Technical Disclosure Commons

Defensive Publications Series

October 14, 2017

Natural Language Video Processing (Machine Learning-based Identification, Search, Extraction)

Nathan Frey

Lloyd Thompson

Jeff Chien

Wade Norris

Brian Mulford

Natural Language Video Processing (Machine Learning-based Identification, Search, Extraction)

Natural language parsers allow extraction of salient or important entities from text, allowing search engines to return relevant results without requiring the user to craft queries in unnatural syntax. The extracted entities may be associated with a knowledge graph, generated via machine learning, identifying relationships between terms. For example, the term “window” may be associated with both the terms “computer” and “house”. Based on other extracted terms and relationships, the parser may determine what the most likely intended meaning was.

In typical implementations, the parser may translate from text (e.g. to knowledge graph ID's or other identifiers) by determining nouns and objects from the input strings. Parsers may also be used to identify image content in static images or frames of video. In such implementations, they may sometimes be referred to as fingerprint generators or image analyzers.

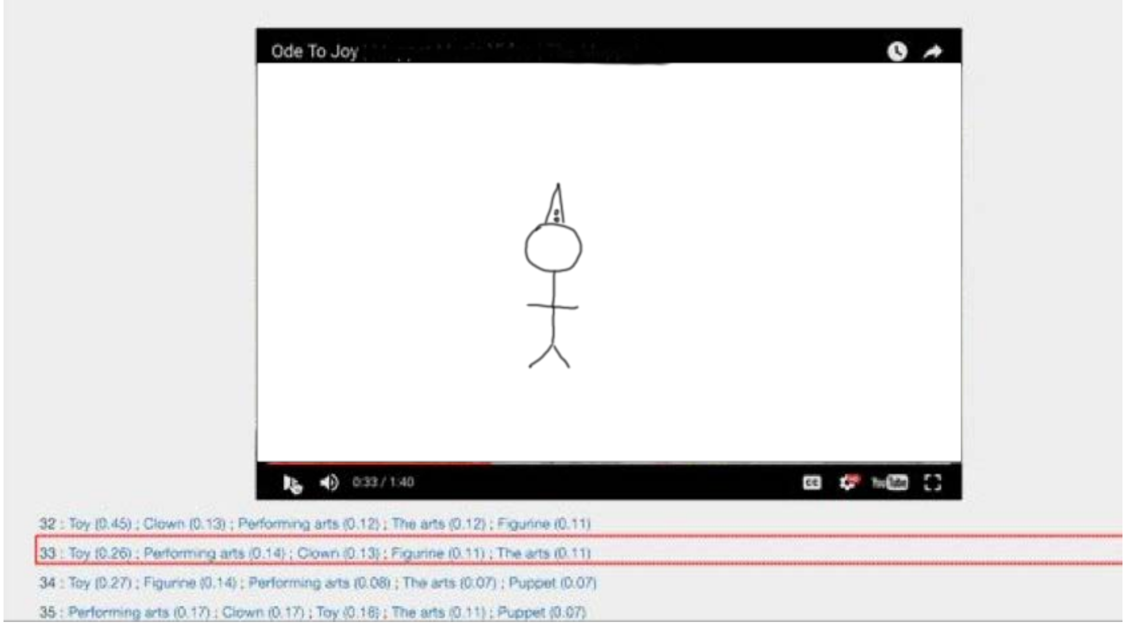
Parsing of text and automatic identification of entities in video may be used together in a combined system to automatically generate videos based off natural language queries. These systems can enable searching within videos for specific elements (rather than just user-entered titles, keywords, and categorization selections) on a frame by frame basis. A user may construct a video using subsets of existing user-generated content without any video editing required (e.g., a person goes to the zoo and makes 25 video-clips of family, animals and the park, and may use the system to automatically generate a video of tigers, apes and giraffes only). Additionally, advertisers wishing to utilize videos for ads but lacking video content can construct a video using stock-footage and their own assets automatically (e.g. an advertiser provides a description of what they wish to create – “extreme sports, winter, and their brand logo” – the system will find matching elements from within a corpus of stock videos and extract relevant clips along with their logo and business name, and deliver a produced set of videos to choose from). Presently, video extraction, building and production is a manual process for video creation. Machine-learning based automation enables non-technical/artistically trained individuals to construct videos easily. In addition, being able to search and extract/identify content from within videos enables a substantial refinement in video search not current possible in social media platforms.

The system works by matching unstructured inputs (e.g. text, images, video, spoken words, inference data from a machine-learning algorithm output) to identifiers within a knowledge graph using entity resolvers that convert descriptive text to factually identifiable entities. Using a semantic matching between the knowledge graph identifiers and annotated values from pre-processed media data, the system can leverage weighted relevancy and taxonomic scores for refinement of matches along with content-safety thresholds to obviate selection of non-usable frames. Matching unstructured to structured data to enable extraction of time-offsets from within videos enables deep search video resolution not previously available. In addition, generative video utilizes automatic machine learning-based graphs that obviate the need for human video editing.

Thus, the system provides a natural-language to generative video process using language-classification and knowledge graph identifiers to find frame segments within a codex of videos that depict the scene described. The mechanics necessary for converting speech to meaningful entities that can be matched to non-text images/videos serve as an underlying basis for generative video. For instance, the system allows generating meaningful videos from pre-existing assets such as text, images and mobile video footage for user-generated content (e.g., extending a photos app to generate videos); or constructing content on-demand based upon descriptive inputs (e.g., images, text descriptions, or market-sourced data).

Concept Examples

The following examples illustrate the concept. This video is captured from a graph that lists entity-based annotations on a per-frame basis from a video. It also demonstrates the inherent limitations with current knowledge graphs’ level of descriptive granularity:



32 : Toy (0.45) ; Clown (0.13) ; Performing arts (0.12) ; The arts (0.12) ; Figurine (0.11)

33 : Toy (0.26) ; Performing arts (0.14) ; Clown (0.13) ; Figurine (0.11) ; The arts (0.11)

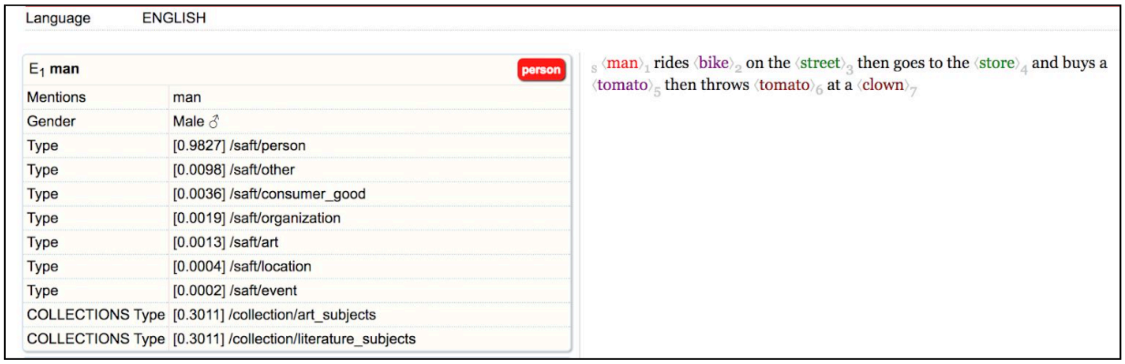
34 : Toy (0.27) ; Figurine (0.14) ; Performing arts (0.08) ; The arts (0.07) ; Puppet (0.07)

35 : Performing arts (0.17) ; Clown (0.17) ; Toy (0.16) ; The arts (0.11) ; Puppet (0.07)

Language examples:

Sample input: “man rides bike on the street. Then goes to the store and buys a tomato. Then throws tomato at a clown.”

Conversion to knowledge graph: [/ID1, /ID2, /ID3.....]



Language ENGLISH

E_1 man	person
Mentions	man
Gender	Male ♂
Type	[0.9827] /saft/person
Type	[0.0098] /saft/other
Type	[0.0036] /saft/consumer_good
Type	[0.0019] /saft/organization
Type	[0.0013] /saft/art
Type	[0.0004] /saft/location
Type	[0.0002] /saft/event
COLLECTIONS Type	[0.3011] /collection/art_subjects
COLLECTIONS Type	[0.3011] /collection/literature_subjects

s ⟨man⟩$_1$ rides ⟨bike⟩$_2$ on the ⟨street⟩$_3$ then goes to the ⟨store⟩$_4$ and buys a ⟨tomato⟩$_5$ then throws ⟨tomato⟩$_6$ at a ⟨clown⟩$_7$

The entity resolver results make it possible to programmatically go from the Text input to knowledge graph identifiers:

Example stubby call:

```
    stubby call $/AnalyzerService.Analyze
"document:{text:'I went to a bowling alley.'}"
```

Salient part of the result:

```
entity <
     name: "bowling alley"
     entity_type: "NON"
     mention <
          phrase <
               start: 4
               end: 5
          >
          head: 5
     >
     type: NOM
profile <
     name: "Bowling alley"
     type: "NON"
     id: 12345678
     identifier <
          domain: FREEBASE_MID
          id: “ID1"
```

Video Creation

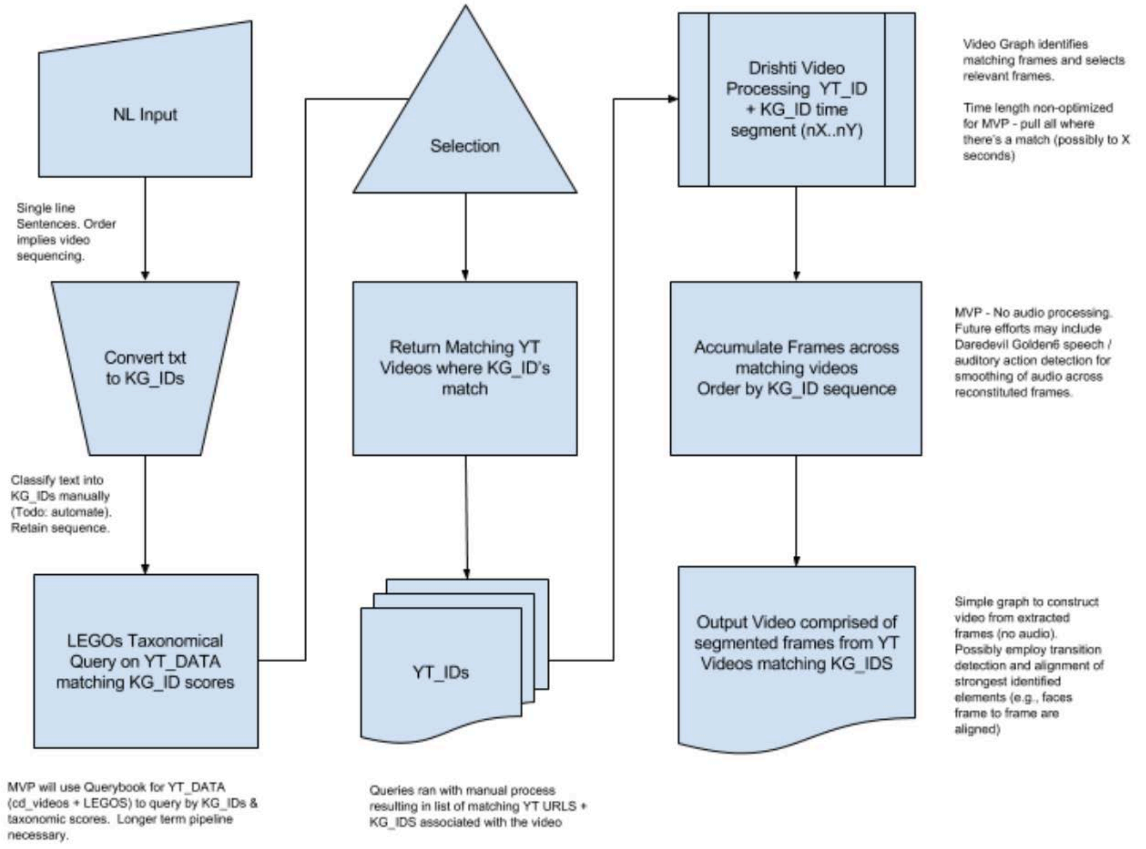
The system can provide a simple Text to Generated Video experience with successful generation of a novel video from identified annotated video frames that match supplied input text. Inputs may be based on any machine-perceivable input data, such as written text, verbal entry, etc. While knowledge graph identifiers are generally coarse-grained and insensitive in discerning homonyms in language, in some implementations, deeper-tree nodes may be utilized to improve input to classifier matching. Semantic text analysis and supervised matching, for example, may be used to enable a semi-automated conversion process.

Video sources may be of any type, including stock-video, user-generated video, or licensed videos, as well as images and rendering. Videos may be identified, on a frame by frame basis, with corresponding entities or identifiers for objects present within the frame or salient points.

As the video library grows, there may be millions of matches for every given knowledge graph identifier. Accordingly, in some implementations, to improve video continuity, searches or video sources may be limited or filtered based on various parameters (e.g. strength of match score; concretely identifiable elements within the video such as brand logo, actors face, product or known object; etc.).

A special consideration may be required for video-to-video stitching. If the system selects video segments with people or places from multiple videos, the resulting generated video may lack continuity of narrative. In some implementations, stricter subsequent matching or filtering may be applied based upon the first identifiable elements with videos. For example, given a search for a broad term (e.g. “shopping”), a broad filter or no filter may be utilized for selection of the first segment (e.g. a video of a man shopping at a grocery store); once identified, a narrower filter may be applied using parameters from the first segment (e.g. in addition to “shopping”, the addition of “man” and “grocery”), preventing retrieval of further segments that match the broad filter but are lack similar context to the first segment (e.g. a video of a woman shopping at a car dealership).

In one implementation, the work flow pipeline may include receiving an input text, converting the text to knowledge graph identifiers, performing a taxonomical query for videos matching the identifiers, selecting segments and returning corresponding identifiers, processing the segments for ordering and relevancy, aggregating the segments according to the processed order, and outputting the generated video:


NL Input
Single line Sentences. Order implies video sequencing.
Convert txt to KG_IDs
Classify text into KG_IDs manually (Todo: automate). Retain sequence.
LEGOs Taxonomical Query on YT_DATA matching KG_ID scores
MVP will use Querybook for YT_DATA (cd_videos + LEGOS) to query by KG_IDs & taxonomic scores. Longer term pipeline necessary.
Selection
Return Matching YT Videos where KG_ID's match
YT_IDs
Queries ran with manual process resulting in list of matching YT URLS + KG_IDS associated with the video
Drishti Video Processing YT_ID + KG_ID time segment (nX..nY)
Video Graph identifies matching frames and selects relevant frames.
Time length non-optimized for MVP - pull all where there's a match (possibly to X seconds)
Accumulate Frames across matching videos Order by KG_ID sequence
MVP - No audio processing. Future efforts may include Daredevil Golden6 speech / auditory action detection for smoothing of audio across reconstituted frames.
Output Video comprised of segmented frames from YT Videos matching KG_IDS
Simple graph to construct video from extracted frames (no audio). Possibly employ transition detection and alignment of strongest identified elements (e.g., faces frame to frame are aligned)

In particular, generating the identifiers may use semantic analysis and knowledge graphs to pull out classifiers from an input sentence, and return content labels representing keys that can be utilized for video segment search and retrieval. The system may generate a set of entity identifiers (e.g. "shopping", "apple", etc.) and corresponding weighted scores representing the syntactical relevancy of each term.

The system may search a media content database comprising identifiers to images, renderings, or video frames or segments (e.g. identified by starting and stopping times, lengths, durations, or similar parameters) and corresponding knowledge graph classifiers. For example, a video segment may be identified by an identifier for a classifier and starting and stopping times within the segment (e.g. "video 1; starting 00:23; ending 00:30; classifiers 'man', 'shopping', 'grocery'", although in typical implementations, hash index values rather than human readable data may be utilized). For example, a video may be identified as an array of times and identifiers, such as:

```
[VID_ID, Time Offset Start, End]
rab0vyWrR7k,4.5,5.5
kI6UN7r9Etk,4,5.5
```

```
M0jmSsQ5ptw,99,103
n93mOqnXEgg,93,95
TvQXsD0UKBI,7,9
TvQXsD0UKBI,31,34
ZIgHDIX9j8U,2,5
ErgLZvtabcs,87,92
SjNi3qPeOrg,0,3
x8UYO8w9XA0,15,16
x8UYO8w9XA0,79,83
nqdboeQ-vYw,26,30
…
```

The system may then generate the video for output by retrieving segments with identifiers matching the classifiers extracted from the query. Segments may be ordered based on similarity between entities for each segment, in one implementation. For example, given a first segment with entities including “man”, “shopping”, and “grocery”, a second segment may be selected with entities of “man” and “grocery” rather than one with just “shopping”. This may provide some measure of continuity between segments.

Abstract

The systems and methods described herein provide for a natural-language to generative video process using language-classification and knowledge graph identifiers to find frame segments within a codex of videos that depict the scene described. The mechanics necessary for converting speech to meaningful entities that can be matched to non-text images/videos serve as an underlying basis for generative video.