Semantic Video Entity Linking

Tim Grams, Honglin Li, Bo Tong, Ali Shaban, and Tobias Weller

Data and Web Science Group, University of Mannheim
{firstname.lastname}@uni-mannheim.de

Abstract. Knowledge graphs are an established technology in the field of information retrieval and question answering. However, the focus is mostly on searching web pages and related documents and less on video formats, resulting in the fact that queries on videos for refining the search are often neglected. In this demo, we show a framework for recognizing faces in YouTube videos and linking them to the matching entities in DBpedia using the thumbnails available in DBpedia. By linking the videos from YouTube with the information from DBpedia, more complex search queries can be made possible. We will present both the frontend of the application, including the search, adding more YouTube videos and formulating complex queries, as well as the architecture and the libraries used in the application.

Keywords: Knowledge Graph · Face Recognition · Video Annotation

1 Introduction

Knowledge graphs (KGs) allow to model information in a semi-structured way and are used especially in information retrieval and question answering. For years, knowledge graphs have been used to improve search queries in search engines, but mostly for web pages and related documents. Using knowledge graphs to optimize search queries about video files is rarely used. In video search engines, searches are commonly based on the title and description, available tags for the video, and meta information such as the video format type and the length of the clip. The content itself and valuable information from a knowledge graph are not taken into account to improve the search results. Yet this information is of considerable use and can significantly improve the search, as demonstrated in the past with text search engines. For example, a search query *Give me scenes showing female actors born before 1970 in California* could not be answered by current video search engines. In order to answer such a query, the content of the video has to be analyzed if this information is not available in the title and description and information stored in a knowledge graph such as DBpedia and Wikidata can be considered. As a further challenge, the title and description of the video does not have to match the actual content of the video, making the analysis of the content of the video essential. For answering search queries such as the one posed above, we propose to link the entities occurring in the video to the corresponding entities of a knowledge graph. Within this work, we

have linked videos from the online video platform YouTube based on the people present in them to the matching entities of the DBpedia knowledge graph. We used state-of-the-art edge face-recognition techniques to link the faces recognized in the videos to the thumbnails provided by DBpedia. We used additional images obtained from search requests to increase the performance of the correct matching. The information about the entities presented in the video with the associated link to the matching entity in a knowledge graph is stored in an RDF graph, using existing standards such as Foaf [2] and Dublin Core [3], and is freely available. Using the information linked in this RDF graph, we can answer queries such as the one posed above, as well as more complex queries as the following within this demo:

- Give me scenes showing the founder of Apple
- Give me scenes showing the winner of the 2017 Grand Prix in Hungary
- Give me scenes showing female actors born before 1970 in California
- Give me videos in which the German chancellor of 2019 speaks together with Emmanuel Macron for at least 20 seconds

The remainder of the paper is structured as follows: In Section 2 we introduce the approach and analyze the performance of matching recognized persons in videos to entities of the knowledge graph. In Section 3 we describe what exactly is shown during the demo and specify the added value for the community. In Section 4 we summarize the demo and present an outlook for future work.

2 Multi-Modal Entity Linking

First, thumbnails of entities of type person were extracted from DBpedia. Furthermore, to improve the performance of the recognition of persons in videos, additional images of the corresponding entities were automatically crawled using a Google search. In order to avoid noise and distortion, a maximum of three additional images were extracted from the Google search. The videos are broken down into frames. For a faster processing, a batch of frames at a time is considered. Multitask cascaded convolutional networks were used to extract five landmark key points for each face from the video frames. For improved matching of the DBpedia thumbnails with the later recognized faces from the videos it is important that the input to the face recognition model is always similar in terms of colors and pose. Since the DeepFace library [6] only performs rotations and no affine transformations, we implemented an own alignment function. Using Arcface [4] we created a 512 dimensional vector representing the face of a person. Afterwards, a k-nearest neighbor algorithm was trained with all of the thumbnails representations to find predictions for unknown faces. Experiments have shown that using an approximation of the classification algorithm provides a better balance between runtime and accuracy. Therefore we used the Non-Metric Space Library [1]. The achieved accuracy on the training benchmark of the Arcface representations of the thumbnails from DBpedia and three additional extracted images from Google with the videos from YouTube was 0.85.

We used the l2 metric to measure the distance between the representations of the thumbnails and the representations of the detected faces from the videos. If the distance was less than 1.25, the video was linked to the entity.

We store the information about the linked videos in a Virtuoso knowledge graph. The basic structure builds on previous work on semantic description of videos [7]. Foaf [2], Dublin Core [3] and MPEG-7 [5] are used for annotation. Figure 1 shows the structure for storing the annotations using one video as an example. The upper half of the structure shows the video itself. It has a title and an identifier which points to the link on YouTube. A video can have multiple scenes with each having a start- and endtime and depicting entities.

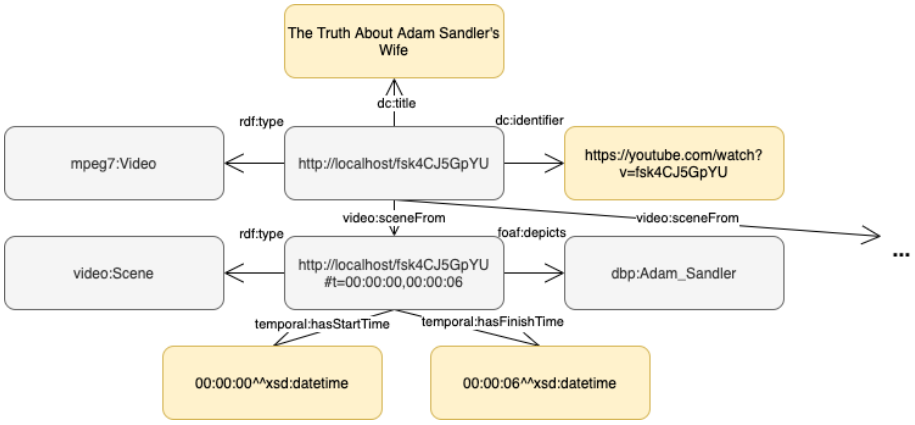

Fig. 1. Example of annotation and linking the recognized entity to DBpedia.

3 Demonstration

Within this demo we will show the application, which is available online[1], including its functionalities and architecture. Interested persons can use a text-based search function to enter names of persons and search for their associated videos, which are publicly available on YouTube. We will suggest examples of persons such as Barack Obama in the demonstration to assist the user in carrying out the demonstration of the application. A complete list of linked entities are available online[2]. The embedded videos from YouTube are annotated with the start and end time in which the entity occurs in the video and the entity linked to DBpedia. In addition to a text-based search, we will also demonstrate a complex search using a SPARQL query. Complex queries can be made and answered using the available information in DBpedia. For example, this can be used to

[1] http://westpoort.informatik.uni-mannheim.de/search
[2] https://bit.ly/3Kfb2fp

retrieve all videos featuring a Canadian actor who lives in Los Angeles. This search query shows the power and the possibilities which are made possible by this application. Along with search, we show how users can index new YouTube videos for our system, thus making them accessible for search. Furthermore, we show how the knowledge graph with the linked entities to the videos can be exported completely or filtered based on a search query using a Virtuoso Knowledge Graph[3]. The code of the demo is available online[4] so that the demo can be run on a local machine.

4 Conclusion and Future Work

In this demo, we show how faces in YouTube videos can be automatically linked to the matching entities from DBpedia to enable complex queries about videos. The information about the recognized entities in the videos is stored in a Virtuoso Knowledge Graph, whose endpoint is publicly available, so that the information can be queried and exported at any time. The source code of the demo is publicly available so that the entire process, including face recognition and link storage, can also be performed locally.

As future work, we aim to use this framework to create a large dataset of YouTube videos linked to DBpedia entities and make the produced data publicity available on the web. Besides, the change in accuracy between images of a person at a different age offers another research opportunity. Therefore, we would like to further refine face recognition by performing age-invariant face recognition to smooth out any outdated thumbnail images of entities in DBpedia.

References

1. Boytsov, L., Naidan, B.: Engineering efficient and effective non-metric space library. In: Similarity Search and Applications. pp. 280–293. Springer Heidelberg (2013)
2. Brickley, D., Miller, L.: FOAF Vocabulary Specification. Namespace Document 2 Sept 2004, FOAF Project (2004), `http://xmlns.com/foaf/0.1/`
3. DCMI Usage Board: DCMI metadata terms. DCMI recommendation, Dublin Core Metadata Initiative (December 2006), published online on December 18th, 2006 at `http://dublincore.org/documents/2006/12/18/dcmi-terms/`
4. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4690–4699 (2019)
5. Martinez, J.: Mpeg-7 overview (version 10) (Oct 2005), `http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm`
6. Serengil, S.I., Ozpinar, A.: Hyperextended lightface: A facial attribute analysis framework. In: 2021 International Conference on Engineering and Emerging Technologies (ICEET). pp. 1–4. IEEE (2021)
7. Sikos, L.F.: Vidont: a core reference ontology for reasoning over video scenes. Journal of Information and Telecommunication **2**(2), 192–204 (2018)

[3] http://westpoort.informatik.uni-mannheim.de/sparql
[4] https://github.com/face-hunters/face-hunter