLP web-based tools that used in [7] are included into AlchemyAPI, DBpedia Spotlight, Evri, Extractiv, OpenCalais, Saplo, Wikimeta, Yahoo! Content Extraction and Zemanta. In this paper, we used all the main portions of NLP web-based tools are listed, but the difference is that we utilized TextRazor NLP web-based tool instead of the Evri.

Compared with [7] and according to experiments, we have proper ability in using all the portions of the NLP web-based tools in both text/URI area and by UM-OWNLP. In addition, UM-OWNEBP is an expert module in boosting resulted NEs. Furthermore, RDF-GM, triple store and SPARQL GUI in this paper play an important role in representing enriched results to cruising on the LOD-cloud by users. On the other hand, "UM-OWNLP ontology" has a special ability in representing ontological mappings obtained from UM-OWNLP.

3. FRAMEWORK OVERVIEW



Ontological types and sub types of the selected media fragment in figure 3

Data flow process is shown in Figure 1. It can be summarized as follows: 1) User must enter desired keyword or video id to receive keyword-based textual metadata of available by YouTube data API. 2) Content of requested keyword or video id is sent by a standard REST protocol using HTTP GET requests to YouTube data API. 3) The popular keyword-based textual metadata redirected by YouTube data API to the end client. All the videos are in YouTube EDU category and with subtitles (captions). Returned textual metadata is containing video title, video ID, description, publishing date, updating date, author information, recording location, rating average, five snapshots, user comments, related tags, views, likes, caption language and etc. 4) In here; user has a special ability to pick interesting metadata. Therefore, keyword-based textual metadata of on LOD-cloud is based-on user choice. Then selected video metadata by connected video id redirected to semantic Annotation. 5) The selected video metadata has been sent towards RDF-GM to save into user disk storage. 6) To get related caption metadata of YouTube EDU subtitled media fragment, SY-E-MFSE (Proposed) back-end sends an automatic REST request to YouTube data API. 7) YouTube data API returns related video caption information. It contains subtitle text, start time of media fragment and its duration. We implemented an optimized JavaScript code so that by click on the media fragment number, user jump to related start time. Then from start time to end time of related media fragment has highlighted. 8) In this step, the user can have an advanced search on the caption

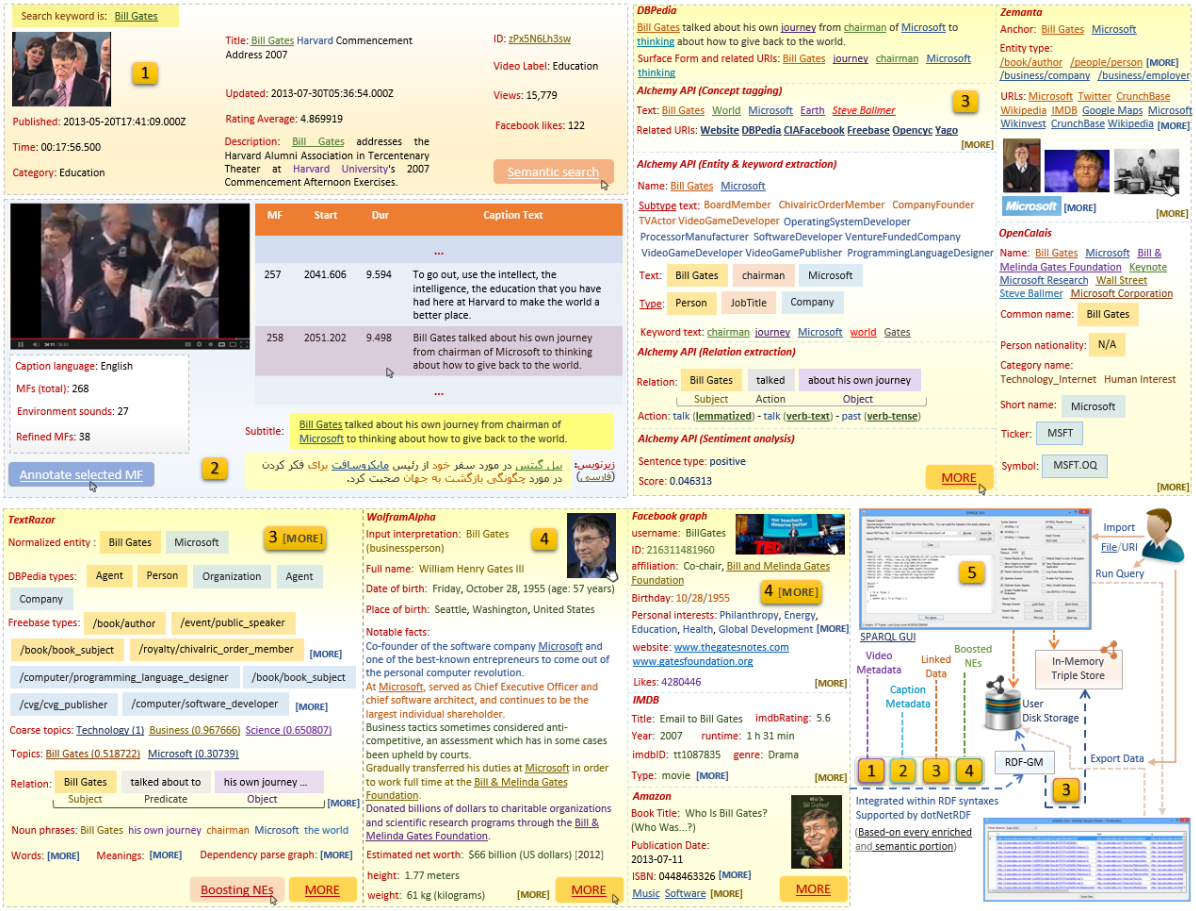


Fig 3: Some overviews on SY-E-MFSE user interface and sample results in a YouTube EDU Media Fragment.

metadata. Simultaneously, he/she has a special ability in select related media fragment and jumps to it. In addition, we have used Google translate API for translate selected subtitle text. default language is Persian. 9) The chosen caption metadata has been sent towards RDF-GM to save into user disk storage. 10) To apply advantages of the web-based natural language processors on media fragment, the selected timed text send to UM-OWNLP. In here we used 10 NLP web-based tools for accomplish an impressive outcome. Apart from the NE extraction and disambiguation, UM-OWNLP use the other main portions of 10 NLP web-based Tools. 11) The unified and enriched output of UM-OWNLP sends towards the user-friendly interface. In addition, we've implemented "UM-OWNLP Ontology" as an efficient module that maintains ontological mappings of generated by 10 NLP web-based tools. For instance, after detect "person" entity type; dbpedia spotlight returns URI of "http://dbpedia.org/ontology/person" towards "UM-OWNLP Ontology". Similarly, we've been collected manually classes, instances, properties, relations and etc. from ontological portions of used within the natural language processors. 12) In this step, optimized result of UM-OWNLP sends towards RDF-GM. We have proposed as a novel RDF Generator Module for the integration of the chosen main keyword-based textual metadata, the selected caption metadata and the unified results of UM-OWNLP and UM-OWNEBP that could be reused for various online media; Generally, according to UM-OWNLP output, RDF-GM generates suitable RDF syntaxes outputs (e.g. RDF/XML, NTriples, Turtle, TriG, TriX, RDFa and etc.) to save them into the user disk storage. By default for every resulted portion of exist in UM-OWNLP, RDF-GM generates correct RDF syntaxes outputs of NTriples, Turtle and RDF/XML. However, this is based-on user choice. We used dotNetRDF library for this purpose, and it supports from many RDF syntaxes outputs. Desired output formats saved on the user disk storage (with the portion name instead of folder name and media fragment number-video id instead of the file name). 13) For every the generated portion of from UM-OWNLP; RDF-GM generates related TriG format to store or load into the in-memory triple store. By this method, user can make special SPARQL query. 14) End client has a particular capability in boost NEs through UM-OWEBP. It contains six the NE booster processors such as Amazon web service, Google map, wolfram alpha, Google book, Facebook graph API and Internet Movie Database (IMDB). For example, "Bill Gates" NE can send to UM-OWEBP and enriched list of about the Relationships of "Bill Gates" with the content of these processors is returned. There is a collection of the lovable NE booster processors. For instance, in Wolfram Alpha API; input interpretation, basic information, image, timeline, notable facts, physical characteristics, estimated net worth, Wikipedia's page hits history about the "Bill Gates" NE is shown. 15) UM-OWEBP outcome sends towards the user-friendly interface. 16) Boosted outcome of UM-OWEBP sends towards RDF-GM. Predefined format that has been stored is RDF/XML. 17) SPARQL is a standard query language for the semantic web and can be used to query over large volumes of RDF data. dotNetRDF provides support for querying over local in-memory data using its own SPARQL implementation. We used SPARQL GUI for testing out

SPARQL queries on arbitrary data sets which user created by loading in RDF-GM (or triple store) and remote URIs. A semantic user can easily query files that have been stored into triple store (with TriG format) and other RDF syntaxes of resulted by RDF-GM. Based-on retrieved YouTube EDU subtitled videos; user can have been cruising on the LOD-cloud. Simultaneously, in here, the semantic user can generate descriptions of resulted objects in a media fragment. In addition, end client can send own annotations and descriptions from the outcomes of integrator modules (text/URI), towards RDF-GM for completing the process of generate enriched RDF/XML file. Finally, we've enriched RDF/XML file connected with video metadata, caption metadata, boosted NEs and LDs of related to the subtitled YouTube EDU video media fragment.

4. EVALUATION

Since the YouTube data API returns the results of related to requested subtitled YouTube EDU video, therefore, SY-E-MFSE search strategy is dependent on to the Google search strategies.

In case of precision, we divided tested search queries into: 1) one-word queries 2) simple multi-word queries and finally 3) complex multi-word queries. For "more relevant", "less relevant" and "irrelevant" categories, the assigned grades are '2', '1', and '0'. The precision is calculated through the formula of sum of the grades of subtitled YouTube EDU videos retrieved by the SY-E-MFSE (proposed), divided to the total number of subtitled YouTube EDU videos that selected for evaluation.

Table1. Precision of SY-E-MFSE for Simple One-word Queries.

Search query	Number of subtitled EDU videos evaluated	More relevant	Less relevant	Irrelevant	Precision
1	50	28	21	1	1.54
2	50	34	16	0	1.68
3	50	33	15	2	1.62
4	50	24	23	3	1.42
Total	200	59.5%	37.5%	3%	1.56

Table 2. Precision of SY-E-MFSE for Simple Multi-word Queries.

Search query	Number of subtitled EDU videos evaluated	More relevant	Less relevant	Irrelevant	Precision
1	50	29	21	0	1.58
2	50	35	12	3	1.64
3	50	34	16	0	1.68
4	50	27	21	2	1.5
Total	200	62.5%	35%	2.5%	1.6

Table 3. Precision of SY-E-MFSE for Complex Multi-word Queries.

Search query	Number of subtitled EDU videos evaluated	More relevant	Less relevant	Irrelevant	Precision
1	50	24	23	3	1.42
2	50	29	14	7	1.44
3	50	30	10	10	1.4
4	50	21	20	9	1.24
Total	200	52%	33.5%	14.5%	1.37

5. CONCLUSION

In this paper, we discussed a new way for efficient semantic indexing subtitled YouTube EDU media fragment content through extracting the main portions from the captions with web-based natural language processors. We introduced the integrator modules that their outcomes are associated with subtitled YouTube EDU media fragments. LD has provided a suitable way to expose, index and search media fragments and annotations on semantic web using URI for identification of resources and RDF as a structured data format. In here, LOD can aims to publish and connect open but heterogeneous databases by applying the LD principles. The aggregation of all LOD data set is denoted as LOD Cloud. Finally, by implementing the important semantic web derivatives such as SPARQL GUI and RDF-GM, end-client can have affective cruising on the LOD-cloud. On the other hand, by tools such as 'CaptionTube' user can effortlessly create suitable captions for own YouTube EDU videos. Therefore, with minimum distribution of hundreds of thousands YouTube EDU videos in a moment and easily convert them to subtitled YouTube EDU videos by owners, we could have efficient semantic indexing and Annotating on subtitled YouTube EDU specific contents by SY-E-MFSE (proposed). For future works, we plan to apply proposed approaches to object-featured video summarization and video categorization field. In addition, we're developing a query-builder interface which creates the SPARQL queries and thereby, end user without have knowledge about the SPARQL queries can easily interact with enriched and semantic files located on the user disk storage.

6. ACKNOWLEDGMENTS

We'd like to acknowledge the contributors to this project such as service supporters of the famous web-based natural language processors such as AlchemyAPI, DBpedia Spotlight, TextRazor, Zemanta, OpenCalais and many others that helped us in implementing this project.

7. REFERENCES

[1] B. Haslhofer, W. Jochum, R. King, C. Sadilek, and K. Schellner, "The LEMO annotation framework: weaving multimedia annotations with the web," International Journal on Digital Libraries, vol. 10, pp. 15-32, 2009.

[2] J. Waitelonis and H. Sack, "Augmenting video search with linked open data," in Proc. of int. conf. on semantic systems, 2009, pp. 1-9.

[3] O. Erling and I. Mikhailov, "RDF Support in the Virtuoso DBMS," in Networked Knowledge-Networked Media, ed: Springer, 2009, pp. 7-24.

[4] R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura, "COMM: designing a well-founded multimedia ontology for the web," in The semantic web, ed: Springer, 2007, pp. 30-43.

[5] T. Steiner and M. Hausenblas, "SemWebVid-Making Video a First Class Semantic Web Citizen and a First Class Web Bourgeois," in ISWC Posters&Demos, 2010, pp. 1-8.

[6] T. Steiner, R. Verborgh, R. Van de Walle, M. Hausenblas, and J. Gabarró Vallès, "Crowdsourcing event detection in YouTube videos," 2012, pp. 58-67.

[7] Y. Li, G. Rizzo, R. Troncy, M. Wald, and G. Wills, "Creating enriched YouTube media fragments with NERD using timed-text," pp. 1-4, 2012.

[8] G. Rizzo and R. Troncy, "NERD: A Framework for Unifying Named Entity Recognition and Disambiguation Extraction Tools," in Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics, 2012, pp. 73-76.

[9] J. Waitelonis, N. Ludwig, and H. Sack, "Use what you have: Yovisto video search engine takes a semantic turn," in Semantic Multimedia, ed: Springer, 2011, pp. 173-185.

[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, 2013.

[11] B. Farhadi, "Enriching Subtitled YouTube Media Fragments via Utilization of the Web-Based Natural Language Processors and Efficient Semantic Video Annotations," Global Journal of Science, Engineering and Technology, pp. 41-54, 2013.

Table 4. Outcomes of SY-E-MFSE precision in semantic level of search.

UM-OWNLP portions category	**YouTube EDU channels 'More relevant' media fragments**				**YouTube EDU channels 'less relevant' media fragments**				**Precision average**
	SE	***AI***	***Human-Computer Interfaces***	***PL***	***SE***	***AI***	***Human-Computer Interfaces***	***PL***	
Entity extraction	98	97	95	97	2	3	5	3	97%
Keyword extraction	99	98	98	97	1	2	2	3	98%
Topic extraction	95	97	98	94	5	3	2	6	96%
Word detection	100	100	100	100	0	0	0	0	100%
Sentiment analysis	97	96	98	98	3	4	2	2	97%
Phrases detection	100	100	100	100	0	0	0	0	100%
Text categorization	93	97	94	98	7	3	6	2	96%
Meaning detection	97	91	98	99	3	9	2	1	96%
Concept tagging	91	88	79	84	9	12	21	16	86%
Relation extraction	100	97	100	98	0	3	0	2	99%
Dependency parse graph	100	100	100	100	0	0	0	0	100%
Average	97%	96%	96%	97 %	3%	4%	4%	3%	97%