
Visual Clustering Service and Similarity Metric

Overview

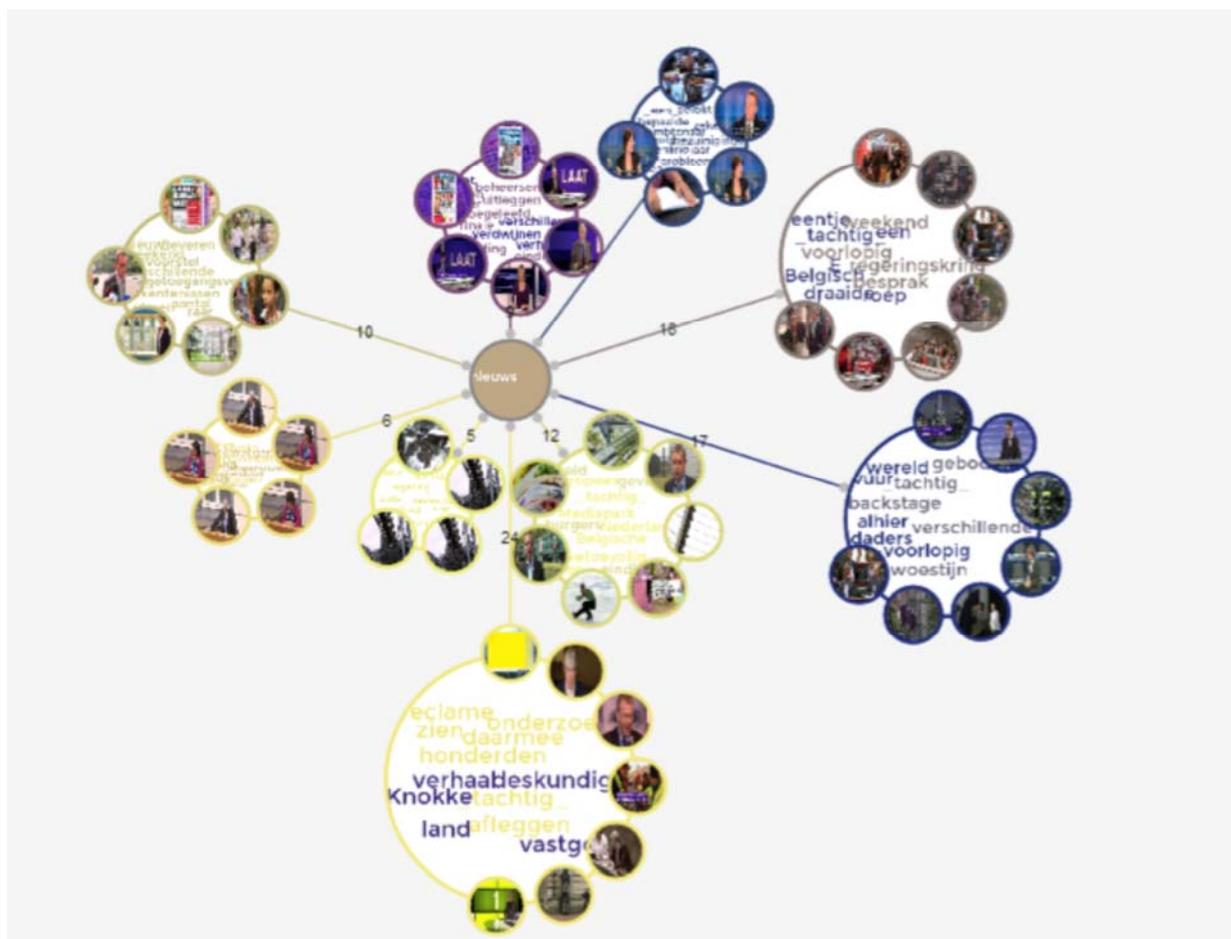
The visual clustering service allows to visually cluster together key frames of a set of content, such as the results of a media search engine. Moreover, the same similarity metric used for the visual clustering can be used to retrieve visually similar search results for any given search result. It consists of a back-end clustering service and a front-end HTML5 visualization.

In depth description

A new search result visualization approach was developed based on clustering of visually similar search results, which may allow users to find items faster in a media collection. Each search result is a video with time-annotated items (shots) and a key frame for each shot. The search result list of the search engine is no longer a linear list of results: instead, the user receives a screen with key frames at the time codes in the video where the query matches. This assumes time-annotated metadata for the video. The key frames on the screen will be clustered based on their visual similarity. For each cluster, only the most representative key frames are shown. The user can select a cluster and receives a new visualization screen with subclustering of the key frames of the selected cluster. Again, only the most representative key frames for each cluster are shown. This hierarchical way of clustering allows the user to refine the search. When the user selects an item, the video can be played and the part of the video transcript that corresponds to the key frame is shown. Users can choose the number of clusters/subclusters, the number of key frames shown for each cluster and the extracted visual feature to cluster upon.

The back-end consists of a Python REST service and database to store descriptors and distances. The REST service contains methods for searching & clustering, subclustering and re-clustering using a different number of clusters. It also contains a method to automatically derive an ideal number of clusters. The search & cluster forwards the search query to the search engine of partner PLY and clusters the output results. The clustering itself is performed with a k-medoids clustering algorithm using color histogram and edge descriptors.

Further, a front-end view has been developed in HTML5 and JavaScript. It displays a summary of the output of the clustering algorithm, with the different clusters displayed as circles. For each circle, some prototypical examples are shown on the edge of the circle. The size of the circle is determined by the number of items that is included in a cluster. By clicking on a cluster, you can zoom in and the contents of the selected cluster are re-clustered.



Potential fields of Application

This technology can be included as a part of a media search engine. It provides an alternative method to view search results as opposed to a linear paged list of results.

Possibilities for exploitation

They include the usage of the component in our own search engine, sharing the knowhow with our spin-off Limecraft and including the technology as background knowledge in the new STEAMER project.

Further Information

Further technical information is available in TOSCA-MP deliverable D3.4.

Contact Person

Dr. Mike Matton
 VRT Research & Innovation
 Auguste Reyerslaan 52
 1043 Brussel, BELGIUM
mike.matton@vrt.be