Deliverable D2.2: Social media filtering and extraction, pre-processing and annotation, intermediate version

Lyndon Nixon, Shu Zhu, Fabian Fischer / MODUL Technology
Evlampios Apostolidis, Foteini Markatopoulou, Vasileios Mezaris / CERTH

30/06/2017

Work Package 2: Media Selection and Analysis

InVID - In Video Veritas: Verification of Social Media Video Content for the News Industry

Innovation Action

Horizon 2020, Research and Innovation Programme

Grant Agreement Number 687786

Dissemination level	CO (A public version of the deliverable will be made available)
Contractual date of delivery	30/06/2017
Actual date of delivery	30/06/2017
Deliverable number	D2.2
Deliverable name	Social media filtering and extraction, pre-processing and annotation, intermediate version
File	InVID_D2.2_v1.0.tex
Nature	Report
Status & version	Final & V1.0
Number of pages	69
WP contributing to the deliverable	2
Task responsible	MODUL
Other contributors	CERTH
Author(s)	Lyndon Nixon, Shu Zhu, Fabian Fischer / MODUL Technology Evlampios Apostolidis, Foteini Markatopoulou, Vasileios Mezaris / CERTH
Quality Assessors	Denis Teyssou / AFP, Roger Cozien / EXO
EC Project Officer	Miguel Montarelo Navajo
Keywords	Topic Detection, Story Detection, Breaking News Detection, Burst Detection, Social Media, Social Networks, Social Web, Twitter Video, Social Media Collection, Social Media Extraction, Social Media Filtering, Social Media Retrieval, Video Fragmentation, Concept Detection, Video Annotation, Thumbnail Extraction, Social Media Metrics, Social Media Reach, Social Media Authoritativeness

Abstract

This deliverable provides an up-to-date summary of Social Media Filtering and Extraction in the InVID project. The update reflects on the progress made in the past 12 months, and references all methods and components that have been implemented and integrated into an InVID platform workflow. This covers:

– Story Detection - an algorithm to extract distinct newsworthy stories out of a Twitter stream, label those stories in terms of the most significant keywords that define that story, and rank those stories by volume of social media content being generated that refers to that story.

– Social Media Extraction and Annotation - a set of components to generate story-based queries at regular intervals on social platform APIs in order to retrieve timely and relevant video content for those stories.

– Video Annotation - a set of components for the fragmentation of video content, the extraction of thumbnails for videos and their fragments, and labeling of each video fragment with a set of most representative visual concepts in that fragment.

Content

List of Figures

List of Tables

1 Introduction

The topic of this deliverable is to outline the successful execution of the social media filtering and extraction pipeline in the InVID project according to our initial planning and prototyping and, based on internal evaluations of the current quality of the filtered and extracted data, the formulation of planned improvements for reaching the final version of the pipeline in one year from now (to be reported in deliverable D2.3).

The structure of the deliverable is as follows:

- Story Detection - the first section will describe our chosen technological method for extracting stories from the social media stream, their disambiguation and relevance.
- Social Media Extraction and Annotation - the following section will describe our applied approach to information retrieval from large scale social media sources in order to acquire newsworthy media, as well as proposed extensions for the original metadata model.
- Video Annotation - the next section will present the development and evaluation of a service for temporal fragmentation of user-generated videos and the conceptual annotation of those fragments.
- Outlook and Next Steps - based on the related evaluations of the above achievements, this final section will outline proposals for further improvements to the social media filtering and extraction pipeline and list the next steps we plan to implement those proposals in the following project year.

1.1 History of the document

Table 1: History of the document.

Date	Version	Name	Comment
2017/05/19	V0.1	L. Nixon	suggested Table of Contents
2017/05/23	V0.2	E. Apostolidis	updated Table of Contents
2017/06/01	V0.4	L. Nixon, S. Zhu	first MOD input
2017/06/02	V0.6	E. Apostolidis, F. Markatopoulou	first CERTH input
2017/06/14	V0.7	L. Nixon, F. Fischer	first full content from MOD
2017/06/16	V0.8	E. Apostolidis, V. Mezaris	first full content from CERTH, sent to QA
2017/06/23	V0.9	L. Nixon	extended story detection & conclusion
2017/06/30	V1.0	L. Nixon, E. Apostolidis	complete final version

1.2 List of abbreviations

Table 2: Acronyms used.

Acronym	Explanation
API	Application Programming Interface
BRIEF	Binary Robust Independent Elementary Features
CCTV	Closed-Circuit TeleVision
DCNN	Deep Convolutional Neural Networks
DCT	Discrete Cosine Transform
FTP	File Transfer Protocol
HHMM	Hierarchical Hidden Markov Models
HSV (histogram)	Hue, Saturation, Value
HTTP	HyperText Transfer Protocol
IPTC	International Press Telecommunications Council
JSON	JavaScript Object Notation
KB	Knowledge Base
MDL	Multi-Domain Learning
MTL	Multi-Task Learning
MXinfAP	Mean eXtended inferred Average Precision
NEK	Named Entity keywords
NEL	Named Entity Linking
NER	Named Entity Recognition
NLP	Natural Language Processing
ORB	Oriented FAST and rotated BRIEF
PLK	Pyramidal Lucas-Kanade
REST	Representational State Transfer
RGB (color model)	Red, Green, Blue
SGD	Stochastic Gradient Descent
SIFT	Scale-Invariant Feature Transform
SKB	Semantic Knowledge Base
SURF	Speeded Up Robust Features
SVM	Support Vector Machine
UGC	User Generated Content
UGV	User Generated Video
UI	User Interface
URL	Uniform Resource Locator

2 Story Detection

In our initial proposal we presented a conceptual workflow for ”Topic Detection”. Our goal was to automatically identify newsworthy events which could guide journalists to online media being posted in association with that event (and which may require verification before it can be used in the professional news cycle). We had three primary requirements to address in the InVID context, which took our work away from the classical research activities in topic detection:

– Timeliness of detection of a new newsworthy event;

– Addressing multilinguality and alternative names in the detection approach;

– Quantifying the newsworthiness of the event as suitable for extracting eyewitness media.

Given that the results of our approach are referenced as Stories and that the InVID Dashboard already has a model for classification of content called Topics, we have chosen to use the terminology ”Story Detection” to refer to the InVID activity of news event extraction from social media, with Topics being used in the dashboard as an additional tool to classify those extracted news events (see later in the section).

The next subsection reflects on how our work relates to the state of the art in the area of topic detection. We established that social media streams, in particular Twitter, are the most effective sources of data for this task and presented a workflow model for story detection (Fig. 1) which can be explained in terms of the current implementation thus:

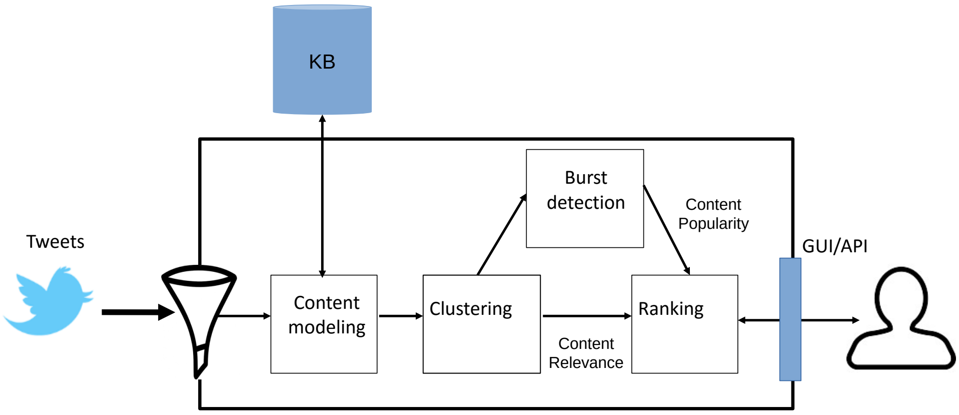


Figure 1: InVID workflow for story detection.

– Content modeling - we model each tweet as a bag of keywords based on natural language processing and we seek to perform keyword alignment based on named entity recognition;

– Clustering - we chose a community detection algorithm as a means to cluster tweets and configured our approach to better disambiguate between distinct stories and merge overlapping stories;

– Burst detection - we experimented with an alternative means of organization of stories based on burst detection approaches to more strongly highlight recently emerged new events;

– Ranking - we explain the alternative ranking possibilities that can be provided and the use of topics in the dashboard as an effective classification mechanism.

Furthermore, we complete this section by presenting our chosen methodology for performing an evaluation of the story detection approach and the baseline results for the current implementation of the pipeline.

2.1 Relation to State of the Art

News are often centered around specific events (happenings), which provide a natural way to group the news stories (Wu, Chen, & Giles, 2015). It can be seen as a specialisation of the idea of topics, where the topic is something happening in the physical world over a certain time period. There exist several on-line services that mine events from news articles in different languages:

– European Media Monitor[1] (Pouliquen, Steinberger, & Deguernel, 2008);

– GDELT project[2] (Leetaru & Schrodt, 2013);

– Event Registry[3] (Leban, Fortuna, Brank, & Grobelnik, 2014; Rupnik et al., 2015)

Recent work on breaking news detection is centered around Twitter as the major source of news stream data. (Hu et al., 2012) confirmed ”Twitters rising potential in news reporting“ that can ”fundamentally change the way we produce, spread, and consume news“. Their analysis showed that ”the people who broke the news were able to convince many Twitter users before confirmation came from mass media“ using Bin Ladens death case study. Thus, it provides an evident motivation for the real-time breaking news detection from the Twitter stream.

[1] http://emm.newsbrief.eu/NewsBrief/clusteredition/en/24hrs.html
[2] http://www.gdeltproject.org/
[3] http://eventregistry.org

Twitter provides location-specific trends (keywords or hashtags) determined by an algorithm, which identifies ”popular topics among users in a specific geographic location ... to help you discover the hottest emerging topics of discussion on Twitter”.[4]. The Twitter API provides access to the list of the current global trending topics as well as localised trends. However, this is a purely algorithmically generated list which does not distinguish topics which may reference newsworthy events from any and all other topics of current popular discussion.

Topic modeling is a common approach that can be applied to detect breaking news on Twitter (Cataldi, Di Caro, & Schifanella, 2010; Aiello et al., 2013; Wold & Vikre, 2015). Topic detection (modeling) algorithms, such as Latent Semantic Analysis (LSA) (Deerwester, Dumais, Landauer, Furnas, & Harshman, 1990; Landauer, Foltz, & Laham, 1998) or Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003), provide means to organize a collection of electronic documents into semantically coherent groups (topics) based on the frequent word co-occurrence matrix (e.g. TF-IDF metrics). Topic detection approaches often involve topic clustering, ranking and labeling stages (Petkos, Papadopoulos, & Kompatsiaris, 2014; Martin & Göker, n.d.; Van Canneyt et al., 2014; Martin, Corney, & Goker, 2015; Ifrim, Shi, & Brigadir, 2014; Elbagoury, Ibrahim, Farahat, Kamel, & Karray, 2015).

A few approaches to extract open-domain events from tweets were proposed (Popescu, Pennacchiotti, & Paranjpe, 2011; Ritter, Etzioni, Clark, & others, 2012; Katsios, Vakulenko, Krithara, & Paliouras, 2015), but none of them supports cross-lingual linking. Lendvai and Declerck (Lendvai & Declerck, 2015) use the Longest Common Subsequence (LCS) method to extract events and also link tweets to web documents. Topic models were also reported to be useful for aligning tweets with the news articles outperforming the language models (Krestel, Werkmeister, Wiradarma, & Kasneci, 2015). These results may be of particular importance on the evaluation stage, when the gold standard data for the evaluation of tweet clustering algorithms is given in terms of news articles.

Only a few studies focus on other data sources than Twitter stream, in particular Wikipedia (Osborne, Petrovic, McCreadie, Macdonald, & Ounis, 2012; Steiner, van Hooland, & Summers, 2013). Steiner et al. (Steiner et al., 2013) implemented Wikipedia Live Monitor application [5] for the task of breaking news detection. They defined a set of criteria for the news to be considered as breaking based on monitoring of the recent changes, such as speed and number of concurrent edits on the Wikipedia web site. However, we note that comparative reviews of topic detection using different sources have demonstrated that Twitter is the most suitable data source, i.e. better results could not be acquired using news articles, Wikipedia or any other sources.

The task of First Story Detection (FSD) was proposed to identify the first story about a certain event from a document stream (Petrovi, Osborne, & Lavrenko, 2012). The state-of-the-art FSD approaches use similarity metrics over documents, such as TF-IDF vectors or Locality Sensitive Hashing (LSH) (Petrovi et al., 2012; Phuvipadawat & Murata, 2010), to determine if candidate documents are close to existing documents or could constitute a new event. This approaches have been improved by combination with external data sources, such as WordNet to build lexical chains (Stokes & Carthy, 2001) or Wikipedia to rank and filter the produced stories (Osborne et al., 2012). However FSD focuses on the retrospective correct identification of stories (and hence the first document in a story thread) rather than the timely detection of the emergence of a new topic in the document stream.

The SNOW 2014 Data Challenge (Papadopoulos, Corney, & Aiello, 2014) stipulated further research in mining Twitter streams to extract and summarize newsworthy topics (Van Canneyt et al., 2014; Martin & Göker, n.d.; Burnside, Milioris, & Jacquet, 2014; Petkos et al., 2014; Ifrim et al., 2014). The notion of ’newsworthiness‘ was further defined as a set of topics for a given time slot ‘covered in mainstream news sites’ (Papadopoulos et al., 2014). Thereby setting the mainstream news sites as a reference point for the gold standard data on ’important‘ news stories. Later, Martin et al. (Martin et al., 2015) utilized the same definition of newsworthiness to evaluate their method for topic detection on Twitter data, but additionally decomposed the notion into ’the combination of novelty and significance‘. One common method to find novel (emerging or recent trending) topics from a data stream is looking for bursts in frequent occurrences of keywords and phrases (n-grams) (Martin et al., 2015; Martin & Göker, n.d.; Fujiki, Nanno, Suzuki, & Okumura, 2004; Cataldi et al., 2010; Aiello et al., 2013).

Since the 2014 Challenge, the most significant development in the area of story detection has been the application of Artificial Neural Networks (ANNs). These allow to generate dense vector representations (embeddings), which can be efficiently generated on the word level (word2vec) (Mikolov, Chen, Corrado, & Dean, 2013) - as well as on the character level (tweet2vec) (Dhingra, Zhou, Fitzpatrick, Muehl, & Cohen, 2016) - and have been found to be more effective in many domains than other algo-

[4] `https://support.twitter.com/articles/101125?lang=en#`

[5] http://wikipedia-live-monitor.herokuapp.com/

rithms such as LSA and LDA. (Brigadir, Greene, & Cunningham, 2014) demonstrated encouraging results using the word2vec Skip-gram model to generate event timelines from tweets. (Moran, McCreadie, Macdonald, & Ounis, 2016) achieved an improvement over the state-of-the-art rst story detection (FSD) results by expanding the tweets with their semantically related terms using word2vec. Neural embeddings can be efciently generated on the character level as well. They repeatedly outperformed the word-level baselines on the tasks of language modeling (Y. Kim, Jernite, Sontag, & Rush, 2015), part-of-speech tagging (dos Santos & Zadrozny, 2014), and text classication (X. Zhang, Zhao, & LeCun, 2015). The main advantage of the character-based approaches is their language-independence, since they do not require any language-specific parsing. We have provided the first evaluation of character-based neural embeddings on the tweet clustering task for news story detection. We could demonstrate in the experimental evaluation that the proposed approach signicantly outperforms the current state-of-the-art in tweet clustering for breaking news detection (Vakulenko, Nixon, & Lupu, 2017).

Scalability is an important requirement when dealing with the data streams of a high volume and velocity source, like Twitter. The algorithms have to be adequately evaluated to be applicable in the real world scenarios. For example, (Martin & Göker, n.d.) focuses on (near) real-time topic detection by simulating the Twitter data stream to evaluate the efficiency of their trend detection mechanism. The BNgram approach they propose takes two minutes to generate the topic model for a set of tweets collected during a 15-minutes time slot. In our case, we trained our character-based model on 2 million tweets which took slightly less than 3 days. Given that in the task of real-time news detection, the requirements are rather different from evaluating a pre-trained model against a historical data set (from SNOW 2014) and would necessitate updating our trained model regularly (as word associations in the news would change at a rapid rate) the tweet2vec approach is not feasible without use of a much more powerful parallel-GPU infrastructure.

The SNOW 2014 Data Challenge has confirmed newsworthy topic detection to be still a challenging task: the top F-score of the competing solutions was only 0.4 (Precision: 0.56, Recall: 0.36). The limitations of the current state-of-the-art approaches include early topic detection, topic relevance, topic representation and the performance evaluation of the topic detection methods. We continue with outlining the approach we currently pursue in InVID, which proves to provide a fair balance between real-time performance and story output relevance.

2.2 Content Modeling

As presented in our initial proposal, we base story detection on content extracted from the Twitter Streaming API. We have implemented a set of pre-processing measures to remove irrelevant data from the stream: a word blacklist to strip out spam, a length filter (of 30 characters) to ensure the tweet has enough textual content, as well as a language check as part of the NLP pipeline to ensure the text is in a supported NLP/NER language (currently: English, French and German). Three Twitter streams are configured:

– “Twitter Accounts” follows 61 professional breaking news sources in English, French and German.

– “Twitter Geo” collects geolocated tweets from around the world, and is intended in due course to make use of the disambiguated location tagging of detected news stories to dynamically collect tweets coming from locations where news events are happening at that time. As location disambiguation will be delivered in InVID WP3 this year, we hope to launch this means to collect news tweets by early 2018.

– “Twitter News” collects user generated tweets using predetermined breaking news terms in English, French and German. Collection can vary from 30 to 90 thousand tweets daily.

To cluster each tweet into distinct stories, appropriate models are needed that reduce the complexity and support the task of clustering, which is based on computational calculation of similarity between documents. Given the scale of documents to be clustered, the model should also support a computationally inexpensive clustering method. The Baeza-Yates algorithm for indexing large text corpora by statistically significant n-grams has been provably efficiently scalable for approximate string matching tasks. Using this algorithm, we model each tweet as a set of n-grams as keywords, with tf-idf measures avoiding that overly frequent or extremely rare strings become keywords. The quality of keywords plays an important role in the clustering of the documents into stories and thus the continual evaluation of the quality of stories also led to feedback on the keyword results which needed to be addressed.

Besides the term blacklist, we gradually added to a specific stoplist for InVID keywords which contains n-grams which should not become keywords in the keyword detection step. For example, we found that many news organisations tweets contain their own organisations name so we had to add those names to the stoplist (e.g. BBC, NBC, SABC) in order to avoid distinct tweets being annotated with the same (organisational) keyword, which in turn would lead to clusters of tweets which are actually about distinct stories but share that keyword.

We also had to deal with the issue that we were collecting tweets in different languages but wanted to have a combined view of the stories emerging in those Twitter streams. For this, we identified the need for a mapping between the keywords in different natural languages so that we could consider distinct keywords as having the same word sense for the clustering step. This also has the additional advantage of providing a means to view and browse the keywords in the Dashboard in the same natural language, independent of the original language from which the keyword was extracted. To do this, we have initiated a Semantic Knowledge Base (SKB) which defines each word sense with an unique identifier and associates it with multiple keyword n-grams (or labels) annotated with their natural language. To reduce complexity, our initial implementation supports as core the three natural languages English, German and French.

In examining our stories, we observed that one of the most significant reasons why the same news story was being represented as multiple stories (clusters), was a lack of alignment between the bag of keywords of each cluster, where a human observer could identify that distinct n-grams had actually the same meaning. If those keywords could be aligned to the same word sense, just as was being done now for multilingual keywords, but within the same natural language, we would achieve a far greater precision in clusters representing single, distinct news stories. To solve this, we have initiated a new approach to keyword extraction which we term Named Entity Keywords (NEKs). We note that classically the problem of alignment in meaning of distinct texts has been addressed by the tools of Named Entity Recognition (NER) and Named Entity Linking (NEL). These identify strings in input text as being a reference to an entity (distinct concept), either unknown (then usually only typed as belonging to an identifiable class such as Person, Organisation or Location) or known (then usually matched to the label of an identifiable entity from a Knowledge Base). Our NER/NEL tool Recognyze works with dictionaries of labels (known as ”surface forms” when matching to snippets of input text) to identify candidate (still unknown) entities in the text, and has already demonstrated significant improvements in precision and accuracy in identifying references to locations in text through the expansion of the content of those dictionaries combined with implementation of additional syntactic rules (see the later D3.2 for explanations and evaluations of the location disambiguation by Recognyze). Named entity keywords are computed by applying an additional cleaning step during keyword computation, where named entity annotations as provided by Recognyze are grounded in the input text, and all their name alterations are subsequently resolved against their preferred labels as provided by linked open data corpora. Thus, by applying the principles of NEKs to the text from which the keywords are being extracted, we can map varying surface forms to the same entity and thus align varying keywords to a single Named Entity Keyword (NEKs). We have tested the application of the NEKs approach within the platform prototypically, with the plan to request its push into the production system in the second half of 2017 alongside the launch of the next generation of the Recognyze tool (again, cf. D3.2 to be delivered Dec. 2017).

2.3 Story Clustering

Building on the continual improvement of the keywords, our clustering algorithm uses the bags of keywords from each document to cluster documents according to the keyword co-occurrence. Since the documents (the JSON metadata produced by our social media extraction pipeline) were being stored on the platform-side in an Elastic Search index, Elastic Search aggregations were effective for this co-occurrence calculation followed by k-means clustering. The baseline approach and its initial results were reported after six months in the project. The next significant improvement to this approach has been the introduction of the Louvain Modularity algorithm. The algorithm is more commonly used to detect communities within social networks - in our case we detect “communities” of related keywords over time. We chose this algorithm as the graph of keywords is specifically structured in the same way as a community of social network contacts and it proved to be much more efficient than k-means as well as performing more scalably than an approach calculating character embeddings with tweet2vec which we also experimented with. This formed the basis of the first full roll-out of stories in the InVID Dashboard (through two visual components called the Story View and Story Graph, described in D5.3).

Observation of the story results in the Dashboard led us to assess story quality and identify errors.

Improving keywords has been discussed in the previous subsection. Besides this, we found we could classify errors in three types:

- disambiguation of the stories (meaning: each cluster should represent one distinct news story);
- relevance of documents in the stories (meaning: all documents within a cluster which is identified as a particular news story should be related to that story);
- irrelevant labels in the story label (meaning: the description of the cluster should clearly identify the news story represented by that cluster and not contain any parts irrelevant to that story).

To find solutions to the above issues, we looked at aspects of our clustering implementation which could help us to improve.

- Keyword post-processing: clusters are defined by groups of keywords. Keywords which are contained by other keywords (e.g. trump, donald trump) are filtered out and keywords which partially overlap with other keywords (e.g. president donald, donald trump) are reduced by removing the keyword with a lower weight in the cluster. This helps align separate clusters according to their contained keywords, although it is only effective in enabling merging when n-gram keywords are currently detected and included in the clusters (e.g. a cluster with 'donald' and 'trump' will not align its keywords if the n-gram 'donald trump' has not also been detected and included in the cluster).
- Merge of overlapping clusters: clusters are defined within a time slice, whose smallest granularity is one hour. Originally, even if two clusters shared the same keywords but there is an hour time slice between them, they were retained as two separate stories. By loosening the thresholds set for the graph partition algorithm (which generates the separated clusters) we could ensure that previously separated clusters which were close enough temporally and semantically could be output as a single story. However, the bags of keywords which made up a cluster proved to often be not syntactically close enough to affect a merge even though a human observer could identify both bags of keywords as being related to the same news story.
- Cluster splitting based on pairwise keyword co-occurrence. We termed the pairwise co-occurring keywords (where a keyword B both co-occurs with keyword A and with keyword C) "triads". The clustering based on graphs of co-occurring keywords relies therefore on binary relations between keywords. We found certain keywords co-occurred heavily with keywords where each binary relation could belong to a separate story (i.e. the keyword was related to several stories at the same time). The graph-based clustering tended to create single clusters across these keyword relations because of the weight of the central keyword ('trump' has been repeatedly a central keyword for such clusters). The result was larger clusters which actually combined two or more stories. Pairwise relations did at times help to split such clusters, e.g. if there was a fire in two places in the same news cycle, there would be a stronger co-occurrence relation between 'fire' and location A and between 'fire' and location B than between location A and location B (if at all). However we found that such splits were generally not as simplistic to detect through pair-wise relations, as the merged stories were not so straightforward to algorithmically separate. For example, if tweets either talked about a London fire or a Kensington fire, we could conclude we have 2 seperate stories when in fact it is the same one.
- Document relevance query. Once stories are detected, the documents in the time period selected in the query (i.e. the time settings of the dashboard) are classified into those stories based on matching document keywords and story keywords. Using the top-3 frequent keywords of the cluster as the story label, we found it could occur that little or no documents in the time period actually contained all three keywords (especially where stories were being merged as one cluster). Using Elastic Search general term aggregation as a relevance sort of matching documents for the selected keywords could mean that documents which match just one keyword are displayed, often leading to documents being shown for the story which were not relevant to that story (especially when the matching keyword was the most general). We had initially set the queries to restrict documents to those matching the same time slice as the story but this reduced too much the set of matching documents with two or three keywords (which were generally the most relevant to the story). Removing the time restriction combined with relevance sort maximises the set of matching documents at the 'top' of the story (those visible under the story label in the dashboard). To further maximise the relevance of documents selected for a story, we would need to alter the keyword

ranking mechanism for the story labels to increase the number of keyword co-occurrences within documents for the set of keywords chosen as the story label.

The above points highlight the experiments we have conducted within the story detection approach to attempt to improve the accuracy of story detection, including appropriate labelling of stories (which in turn can influence the social media retrieval, see later) and relevant classification of documents to stories. The evaluation results below reflect what we have achieved to date to address the identified problems with the recognition that these problems, at some level, all still persist. Our studies have recognised that future improvements to the keywords (particularly the introduction of NEKs) could potentially play a significant role in improving story detection, as we need to be able to better align sets of keywords semantically (rather than syntactically). Our requirement would be to be able to normalise distinct keywords to common entities (President Donald Trump is referenced by at least six different keywords at present), and understand semantic properties of and relations between those entities for more precise cluster merging and splitting (e.g. knowing that a fire in London and Kensington can be the same thing - as Kensington is a part of London - but that a fire in London and in Dhaka are two separate things).

2.4 Burst Detection

Another aspect of the story detection, which became much clearer through the story graph visualisation, was the dominance in the results of our approach of news stories which are the subject of more tweets over more time. This is a natural result of clustering which determines its clusters based on all documents input to the clustering algorithm. In our case, those documents also have a time component (the date-time the document was published) but the temporal distribution of the documents has a limited role in the results of the clustering (we do look at the keyword co-occurrences within time splits, when clustering with a time span of 24 hours of documents this time split is one hour, but we then connect the clusters per time split together to form larger clusters within the overall time span). So for example, if at the end of the current time span a new story appears, it may be highly relevant for the journalist and indeed it may have the most related tweets within the final time split. However, for the Story View earlier stories have already been the subject of tweets over many previous time splits and thus (from volume of documents) are ranked higher, dominating the list of results. To address this, we decided to consider the area of ”burst detection” as a means to better highlight quickly emerging news stories from those which are continually present in the news coverage.

While a story is any distinct news event within the selected time period, we consider a story line (**episode**) to be a set of temporally linear and semantically connected news events over multiple time periods, whereas we consider an emerging story (**burst**) to be a single news event which appears suddenly in the timeline.

Burst detection is a widely applied approach in recent research of event detection (Zimmermann, 2014) (He, Chang, & Lim, 2007) (Yin et al., 2015) (Weng & Lee, 2011) (Cameron, Power, Robinson, & Yin, 2012). However, the above mentioned works focus on the term level detection, i.e. aiming to detect those ”bursty terms” from the whole collection of words in all documents, which is not designed for real-time applications.

It has also become a common method to find novel (emerging or recent trending) stories from a data stream by looking for bursts in frequent occurrences of keywords and phrases (n-grams). Our approach is to break the keyword-clusters detected by the community algorithm into smaller keyword-clusters based on their document timestamps. Thus, we both obtain Episodes (keyword-clusters across several time intervals) and Bursts (keyword-clusters located in a single time interval). A sampling of results comparing Episodes and Bursts indicated that sharply emerging stories in a time span were indeed highlighted better in the Bursts results.

As of the most recent update to the InVID Dashboard, a visualisation of the story burst detection has been added to complement the existing visualisation of story episode detection (stories over time), both forming the Story Graph.

Burst detection has particular significance for most recently emerging stories, since they do not have the same comparative weight in terms of number of documents as stories which have been present in the Twitter stream since a longer time. This means in the classical episodic view those stories take some time to be visible, but in the burst detection view they are more immediately highlighted. This is a contribution to the area of research which is called First Story Detection, i.e. the ability of an algorithm to identify a story from a stream in as early a time as possible after that story first appears.

2.5 Ranking

In the final step, the stories get sorted by the number of documents in each cluster. We are also aggregating the weights of their keyword relations. These both can provide a ranking for the list of stories in the Story View of the InVID Dashboard.

Story results can still be dominated by a small number of persistent stories which are thus repeatedly being ranked higher, as can be seen in the past months with the stories around Donald Trump. This pushes down stories about other subjects which can be just as or more newsworthy but receive less persistent coverage in the news. It does not make sense for us to try to unilaterally decide for a user precisely which story is more or less relevant to them, such as we can not say if a journalist wants or doesn't want to track over time the stories around Donald Trump. However, we can provide means for the user to access different views of the stories, also beyond the distinction between Episodes and Bursts.

We noted the InVID Dashboard contains support for the definition and use of ”Topics”. These allow users to define lists of words and expressions which indicate that a document belongs to a certain topic of discussion, where the domain of the topics can be freely chosen. In InVID, topics can be used to refer to categories of news as used typically in journalism and news reporting. We first used the categories used by the Wikipedia Current Events portal as topics, but realised that these were confusing for the journalists who would eventually use the dashboard. So, after discussion with AFP (a member of the project consortium) we selected the IPTC NewsCodes [6] as our basis for topics, largely aligned with their top level NewsCodes and the pre-defined second and third level NewsCodes by IPTC helping to clarify which stories would belong in which topic (and of course stories may match several topics too). For each topic, a list of regular expressions is defined, based on the labels of the NewsCodes and their synonyms and using regular expressions to support for each alternative spellings, plural forms and disambiguating usage in phrases.

While topics were used in the dashboard to filter the documents and aid browsing, it was easy to observe through the InVID Dashboard how our news topics enabled more stories to be identified. With the topic being used to provide a filtered set of input documents to the clustering algorithm, where a story was dominating in the news coverage the choice of other topics not related to that story would allow to filter out the documents related to that story and uncover other stories despite being the subject, comparatively, of less news coverage. This is illustrated by Fig. 2 which shows the top stories on 14 June 2017, dominated by the London apartment tower fire. However, switching to the Sports topic, as illustrated by Fig. 3, shows how other stories can be easily presented where the topic of the dominant story is filtered out. Figure 4 provides an example of the regular expressions used to define if a document belongs to a topic, in this case Crime, law and justice.

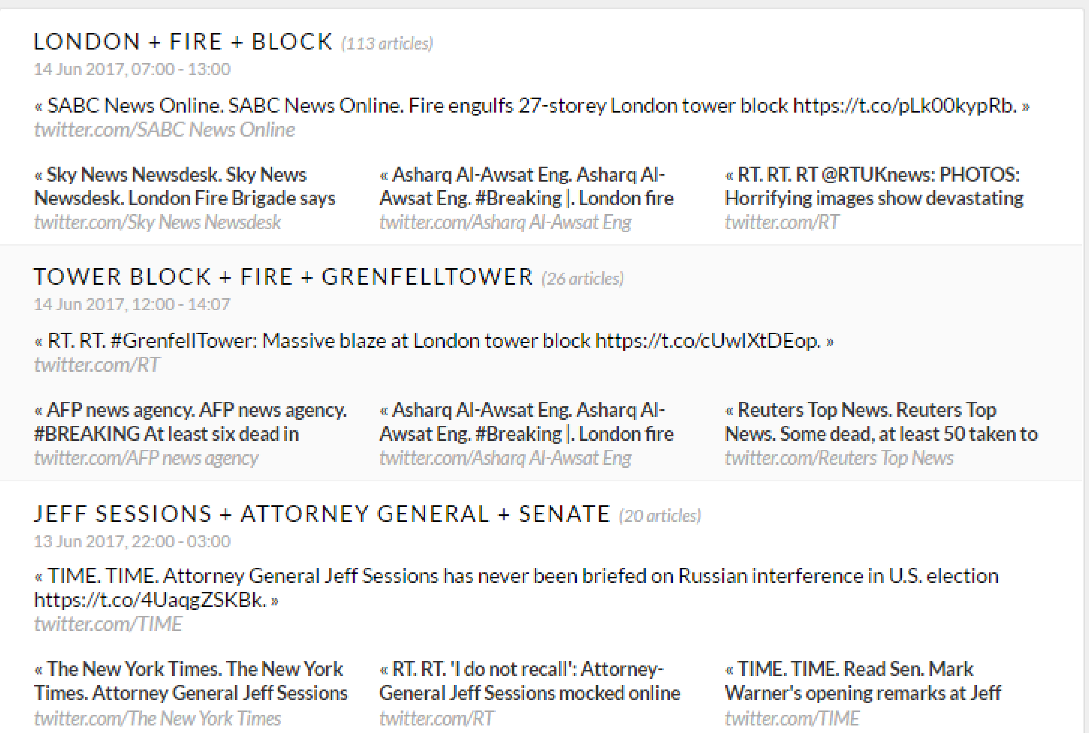
2.6 Story Evaluation Methodology

While in the first six months of InVID we could present only a very first proposal and implementation of story detection, the subsequent 12 months have brought much continual improvement as errors were found, solutions for observable problems tested and updates rolled out on the dashboard. At this stage we consider it important to evaluate the output of the story detection, to provide us with a baseline for the usability of the current version and enable us to subsequently measure quantitatively the improvements made by future updates.

To conduct an evaluation, we need to agree first on the methodology to be followed. The initial implementation was assessed based on an existing ground truth annotated dataset (SNOW 2014) but we observed several issues with the methodology used.

We expect our stories to be news, i.e. they represent events or actions which are valid for reporting through journalists and news organisations. This can be reduced to a metric of ”newsworthiness”. It is defined as a set of stories for a given time slot ”covered in mainstream news sites” (Papadopoulos et al., 2014). However, every news site will have its own criteria for determining what is ”news”, and the validity of the news chosen by one site may be questioned by another site. Experts can be used to assess what is news based on positive and negative examples (Papadopoulos et al., 2014). This then leads to a set of reference stories for a data set for which a story detection algorithm can be evaluated based on recall, as is the case with the SNOW 2014 data set. Wikipedia's Current Events portal can be seen as a crowdsourced version of this, where the crowd decides what is news. It can be observed how this leads to a more diverse set of stories as in expert-based definition, since the latter also relies on agreement between the experts to accept any story to the reference list. The result of either

[6] https://iptc.org/standards/newscodes/



LONDON + FIRE + BLOCK (113 articles)
14 Jun 2017, 07:00 - 13:00

« SABC News Online. SABC News Online. Fire engulfs 27-storey London tower block https://t.co/pLk00kypRb. »
twitter.com/SABC News Online

« Sky News Newsdesk. Sky News Newsdesk. London Fire Brigade says
twitter.com/Sky News Newsdesk

« Asharq Al-Awsat Eng. Asharq Al-Awsat Eng. #Breaking |. London fire
twitter.com/Asharq Al-Awsat Eng

« RT. RT. RT @RTUKnews: PHOTOS: Horrifying images show devastating
twitter.com/RT

TOWER BLOCK + FIRE + GRENFELLTOWER (26 articles)
14 Jun 2017, 12:00 - 14:07

« RT. RT. #GrenfellTower: Massive blaze at London tower block https://t.co/cUwIXtDEop. »
twitter.com/RT

« AFP news agency. AFP news agency. #BREAKING At least six dead in
twitter.com/AFP news agency

« Asharq Al-Awsat Eng. Asharq Al-Awsat Eng. #Breaking |. London fire
twitter.com/Asharq Al-Awsat Eng

« Reuters Top News. Reuters Top News. Some dead, at least 50 taken to
twitter.com/Reuters Top News

JEFF SESSIONS + ATTORNEY GENERAL + SENATE (20 articles)
13 Jun 2017, 22:00 - 03:00

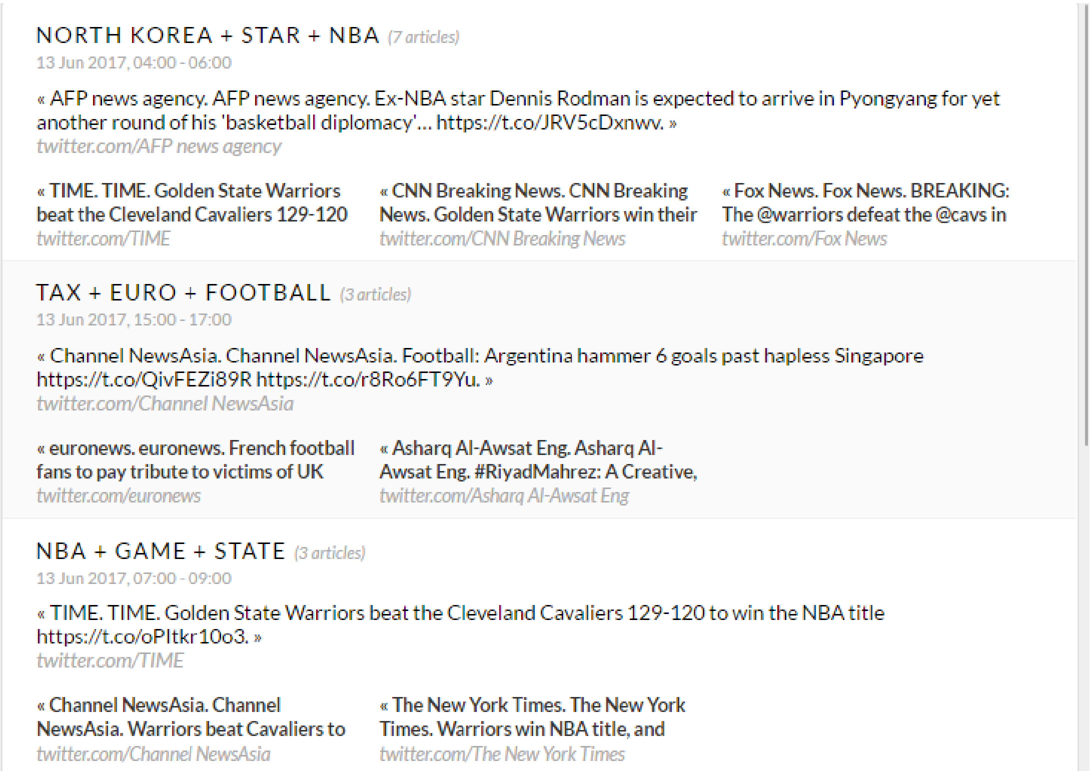
« TIME. TIME. Attorney General Jeff Sessions has never been briefed on Russian interference in U.S. election https://t.co/4UaqgZSKBk. »
twitter.com/TIME

« The New York Times. The New York Times. Attorney General Jeff Sessions
twitter.com/The New York Times

« RT. RT. 'I do not recall': Attorney-General Jeff Sessions mocked online
twitter.com/RT

« TIME. TIME. Read Sen. Mark Warner's opening remarks at Jeff
twitter.com/TIME

Figure 2: Top-3 stories (aggregated) on 14 June 2017.



NORTH KOREA + STAR + NBA (7 articles)
13 Jun 2017, 04:00 - 06:00

« AFP news agency. AFP news agency. Ex-NBA star Dennis Rodman is expected to arrive in Pyongyang for yet another round of his 'basketball diplomacy'… https://t.co/JRV5cDxnwv. »
twitter.com/AFP news agency

« TIME. TIME. Golden State Warriors beat the Cleveland Cavaliers 129-120
twitter.com/TIME

« CNN Breaking News. CNN Breaking News. Golden State Warriors win their
twitter.com/CNN Breaking News

« Fox News. Fox News. BREAKING: The @warriors defeat the @cavs in
twitter.com/Fox News

TAX + EURO + FOOTBALL (3 articles)
13 Jun 2017, 15:00 - 17:00

« Channel NewsAsia. Channel NewsAsia. Football: Argentina hammer 6 goals past hapless Singapore https://t.co/QivFEZi89R https://t.co/r8Ro6FT9Yu. »
twitter.com/Channel NewsAsia

« euronews. euronews. French football fans to pay tribute to victims of UK
twitter.com/euronews

« Asharq Al-Awsat Eng. Asharq Al-Awsat Eng. #RiyadMahrez: A Creative,
twitter.com/Asharq Al-Awsat Eng

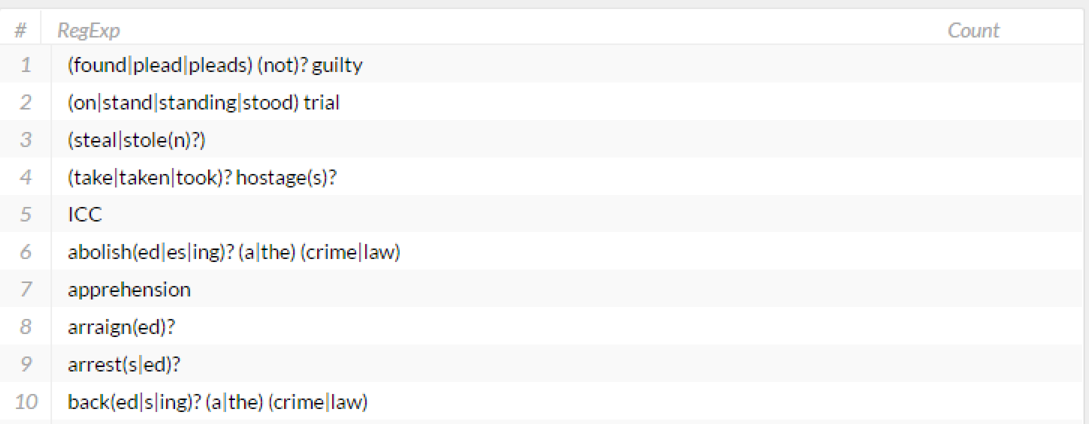
NBA + GAME + STATE (3 articles)
13 Jun 2017, 07:00 - 09:00

« TIME. TIME. Golden State Warriors beat the Cleveland Cavaliers 129-120 to win the NBA title https://t.co/oPltkr10o3. »
twitter.com/TIME

« Channel NewsAsia. Channel NewsAsia. Warriors beat Cavaliers to
twitter.com/Channel NewsAsia

« The New York Times. The New York Times. Warriors win NBA title, and
twitter.com/The New York Times

Figure 3: Top-3 stories in topic "Sports" on 14 June 2017.



<table>
<tr><th>#</th><th>RegExp</th><th>Count</th></tr>
<tr><td>1</td><td>(found|plead|pleads) (not)? guilty</td><td></td></tr>
<tr><td>2</td><td>(on|stand|standing|stood) trial</td><td></td></tr>
<tr><td>3</td><td>(steal|stole(n)?)</td><td></td></tr>
<tr><td>4</td><td>(take|taken|took)? hostage(s)?</td><td></td></tr>
<tr><td>5</td><td>ICC</td><td></td></tr>
<tr><td>6</td><td>abolish(ed|es|ing)? (a|the) (crime|law)</td><td></td></tr>
<tr><td>7</td><td>apprehension</td><td></td></tr>
<tr><td>8</td><td>arraign(ed)?</td><td></td></tr>
<tr><td>9</td><td>arrest(s|ed)?</td><td></td></tr>
<tr><td>10</td><td>back(ed|s|ing)? (a|the) (crime|law)</td><td></td></tr>
</table>

Figure 4: Part of the definition of the topic ”Crime, law and justice”.

approach is that the story detection is evaluated as successful if it can detect the stories CHOSEN by the experts or the crowd. If it detects other stories, it is penalised even if these may also be arguably newsworthy. Another issue is how to compare the stories output by the tool to the stories defined as references in the evaluation dataset, unless the tool has been developed specifically to use the same story model. SNOW 2014, for example, provided a timestamp, descriptive sentence of the story, a set of top keywords and a set of reference documents. It expected tools to produce up to 10 stories per 15-minute time slot. The final evaluation required a set of experts, and given the size of the data to be compared, a sampling approach was taken. Five time slots were chosen and for the up to 50 detected stories, each expert looked for detected stories which matched a part of the set of 59 reference stories. While difficult to scale this up, this means only 5% of the story detection output is actually checked and our implementation works with larger time slots than these (hourly for the bursts, 24 hours is the default time span for the stories in the dashboard).

SNOW 2014 did confirm newsworthy story detection to be a challenging task: F-score: 0.4, Precision: 0.56, Recall: 0.36 (Ifrim et al., 2014). For our evaluation, we want to use the tweets being collected by us in InVID as opposed to a ground truth evaluation data set such as SNOW 2014. This of course raises the question of how we will get our ground truth and a measure for how to compare the results of our algorithm with the ground truth. We choose to evaluate two metrics of our story detection: quality and correctness, yet avoiding the use of reference stories for the main reason that we do not feel it is for us to choose what should be newsworthy for the user. Instead, manual observation of the story results in the dashboard, something we have been doing for the past 12 months, will be formalised to produce quantitative measures for these metrics through observation over a consistent time period.

Correctness of the stories can be measured as a factor of whether a cluster can be unambiguously identified as representing a distinct story (action or event which may be newsworthy). Hence the observer must not decide on the newsworthiness of the story, but simply that the cluster can be seen as representing a potential story. While this needs to be reflected by the (top) documents in the cluster, since we will look at cluster quality below, we will use the story label in this case as the determinant of the story of the cluster. Since different labels (which determines the separation between clusters) may, in fact, be referring to the same news story we add a complementary measure for distinctiveness, i.e. how many of the clusters identifiable as stories actually represent distinct news stories.

Clustering quality can be measured in terms of completeness and homogeneity. Two structural observations can be made on a list of clusters: whether a cluster actually merges two or more stories or whether a story is represented by two or more different clusters. This requires that every document in a cluster is individually marked as belonging in its cluster, in a different cluster, or in no cluster. Completeness refers to the extent to which documents for the same story are in the same cluster. Homogeneity refers to the extent to which clusters only contain documents from the same story.

Hence our methodology for the story evaluation is as follows. We will conduct a ”live” evaluation, i.e. in real time using the story results visible in the InVID Dashboard. It will be a manual exercise, as the human evaluator will make the assessment about the quality of the stories. Values will be drawn through a daily assessment of the stories detected from our Twitter Accounts stream (we will use the same process to compare stories from this stream with the stories from the Twitter News stream, which draws from user-generated tweets claiming to report breaking news). Detected stories will be assessed in terms of ”universal” stories (drawn from the entire Twitter stream) as well as the stories for the top-5 topics in that time period (measured in terms of topic size, i.e. the number of documents currently matching the defined topic). We take only ”top” topics as we observed already that topics with smaller matching document sets are less accurate for story detection, which is a factor of topic size and not related to the correctness of the topic definition. Our focus will be on English results in this evaluation as the volume of French and German tweets is still much lower. In each daily observation, the evaluator will determine the values of the following metrics, based on the (up to) ten stories given in the dashboard:

– Correctness - number of clusters identifiable by their labels as a distinct story / total number of clusters.

– Distinctiveness - number of distinct, individual stories among the correct clusters / total number of correct clusters.

– Homogeneity - average of the individual homogeneity score for each cluster, which is the sum of documents relevant to the stories represented by the cluster divided by the total number of documents (in our case, max 10 sorted by relevance). This measure will not penalise merged stories in that it will count all documents relating to the one or the other story in the cluster as relevant. However, a cluster which merges stories should thus contain documents relevant to each story, so a penalty is applied if a cluster only contains documents relevant to one story but not the other; in such a case we divide the homogeneity score by one plus the number of missing stories (e.g. a cluster merging 2 stories but with all documents only relevant to 1 will result in a homogeneity score of 0.5).

– Completeness - average of the individual completeness score for each distinct story, which is the sum of documents relevant to the distinct story represented by the cluster divided by the total number of documents. This measure will penalize merged stories in that it will only consider documents as relevant which relate to the primary story of the cluster (as the story label of merged stories will likely refer to both stories, we use the top document of the cluster to determine the primary story).

2.7 Story Evaluation Results

We conducted the story evaluation over the period June 19 to June 23, 2017 with Lyndon Nixon (MOD) acting as the evaluation expert. Each afternoon in these five days we checked the story results in the InVID Dashboard, considering the stories listed for the Twitter News feed (user tweets) as well as Twitter Accounts feed (professional news channels), and for the latter we additionally checked the stories for each of the top-5 news topics of that day. For each story, we evaluated the label (does it meaningfully refer to a news story? does it reflect a single story or multiple stories?) and the documents presented for that story (do they relate to the story represented by its label?). From this, as outlined in the previous section, we could combine our insights into 4 metrics which we can compare across sources and days:

– Correctness: are the generated clusters correctly related to newsworthy stories?

– Distinctiveness: does each individual cluster precisely relate to an individual story?

– Homogeneity: are the documents in the cluster only relevant to the newsworthy stories represented by the cluster?

– Completeness: are the documents in the cluster relevant to a single, distinct news story?

Individual lists of the stories detected each day in the Twitter Accounts (aggregated), Twitter News (aggregated) and the top-5 news topics (for Twitter Accounts) are shown in Appendix A alongside (from left to right by column) the definition of whether the story (label) was identifiable as a news story (Y), whether the story was distinct (through numbering from one onwards), the number of documents returned for that story (with a maximum of ten considered), the number of those documents which were relevant to the story, the number of those documents which were relevant to the other story (where the label merged details of two or even three stories), the remaining number of documents which were not relevant to any story, the calculated homogeneity score for the story and the calculated completeness score for the story.

Since it is used as a basis for the query construction for the social media extraction we look first at the aggregated Twitter Accounts stories. Here, the average of the values over the five days is:

- Correctness: 0.895 (The 5 non-stories out of the total 46 were all generated from the tweets of one news organisation, we found it was tweeting very often and many tweets were commentary rather than news stories so we will remove this account for the next phase of story detection.)
- Distinctiveness: 0.598 (Values varied from 0.44 to 0.71 showing this is still an issue of concern. It penalises both merged stories and split stories, both of which occurred daily.)
- Homogeneity: 0.93 (It is confirmed that the documents are nearly always relevant to the story. The few irrelevant documents that were found were always related to a story label being a merge of two stories, but documents from one of those stories were missing or a story label related to one story, but documents from another story were present.)
- Completeness: 0.93 (While the values varied slightly with homogeneity, the average is the same. This highlights that most clusters were indeed single stories with relevant documents. Twice completeness scored lower than homogeneity due to single clusters referring to two stories and containing documents from both, since completeness penalises this.)

We are encouraged that the current approach already approaches 100% for clustering documents into newsworthy stories (it seems, if we removed the one Twitter account that generated the non-stories, we would have scored 100% here), which reflects the quality of the data source chosen for the task - Twitter accounts of professional news channels. We can compare this with the correctness score achieved from the Twitter News feed which takes user tweets of ”breaking news”. Here, we average a correctness of 0.76 which reflects the expected lower data quality of user tweets. It should be noted that the stories detected from this feed largely consisted of two types: firstly, only some of the significant news from the professional feeds was also being discussed in volume by users, e.g. the Otto Warmbier North Korea story was detected over two days in the Twitter News feed while the Saudi Arabia new crown prince story did not occur at all, despite it leading the Twitter Accounts stories for two days. Thus, the Twitter News feed offers an interesting insight into what news the wider Twitter user community is discussing (or ignoring). Secondly, some stories were detected in this feed that we did not detect at all using the professional news feed, showing that the general public were sometimes more interested in other news than what the professional news channels were reporting. This occurred on Wed. 21 June, where 2 of the top-3 clusters extracted for Twitter News were a single news story: ’Two elephants work together to save a baby elephant from drowning at Grand Park Zoo in Seoul, South Korea’.

We also observed that using the topics defined in the dashboard allowed us to find many more stories distinct to the ’top-10’ from the aggregated view. Just as with the aggregated stories, correctness nears to 1 but distinctiveness is consistently lower. While there are merged stories, the most common issue is split stories, i.e. stories with a higher volume of tweet reporting tend to diverge more with respect to the words used to report them which in turn leads to a larger set of distinct keywords being extracted. As merging clusters is based on the overlap between the keyword sets that define them, there is still a point where the keyword variation is too high to affect a merge. The described plans to align keywords to entities should be a first step towards solving this issue. Similarly, homogeneity and completeness scores are consistently high across the topics and across all five days.

The evaluation shows that our story detection already performs very well, while focusing our future work on a better disambiguation of clusters into distinct, individual stories.

3 Social Media Extraction and Annotation

As outlined in D2.1 and whose implementation into the InVID Platform has been detailed in D5.2, we set up a social media extraction pipeline which is configurable and extendible to support additional sources. The initial pipeline supports YouTube, DailyMotion and Vimeo APIs. Each component, as implemented for the platform, is called a ”social media mirror” (previously, other mirrors already existed in the platform but were focused on retrieval of textual documents). A list of queries is generated out of the Twitter Accounts stream and consists of the most frequently occurring keyword pairs out of the documents in that stream over the past 24 hours. We take the most frequent 10 keywords and for each keyword the top-10 co-occurring keywords, thus generating 100 keyword pairs which are then used to perform conjunctive queries over each API at a 6 hourly interval. Since the list of keyword pairs is updated at each interval, we refer to this as the ”dynamic input filter” (previously in the platform, social media mirrors were set up with fixed, unchanging lists of terms for creating the queries at a regular interval, which was known as the static input filter). Currently this results in the collection of 700 - 1100 videos daily, as queries are time-restricted (to only videos uploaded in the previous 24 hours) and hence some queries do not find matching video content as a result of the keywords being paired (either because they are irrelevant or not used in a video posted in the past 24 hours). YouTube has tended to be more effective in matching queries with videos (and the retrieved videos are more relevant), whereas retrieval from Vimeo and DailyMotion has tended to provide a lower volume of documents.

3.1 Sources

Following the launch of the initial video collection pipeline with the set of new social media mirrors for YouTube, Vimeo and DailyMotion and confirmation of their stability when integrated with the platform, we prioritized the new video sources we should add. During this period, Instagram changed its terms and conditions [7] and seriously restricted third party access to its API, with all applications which do not comply with the new terms and conditions by June 1st 2016 being banned from future API access. Since that change included any retrieval from the full Instagram feed it became unfeasible for us to include Instagram any more in our sources. Hence the next priority source was determined to be Twitter, which provides both tweets with native videos and tweets which embed videos from the platforms Periscope (a live video streaming platform, where live streams can be replayed up to 24 hours after the broadcast ended; since May 2016, Periscipe broadcasts became permanent by default), Amplify (Twitter's video ad platform) and Vine (since Jan 17, 2017 the original, independent video platform Vine has been shut down and relaunched as Vine Camera, an app which creates short, looping video where online sharing is only possible via Twitter).

The microblog service **Twitter** has been supporting sharing *native videos* on its platform [8], which are native videos posted directly in a tweet via a Twitter application and not as a link to a video on an external service (e.g. YouTube). While we use Twitter's streaming API [9] to detect breaking news, the streaming API does not support filtering for tweets containing native video. To retrieve video posted on Twitter, we access the Search REST API [10] instead. In order to obtain only tweets containing native video, we use the query operator ”filter” set to the value ”native_video” and to restrict the results to a single language, we use the ”lang” query operator.

We use the dynamic input filters to obtain the search terms for the search queries. We periodically (once every hour) search for native videos, with the search terms being updated when the dynamic input filter gets updated. To prevent duplicate results, the time of the previous search is used as a temporal query operator (`since`). In order to minimize the requests against the API endpoint, we execute several queries in one API call using the `OR` query operator and allowing the responses to contain 100 tweets (the maximum supported by the API). Since twitter allows only 10 query operators in total, we limit our API calls to at most 8 queries each time. If too many requests to the API are made too fast, the API responds with instructions to wait. If the API instructs us to wait for more than 20 minutes, the retrieved tweets are returned but our code does not wait to prevent the pipeline getting stuck in waiting mode. From the response, we filter out retweets using the `-filter:retweets` query operator.

The response from the search API includes an entry that contains information about embedded media. For native video, the aspect ratio, the duration in milliseconds and a set of *variants* are provided, varying in content type (video encoding format), resolution and bitrate. Each variant points to an URL where this video can be accessed. For InVID, we select the video variant of content type `mp4` and with the highest bitrate and map the URL to the `ma:locator` attribute in our internal XML document structure.

With the described setup, we are obtaining around 600 tweets with news-relevant video content in a 24 hour period.

3.2 Queries

With the implementation of specific and distinct topics we can now explore the news stream according to each topic. The dynamic input filter is populated by the keywords and associations aggregated from the entire news stream, and we observed that the keyword lists, when a news story was dominating the global news coverage, tended to be dominated by multiple keywords associated with the dominant story. This meant that, while many more stories were being detected from the Twitter stream, video retrieval was tending to be based on queries relating to a more narrow set of more dominant stories. However, we could consider to take the top keywords and associations from each topic to generate a

[7] https://www.theverge.com/2015/11/17/9751574/instagram-app-developers-api-restrictions-security-privacy

[8] https://support.twitter.com/articles/20172128

[9] https://dev.twitter.com/streaming/overview

[10] https://dev.twitter.com/rest/public/search

dynamic input filter which would query for a wider range of stories. Since there are eleven topics defined, this could mean eleven times more queries, unless we reduce the time interval between queries, which is undesirable as we want to keep social media retrieval for news content as timely as feasible. The other option we consider is to use, as an alternative to the keyword-association pairs, the labels of the detected stories. These currently consist of the three most frequent keywords in the cluster representing a story. While a triple term query may reasonably retrieve fewer matching documents than a term pair, the actual volume and relevance of retrieved documents will depend on the quality of the story labelling, just as the current approach depends on the quality of the keywords and their associations. However, we had already observed in where topics were smaller (containing a smaller number of documents) the relevance (or generality) of the keywords was dropping quite substantially early in the list, which would of course lead to queries that were either not matching any videos or retrieving only irrelevant video. On the other hand, (correct) story detection should ensure that only stories exist where one can be determined (by a sufficient number of documents sharing relevant keywords), and thus labelling should lead to queries which are at least more specifically designed to retrieve video relevant to a distinct and recent news story. Since the labels of stories would be taken from each topic (where each topic may produce up to ten stories) to feed the dynamic input filter, the result should be a set of queries that collect video concerning a wider range of news stories. To test this hypothesis, we conducted an evaluation of both the current and the proposed social media queries.

3.3 Evaluation

Precision and recall are the standard measures for information retrieval. To conduct an evaluation on social media information retrieval, we can not use classical recall measures since we can not say what is the total number of relevant documents on any social media platform at any time for one query. Precision can be calculated based on the set of results retrieved for each query, where the maximum number of results evaluated will be capped at 20 documents (which is also the first page of video results on YouTube). It is necessary to cap the maximum number of results to evaluate as it would not be feasible to automate the determination of whether results of a query for newsworthy video were newsworthy or not, since newsworthiness is a subjective determination that can only be made by a human evaluator. On the other hand, we should consider whether success in information retrieval only occurs if and only if the retrieved video is relevant to the story represented by the query, or if any newsworthy video being retrieved can be considered a metric for success. Indeed, whereas the timeliness of the story detection is important to ensure stories are detected as they emerge and queries are made for relevant video at the moment the news story is still newsworthy, in the video retrieval videos will continue to be posted for a news story for a longer time after the news story initially occurred and those retrieved videos can still be relevant for discovery and verification in the InVID context. Thus, queries which reference keywords that persist in the news discussion (such as Donald Trump) are likely to return other videos which are not relevant to the current story but still reference an earlier newsworthy story. Since while our precision measure can indicate how many videos in our results are relevant to the query itself, low precision may hide the fact we still collect a high proportion of newsworthy video content. Still, precision may act as an evaluation of the quality of our query to collect media for the specific story. On the other hand, we choose a second precision measure, which we will call “accuracy”, which measures the proportion of all newsworthy video returned for a query. Since this measure will include video not directly related to the story being queried for, this acts as an evaluation of the appropriateness of our query to collect newsworthy media generally. Finally, we will include a recall measure which is defined as the proportion of newsworthy video retrieved which is relevant to the story being queried for, ergo our recall is the precision divided by the accuracy and acts as a measure of the specificity of our query for the news story. For comparative evaluation, we would of course prioritize higher precision for our queries, but also since social media retrieval will invariably involve the possibility of false positives (due to the varying relevance of user provided titles and descriptions which determine the results of a query), we will also welcome higher accuracy (once we accept the existence of false positives in the results, we desire to minimize the extent of completely irrelevant video that may be collected as a result). Finally, to acquire a final metric to measure the overall accuracy of the information retrieval we calculate the F-Score, classically seen as the harmonic mean of precision and recall, based in our case on the precision and accuracy values (since we want to prioritize the precision but also reflect a more useful result in retrieving other newsworthy video while minimizing fully irrelevant video, as measured by accuracy).

Until now, we have adapted classical measures for information retrieval, which can indicate to us if our queries are effective in finding (a) video about a specific news story and (b) video relevant to the news generally. However, none of these measures indicates if we achieve a further goal, which is to collect a broader range of video material related to the news. To do this, we need to annotate each query with the story it is querying for. As there is no official classification of news stories each day with which we could annotate queries, we will take a simpler approach. The first query will be annotated as belonging to a story S1. The subsequent query, if we determine it to be querying on the same story, will also be annotated with S1. If it is querying on a different story, we will annotate it with a new story S2. Ongoing, every new story that is being queried will receive a new identifier, following a standard numbering order. As a result, we may conclude with 2 further measures. Story breadth is the sum of unique stories queried for within the set of queries being evaluated. Story depth is the measure of the extent retrieved relevant videos are distributed across distinct stories, i.e. we want to reward a higher average precision value across different stories as opposed to having higher precision for only some stories while other stories suffer from low precision in video retrieval. Since this is a factor of uniform distribution of the precision across stories, we calculate story depth as the mean of the precision of video retrieval across stories (which is itself the average of the precision of video retrieval across all queries related to a story) multiplied by one minus the standard deviation (of the values for average precision for each story). Hence a more uniform distribution for precision of retrieval across stories will tend towards a value closer to the overall average precision of the queries as a whole (where standard deviation is zero, story depth = overall average precision). However, as the distribution becomes more non-uniform (some stories have higher precision and some have lower precision, so the relevant results start to become skewed towards having more content for a smaller number of stories than what was queried for), standard deviation will increase and the story depth will drop.

For manual evaluation, we will look at two one day's results from the Twitter Accounts stream. We can examine both the current query constructions (keyword-association pairs) and the potential query constructions (story labels) in the dashboard, manually making the queries on the YouTube API (DailyMotion and Vimeo APIs generally return less content and less relevant content in any case) to check the list of videos returned. We choose two separate days in order to attempt to consider one day where news coverage has been quite generally spread (we choose the results from the day of writing which is 12/13 June 2017, as there is no single dominating story in the aggregated news) and one day where news coverage has been skewed towards one larger news story (we choose 9/10 May 2017, which covers the firing of FBI director James Comey). Since the full data set would be too much for a manual evaluation (100 queries for the current dynamic input filter, up to 110 queries for the potential story-based querying) we choose a number of 25 queries for the comparison of each approach: for the current approach we take the top-5 keywords and their top-5 associations; for the potential approach we take the top-5 topics (by size so the aggregated stories from all tweets as default and then the next 4 topics in order) and their top-5 stories.

We present the results of the evaluation here in terms of the overall metrics (see the Appendix B for the list of queries and individual metrics for each query), with decimals rounded to 2 places:

– 13Jun-current: the results of social media retrieval using the current keyword based approach for content from 13 June 2017;

– 13Jun-proposed: the results of social media retrieval using the proposed story based approach for content from 13 June 2017;

– 10May-current: the results of social media retrieval using the current keyword based approach for content from 10 May 2017;

– 10May-proposed: the results of social media retrieval using the proposed story based approach for content from 10 May 2017.

Looking at the results from a news day where there was a mix of stories in the news reporting (13 June 2017), we observe that the proposed approach achieves a wider breadth of news stories, querying for 18 distinct stories within the top-25 queries whereas the current approach queried for 9 distinct stories. More commonly tweeted stories tend to generate more individual queries in the current approach, e.g. the story on the tornado warning in different US counties had 5 different queries in the current approach while was the subject of 1 specific query in the proposed approach. The average precision for the current approaches' 5 different queries was 0.16 compared to a precision of 0.1 for the specific query in the proposed approach, suggesting one does not retrieve significantly fewer documents for the story with less queries while of course achieving a greater breadth of news story coverage in the total retrieved document set. In fact, while average precision was higher for the proposed approach average accuracy was almost the same, suggesting the same quantity of newsworthy documents may be retrieved by the proposed approach with a higher proportion of them being relevant to the current news stories (recall of the proposed approach was 0.64 compared to 0.42 for the current approach). Similarly, the F-Score for the proposed approach was 0.59 compared to 0.425 for the current approach. The remaining issue for the proposed approach, with its greater breadth of news coverage in the retrieved document set, would be that having less distinct stories in the set could mean having better coverage of those stories in the documents. Our story depth measure for how many specifically relevant documents are retrieved for each of the stories suggests that this is not an issue, as while double as many distinct news stories are covered in the retrieved documents the story depth is 0.30 compared to a value for the current approach of 0.17.

By considering a second news day we can check if the two approaches compare equally when the news context is different. We have observed through the dashboard that the global news coverage being collected in the Twitter Accounts stream does mean that days in which a major news story occurs (with global significance) lead to that story being referred to in the twitter stream at a significantly higher frequency than any other story (which we have termed the ”dominant” story). This makes sense: whereas stories of regional interest will only be reported by a subset of our Twitter news accounts, global stories will be covered by potentially all of those accounts leading to a significant difference in volume of documents. Since such ”dominant news story days” are an unavoidable aspect of daily news detection, we took the day of James Comey being fired as FBI director by President Donald Trump (10 May 2017). As expected, our keyword based queries almost all relate to this story, since there are multiple top keywords for the same story (trump, fbi, james, house, director, chief). Indeed, 24 of the top-25 keyword pairs for queries were about this story, only ”president + korea” referring to another story on that day, the election of Moon Jae-in as the new president of South Korea. The dominant story also generates a lot of video content on the video platforms on that day, as a response to global discussion and reaction to the event, so these queries also show a high average precision. While our story breadth is just 2, our average precision is 0.89. Accuracy is only slightly higher but almost perfect at 0.97; with most retrieved documents relating specifically to the dominant story recall is at 0.91. While there is only one other story to retrieve documents for in the top-25 queries, and that is due to 1 query, it’s precision was also high (0.85) and thus story depth (the extent to which all stories in the query list are represented by retrieving relevant documents) stands at 0.85. These high values seem to communicate that this has been a highly effective approach to news video collection but again we must remind ourselves that only 2 different stories are present in that collection, and 96% of the collected and relevant video is about a single story.

In the proposed approach for this news day, there is a significant difference in story breadth. Now 15 stories are distinctly queried for in the top-25 queries. The lead story is the same dominant story as in the current approach, FBI director James Comey being fired. Three queries are made for this story with an average precision of 0.82 (due to the third story having an irrelevant keyword in its label, otherwise precision was at 1 for the queries based on story labels for this story). Compared to the current approaches precision of 0.89 for this story, this is not much less. Potentially much more video could be collected for the story in the current approach (retrieved by 24 different queries as opposed to 3) - however we have not considered here how much duplication of content occurs in subsequent queries targeting the same story. The number of unique video documents retrieved by both approaches may not differ as much. Overall average precision is 0.52, reflecting that some story queries are very effective and some are not (e.g. ”attack + court + brisbane” which references very well a news story current at that time - a man standing trial in court for an attack in Brisbane, Australia - but failed to return any relevant video documents). Accuracy averages at 0.69 and our recall is 0.59. Finally, story depth is measured as 0.35. The figures are all lower than the current approach but we need to ask if we prefer 85% story depth for 2 stories on the day or 35% story depth for 15 stories, ensuring a much broader choice of video content for news on the platform. Comparing the 10 May proposed approach results with the current approach when it is also querying for a broader range of stories (the 13 June data), we can note that the proposed approach has higher precision (0.52 to 0.36), recall (0.59 to 0.42) and story depth (0.35 to 0.17). This suggests that the proposed approach provides more accurate queries for the stories and hence performs better on retrieving relevant documents across all stories.

We can also consider how the two approaches performed comparatively on both days. It is clear that the current approach is strongly influenced by dominant stories, significantly reducing the story breadth in the collected documents on such days. The proposed approach performed, on the other hand, very similarly on both days despite the clear difference in the distribution of tweets about news stories. It is possible that as story breadth reduces, the current approach performs better on the evaluation metrics

Metric	value
avg precision	0.36
avg accuracy	0.84
avg recall	0.42
f-score	0.425
story breadth	9
story depth	0.17

Table 3: Results 13Jun-current.

Metric	value
avg precision	0.54
avg accuracy	0.82
avg recall	0.64
f-score	0.59
story breadth	18
story depth	0.3

Table 4: Results 13Jun-proposed.

(precision, accuracy, story depth) but indeed at the cost that the collected documents cover a smaller number of news stories. However, the proposed approach collects documents from a broader range of stories, and while precision and story depth may then be lower they perform better than when the current approach should be equally broader in its collection and appear that they should be more stable over time regardless of how tweets about news stories are distributed day by day; a consistent precision of around 0.5 and story depth of around 0.3 would indicate that the document collection in the proposed approach would be much more balanced across all the stories.

3.4 Extending the Metadata Model

The document metadata model extensions for the webLyzard platform are presented in Table 7. These extensions have been validated by their usage in the metadata produced by the social media extraction pipeline (now currently mapped from four different video APIs), which is stored in the platform and used in query and retrieval processes (find and browse the video in the InVID Dashboard).

To support a common understanding of the InVID document metadata model (considering also the possibility to return metadata properties via the Platform API) we have aligned our metadata properties to the properties defined in the following commonly used semantic media standards/recommendations:

- ma: prefix - W3C Ontology for Media Resources; https://dev.w3.org/2008/video/mediaann/mediaont-1.0/mediaont-1.0.html. Core set of metadata properties for media resources, along with their mappings to elements from a set of existing metadata formats.
- no prefix - Schema.org; http://schema.org. Schema.org is a collaborative, community activity with a mission to create, maintain, and promote schemas for structured data on the Internet, on web pages, in email messages, and beyond.
- foaf: prefix - Friend of a Friend (FOAF); http://xmlns.com/foaf/spec/. Linking people and information using social networks.

Metric	value
avg precision	0.89
avg accuracy	0.97
avg recall	0.91
f-score	0.92
story breadth	2
story depth	0.85

Table 5: Results 10May-current.

Metric	value
avg precision	0.52
avg accuracy	0.69
avg recall	0.59
f-score	0.57
story breadth	15
story depth	0.35

Table 6: Results 10May-proposed.

InVID attribute	Description	Data type	Sample values
media_type	key attribute to filter media content controlled list	string	video image
title	title of the document	string	”Tweet by Doprava v Praze”
text	text of the document	string	”Sample text”
url	link to the document	url	”twitter.com/doprava_v_praze/status/715419623450144768”
date	timestamp of the document creation	unix timestamp	”1462310408000”
keyword	list of keywords	array	[{’name’: ’keyw1’}]
duration	duration of the video in seconds	unsigned long	609
media_license	empty or a Creative Commons URL	string	empty https://creativecommons.org/licenses/by/4.0 https://creativecommons.org/licenses/by-sa/4.0 https://creativecommons.org/licenses/by-nd/4.0 https://creativecommons.org/licenses/by-nc/4.0 https://creativecommons.org/licenses/by-nc-sa/4.0 http://creativecommons.org/licenses/by-nc-nd/4.0/
thumbnail	link to a thumbnail image for the video	string	”https://i.vimeocdn.com/portrait/6659936_30x30.jpg?r=pad”
viewcount	number of video views	unsigned long	201008
comments	number of comments	unsigned long	71
likes	user rating for the video, number of likes	unsigned long	3488
user_id	platform specific identifier of the user	string	”x24vth”
user_name	user name	string	”iTELE”
media_url	url link to the actual image or video file	string	”https://video.twimg.com/ext_tw_video/560070131976392705/ pu/vid/320x180/nXXsvs7vOhcMivwl.mp4”

Table 7: InVID document model attributes.

– sioc: prefix - SIOC; `http://rdfs.org/sioc/spec/`. Modelling Social Media sites main concepts and properties required to describe information from online communities (e.g message boards, wikis, weblogs, etc.) on the Semantic Web.

– dc: prefix - DCMI Metadata Terms; `http://dublincore.org/documents/dcmi-terms/`. Fifteen terms of the Dublin Core Metadata Element Set, as well as the qualified terms.

The mappings are presented in Table 8. It also indicates how different types of metainformation for a media document are defined in different metadata models, thus justifying the need for a combined InVID metamodel for expressing this information for the retrieval and verification processes.

Looking ahead, considering future use cases for the metadata and its use in the dashboard, we propose two further metadata properties we plan to introduce in the next period:

InVID attribute	Schema.org mapping	MediaOnt mapping	Other mapping
media_type	fileFormat	ma:format	dc:format
duration	duration	ma:duration	dc:extend
media_license	license	ma:hasPolicy	dc:license
thumbnail	thumbnailUrl	ma:hasRelatedImage	
viewcount	interactiveStatistic		sioc:num_views
comments	commentCount		sioc:num_replies
likes			
user_id		ma:hasPublisher	
user_name			foaf:accountName
media_url	url	ma:locator	dc:source

Table 8: InVID document model mappings.

– **Reach** - individual social media metrics can be directly acquired from the social network APIs with the need of course to update when necessary as they change over time; however, absolute values for each social media document can not help immediately in understanding which documents are - compared to others around the same story - being shared or engaged with more. We propose a normalised measure of reach which provides a value in the range 0...1 for each document, where that value is the relative rate of distribution of the document in the social network compared to all other documents in the same social network relating to the same news story. Therefore sorting a set of documents in a story by reach, would help journalists to more easily see which documents are currently spreading at a greater rate within social networks and potentially thus should be prioritized for verification.

– **Authoritativeness** - another social media measure that we plan to provide for each document is that of its authoritativeness. While this is a more subjective measure to be determined together with the partners working on contextual verification in InVID, we recognize its important in enabling journalists to more easily find the documents which should rather be subjected to verification. We have observed in the dashboard that we currently retrieve from the video APIs a mix of video material coming from both official news channels and user uploads (UGC, or User Generated Content). It is not easily possible to separate these sources at the query stage (only DailyMotion API has a ”ugc” flag on content, meaning content which is not from an official DailyMotion partner). We also need to consider that many user uploads now re-upload (parts of) official news video for a story. Regardless, a user seeking to determine the official details of a news story may appreciate ordering retrieved documents by the most authoritative first, while a user seeking to verify unofficial claims for a news story may order the same documents by the least authoritative first. We will consider the authoritativeness measure as a product of two metrics to be calculated in the platform: account properties and content properties. Account properties will combine social network analyses - both the characteristics of social network accounts (e.g. Twitter number of followers/number followed ratio) and the graph of the accounts interacting with that content in the social network (e.g. which account is the content shared from, which accounts have shared this user's post). Content properties will use extracted media features (for video documents) or textual features by the InVID verification services to compare the content's features with the features most commonly associated with authoritative or non-authoritative content (e.g. logo detection can indicate if video is coming from an official news source which is widely accepted as authoritative or a propaganda news source which is generally regarded as non-authoritative; the contextual verification service identifies claims of fake or falseness in comments on social media posts).

4 Video Annotation

4.1 Video Fragmentation and Thumbnail Extraction

4.1.1 Motivation and foreseen use

A core operation for many video analysis applications, such as video annotation, indexing and summarization, is the identification of the temporal structure of the video. For edited (i.e. professional) videos this typically corresponds to the detection of the video shots (i.e. sequences of frames captured uninterruptedly by a single camera) through a shot boundary detection approach, such as the one described in (Apostolidis & Mezaris, 2014). However, when dealing with user-generated videos (UGV) the shot-level fragmentation is a coarse one and does not reveal too much information about the structure of the video, due to the fact that these videos are most commonly captured without interruption with the help of a single camera/smartphone, thus being single-shot videos.

Based on this condition, in InVID we tried to develop a method capable of defining visually discrete parts of a UGV (called sub-shots subsequently in this chapter) and use these fragments for representing the structure of the video. To this end we investigated two different approaches. The first approach segments a single-shot video based on the detection of a set of typical video recording activities, such as horizontal and vertical camera movement (also known as camera pan and tilt) and camera zoom in/out, which result to the alteration of the visual content. The second approach applies temporal fragmentation of the video by indicating visually dissimilar segments of it after assessing the visual similarity of neighboring frames of the video with the help of a global descriptor. In each case, a number of representative keyframes is extracted for each video fragment, while a smaller subset of these keyframes is selected through an image clustering process and used as thumbnails of the entire video.

The foreseen use of this analysis component is to process the videos ingested in the InVID platform from the InVID Dashboard, the InVID Verification Application and the InVID Mobile Application. This analysis will create metadata about the video structure and will extract a set of representative keyframes and thumbnails. The produced data will be ingested in the InVID platform and exploited by the developed software components for concept-based video labeling, logo detection and near-duplicate detection, thus enabling the extraction of additional metadata concerning the semantics and the visual content of the video.

4.1.2 State of the art

A variety of different methods have been proposed over the last couple of decades, dealing with the temporal segmentation of single-shot videos. Most of them are related to approaches for video summarization and keyframe selection (e.g. (Kelm, Schmiedeke, & Sikora, 2009), (Cooray, Bredin, Xu, & O'Connor, 2009), (Mei, Tang, Tang, & Hua, 2013), (González-Díaz, Martínez-Cortés, Gallardo-Antolín, & Díaz-de María, 2015)), some focus on the analysis of egocentric or wearable videos (e.g. (Lu & Grauman, 2013), (Xu et al., 2015), (Karaman et al., 2014)), others aim to address the need for detecting duplicates of videos (e.g. (Chu, Chuang, & Yu, 2010)), a number of them is related to the indexing and annotation of personal videos (e.g. (Luo, Papin, & Costello, 2009)), while there is a group of methods that targeted the indexing and summarization of rushes video (e.g. (Dumont et al., 2008), (Liu, Liu, Ren, & Chan, 2008), (Bai, Hu, Lao, Smeaton, & O'Connor, 2010), (Pan, Chuang, & Hsu, 2007)).

Regarding the suggested methodology for partitioning a single-shot video or the shots of an edited, multi-shot video into sub-shots, two basic directions have been introduced. The techniques of the first direction are driven by the observation that a sub-shot is an uninterrupted sequence of frames within a shot only having a small variation in visual content (Petersohn, 2009), and they try to define sub-shots by assessing the visual similarity and coherency of groups of consecutive video frames. Within this context the method of (Pan et al., 2007) applies a straightforward approach which relies on the evaluation of the similiary between pairs of frames with the help of color histograms and the x^2 test. The algorithm selects the first frame F_a of a shot as the base frame and compares it sequentially with the following ones until some frame F_b is different enough. Then, frames between F_a and F_b form a sub-shot, while F_b is used as the next base frame in a process that is repeated until all frames of a shot have been processed. In (Dumont et al., 2008) sub-shot segmentation is performed by using a sliding window over video frames centered on a current frame. The distance (i.e. visual dissimilarity) between a pair of frames is then computed with the help of 16-bin HSV histograms. For each frame the algorithm computes its similarity with the previous and the following one, and compares the ranking of pre-frames and post-frames. Finally, a sub-shot boundary is defined when the number of top ranked pre-frames is greater than a predefined threshold. The technique presented in (Bai et al., 2010) splits each frame into 8x8 pixel grids and calculates the mean and variance of RGB color in each grid, while the Euclidean distance is used to measure the difference between neighboring frames. Based on the fact that the cumulative frame difference shows gradual change in a sub-shot, the algorithm indicates sub-shot boundaries by identifying high curvature points within the curve of cumulative frame differences. A similar approach which sub-divides each image in a 4x4 grid and computes the difference and cumulative difference of local 16-bin color histogram between consecutive frames was described in (Liu et al., 2008). Another method that estimates the brightness, contrast, camera motion and object motion of each video frame with the help of YUV histograms and optical flow vectors was discussed in (Ojutkangas, Peltola, & Järvinen, 2012). Sub-shot boundaries are then detected by applying a coherence discontinuity detection mechanism on the set of extracted features with the help of a moving window of size R, that considers the feature values of two consecutive frames and the average of these values in a number of previous and following frames. Finally, two other histogram-based approaches were introduced in (Lei, Xie, & Yan, 2014) and (Petersohn, 2009).

The second class of techniques detect a video sub-shot based on the concept that each individual sub-shot corresponds to a different activity of the camera during the recording of the video. So, these methods try to identify the sub-shots of the video by detecting the camera motion over sequences of video frames. An early approach introducing camera motion analysis for indexing MPEG videos was presented in (J.-G. Kim, Chang, Kim, & Kim, 2000). It fits the motion vectors in the MPEG stream into the 2D affine model to detect basic camera operations automatically. Then, sub-shot segmentation is defined based on the homogeneity of camera motion in each unit. This algorithm was also utilized in (Kang & Hua, 2005) which describes an approach for assessing the representativeness of video frames, and in (Tang, Mei, & Hua, 2009) which addresses the problem of video summarization. Another method that exploits the motion vectors of the MPEG-2 video stream and estimates the optical flow and the type of conducted global camera motion via a multiresolution scheme that uses the weighted least square method (Tukey estimator (Odobez & Bouthemy, 1995)), was presented in (Durik & Benois-Pineau, 2001). An extension of this method, which utilizes the affine model for motion characterization, was described in (Krämer & Benois-Pineau, 2005) and used in (Karaman et al., 2010). A different approach that segments a video into sub-shots by evaluating the change in dominant camera motion with the help of local descriptors and the affine transformation was introduced in (Cooray et al., 2009). The algorithm estimates this change based on the analysis of the dominant-motion transformation matrix between two consecutive frames in the video. This matrix is the 3x3 affine transformation matrix M created after extracting and matching the SIFT descriptors of these frames. Using matrix M, the algorithm is able to classify, for each frame, the corresponding camera motion into the following categories: pan left/ right, tilt up/down and zoom in/out. A similar approach was presented in (Nitta & Babaguchi, 2013), while another algorithm that combines the motion analysis of (J.-G. Kim et al., 2000) with a method that computes the affine model parameters was described in (Mei et al., 2013). In the same direction, the technique of (Cooray, Lee, & O'Connor, 2011) computes the optical flow field for each pair of consecutive frames. Then, the best transformation between each pair of frames is computed by fitting the extracted motion vector fields to a 2D affine model and estimating the homography with the help of the RANSAC algorithm. As in other works, e.g. (Cooray et al., 2009), the algorithm compares the values of the affine model parameters with a set of experimentally determined thresholds in order to detect and classify the global motion for each frame of the video, and a final filtering step is applied to all classified frames to determine the type of sub-shots present in the video. Contrary to the use of experimentally determined thresholds for identifying the type of the detected camera motion, the method of (Abdollahian, Taskiran, Pizlo, & Delp, 2010) introduces the concept of "camera view" as the basic structural unit of a single-shot UGV. It detects changes in the camera view using a simplified three-parameter global camera motion model in the three major directions (i.e. horizontal, vertical, and radial) which is estimated with the help of the Integral Template Matching algorithm (Lan, Ma, & Zhang, 2003). Finally, trained binary Support Vector Machines (SVMs) are used to classify the camera motion of each video frame, and the final segmentation is created by grouping together neighboring frames that exhibit the same type of camera motion. Following a similar, threshold-less approach, the authors of (Karaman et al., 2014) propose the use of hierarchical hidden Markov models (HHMM) for the indexing of daily living activities in videos acquired from wearable cameras, while an algorithm that combines the concept of "camera views" and the use of hidden Markov models for performing camera motion-based segmentation of UGVs was presented in (González-Díaz et al., 2015). Other motion-based approaches can be found in (Luo et al., 2009) and (Treetasanatavorn, Heuer, Rauschenbach, Illgner, & Kaup, 2004), while a study on different approaches for the estimation of motion in colour image sequences can be found in (Benois-Pineau, Lovell, & Andrews, 2013).

Besides the aforementioned classes of methods, other techniques for the fine-grained segmentation of videos into sub-shots have been proposed. The early approach from (Koprinska & Carrato, 1998) and the more recently proposed algorithm from (Kelm et al., 2009) exploit motion vector information from the compressed video stream at the macro-block level. The methods in (Grana & Cucchiara, 2006) and (Chu et al., 2010) extract various descriptors from the video frames (e.g. color histograms and motion features) and subdivided each shot of the video into sub-shots by clustering its frames into an appropriately determined number of clusters with the help of the c-means and k-means clustering algorithms, respectively. A couple of techniques that utilize data from auxiliary camera sensors (e.g. gps, gyroscope and accelerometers) to identify the camera motion type for every video sub-shot or to identify a group of events in UGVs were discussed in (Wang, Seo, & Zimmermann, 2012) and (Cricri, Dabov, Curcio, Mate, & Gabbouj, 2011) respectively. On a slightly different context, algorithms capable to analyze egocentric or wearable videos were presented in (Lu & Grauman, 2013) and (Xu et al., 2015). Last but not least, several video sub-shot segmentation approaches based on the extraction and processing of 3D spatio-temporal slices (e.g. (Ngo, Pong, & Zhang, 2003) and (Ngo, Ma, & Zhang, 2005)) and using statistical analysis (e.g. (Mohanta, Saha, & Chanda, 2008), (Omidyeganeh, Ghaemmaghami, & Shirmohammadi, 2011) and (Guo, Xu, Sun, Luo, & Sbert, 2016)) were reported, while a comparative study evaluating the performance of different approaches for sub-shot segmentation relying on the extraction and matching of local descriptors, the Pyramidal Lucas-Kanade (PLK) local feature tracker, or block matching motion estimation techniques can be found in (Cooray & O'Connor, 2010).

4.1.3 Developed approaches

4.1.3.1 Video sub-shot fragmentation based on the detection of camera activity Motivated by the motion-based algorithm in (Cooray et al., 2009), our first implementation aims to identify several typical activities that take place during the recording of the video. These include camera pan (i.e. horizontal movement), camera tilt (i.e. vertical movement), any combination of them (i.e. diagonal movement), camera zoom in and camera zoom out. Through the detection and combination of these basic activities more complex video capturing actions can be also identified, such as the ones imitating the use of camera dolly [11] or camera crane [12] equipment. The detection of the aforementioned activities is performed by estimating the spatial displacement between a pair of consecutive or neighboring video frames through the following image matching procedure.

The visual content of each frame is represented by extracting a set of SURF local descriptors (Bay, Ess, Tuytelaars, & Gool, 2008) after setting the Hessian parameter of the algorithm, which affects the number of detected keypoints, equal to 400. Then, for a pair of neighboring frames F_i and F_{i+j} (with j being the frame sampling step) the algorithm applies a descriptor matching process in a brute force manner (i.e. each descriptor extracted from one frame was matched against all the descriptors extracted from the following frame), looking each time for the 2 best matches via k-Nearest Neighbor (k-NN) search. So, for each detected keypoint in frame i it searches for the best matches in frame $i+j$ that correspond to the two nearest neighbors N_1 and N_2. Erroneous matches are then filtered-out based on the following rule: keep a keypoint in frame i and its corresponding best match in frame $i+j$ if $\|N_1\| \; / \; \|N_2\| \leq 0.8$, where $\|\cdot\|$ is the Manhattan distance between the keypoint and the corresponding nearest neighbor. Finally, the algorithm uses the set of remaining matches for calculating a $2x3$ homography matrix between the pair of frames using the RANSAC algorithm (Fischler & Bolles, 1981). The entries of the last column of this $2x3$ matrix correspond to the computed translation (in pixels) at the x- and y-axis. The result of aforementioned image matching process after processing a couple of neighboring frames from the introductory shot of the “Spectre” film is illustrated in Fig. 5.

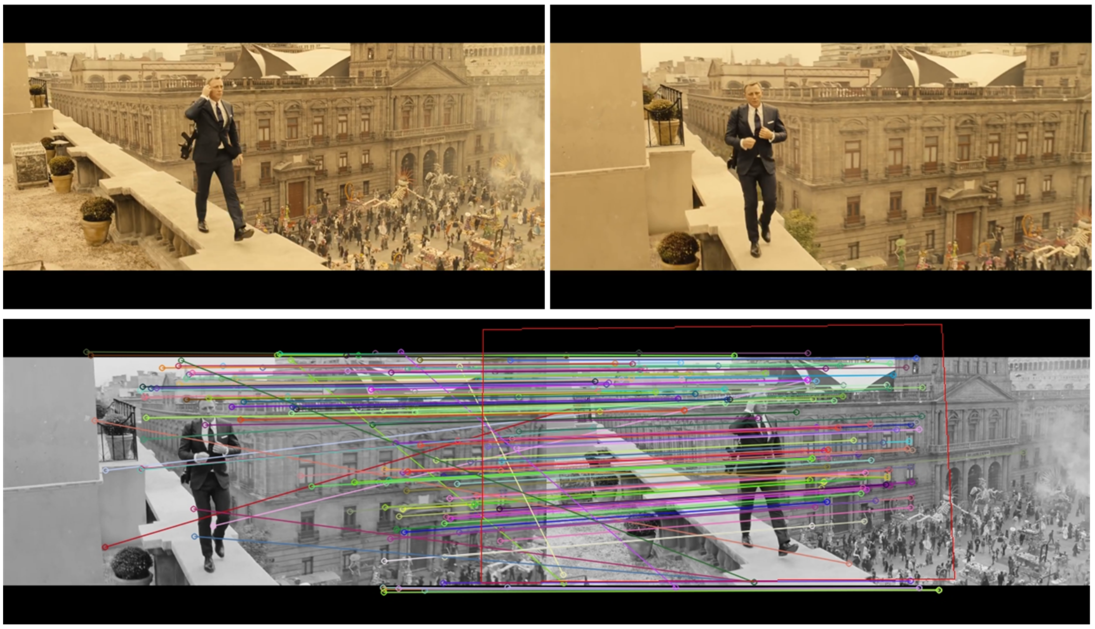


Figure 5: The utilized image matching process on a couple of neigboring video frames from “Spectre”. The top row shows the two neigboring frames. The bottom row shows the matched local descriptors and the estimated spatial displacement between these frames based on the computed homography matrix.

The computed homography matrix for this couple of frames is the following. From the entries in the last column of this matrix is seems that the algorithm estimated a (left) translation of approx. 75 pixels at the x-axis and a (upward) translation of approx. 12 pixels at the y-axis.

$$\begin{bmatrix} +0.838 & +0.001 & -74.871 \\ -0.009 & +0.801 & +12.039 \end{bmatrix} \tag{1}$$

For “enhancing” the differences in the visual content of a pair of frames and, thus, facilitating the detection of any spatial displacement among them, the above mentioned image matching process is not

[11] https://en.wikipedia.org/wiki/Camera_dolly

[12] https://en.wikipedia.org/wiki/Crane_shot

applied on the entire set of frames but in a subset of them, created after sampling the video frames with a constant step equal to the 20% of the video frame-rate (i.e. after selecting 5 equally distant frames per second). After processing each consecutive pair of frames of this subset, the algorithm stores the values of the computed homography matrix that correspond to the translation at the x- and y-axis. The latter results in two vectors of scores when the analysis of the entire subset of frames is completed; one with the x-translation values (denoted as V_x in the sequel) and another one with the y-translation values (denoted as V_y in the sequel). These vectors are post-processed in order to extract information about the type of the video recording activity that caused the estimated spatial displacement over a sequence of video frames. In particular, the algorithm:

- computes the moving average of each vector with the help of a sliding window of size equal to 3, creating vectors V_x' and V_y' respectively;
- identifies the local minima and maxima of the vectors V_x' and V_y' and keeps the local extrema that exceed the upper and lower, experimentally set, thresholds t_1 and t_2;
- for each local extrema of V_x' it finds its previous and following inflection points that correspond to the starting and ending time of a video fragment which contains translation at x-axis;
- for each local extrema of V_y' it finds its previous and following inflection points that correspond to the starting and ending time of a video fragment which contains translation at y-axis;
- the identified sets of video fragments with translation at x- and y-axis from the above analysis are merged and any short-term fragments are filtered-out as outliers, resulting in a new set of video fragments.

This set contains fragments of the four following types:

- fragments containing translation at x-axis only;
- fragments containing translation at y-axis only;
- fragments containing translation at both x- and y-axis;
- fragments with no significant translation at both x- and y-axis.

However, UGVs are, most commonly, captured by amateurs without the use of any professional equipment that ensures the stability of the visual content. So, they might depict a number of short-term minor movements caused by slight hand motion of the camera holder. For filtering-out video fragments depicting this type of minor visual alteration the algorithm computes the total conducted translation at x- and y-axis for each defined fragment in the above collection, by summing the individual translation scores (absolute values) of each pair of frames in the fragment. The computed total translation scores are then expressed as percentage of the video dimensions (i.e. the total translation score at x-axis as percentage of the video's width, and the total translation score at y-axis as percentage of the video's height). The video fragments with total translation scores at both x- and y-axis less than a threshold t_3, (which is also a percentage of the video dimensions), are marked as fragments with minor or no activity and merged with neighboring video fragments of the same type (i.e. with no or minor activity). By adjusting the value of threshold t_3 we are able to modify the sensitivity and response of the developed sub-shot segmentation approach when minor movement of the video recording device takes place. The remaining fragments (i.e. the ones with total translation scores at x- and/or y-axis greater than threshold t_3) are further examined for indicating the type of camera activity that took place during their capturing.

Specifically, through a set of experimentally defined rules and conditions about the curvature and correlation of vectors V_x' and V_y' (see Fig. 6 for indicative examples of this correlation for different types of camera activity), the developed approach distinguishes the following eleven types of camera activity: i) minor camera or object motion, ii) mainly right camera motion, iii) diagonal right and upward camera motion, iv) diagonal right and downward camera motion or camera zoom in, v) mainly left camera motion, vi) diagonal left and downward camera motion, vii) diagonal left and upward camera motion or camera zoom out, viii) mainly upward camera motion, ix) diagonal upward and right camera motion, x) mainly downward camera motion, and xi) diagonal downward and left camera motion.

Finally, for every sub-shot the algorithm extracts three representative keyframes that are uniformly distributed in the video fragment, i.e. are equally distant from each other and from the starting and

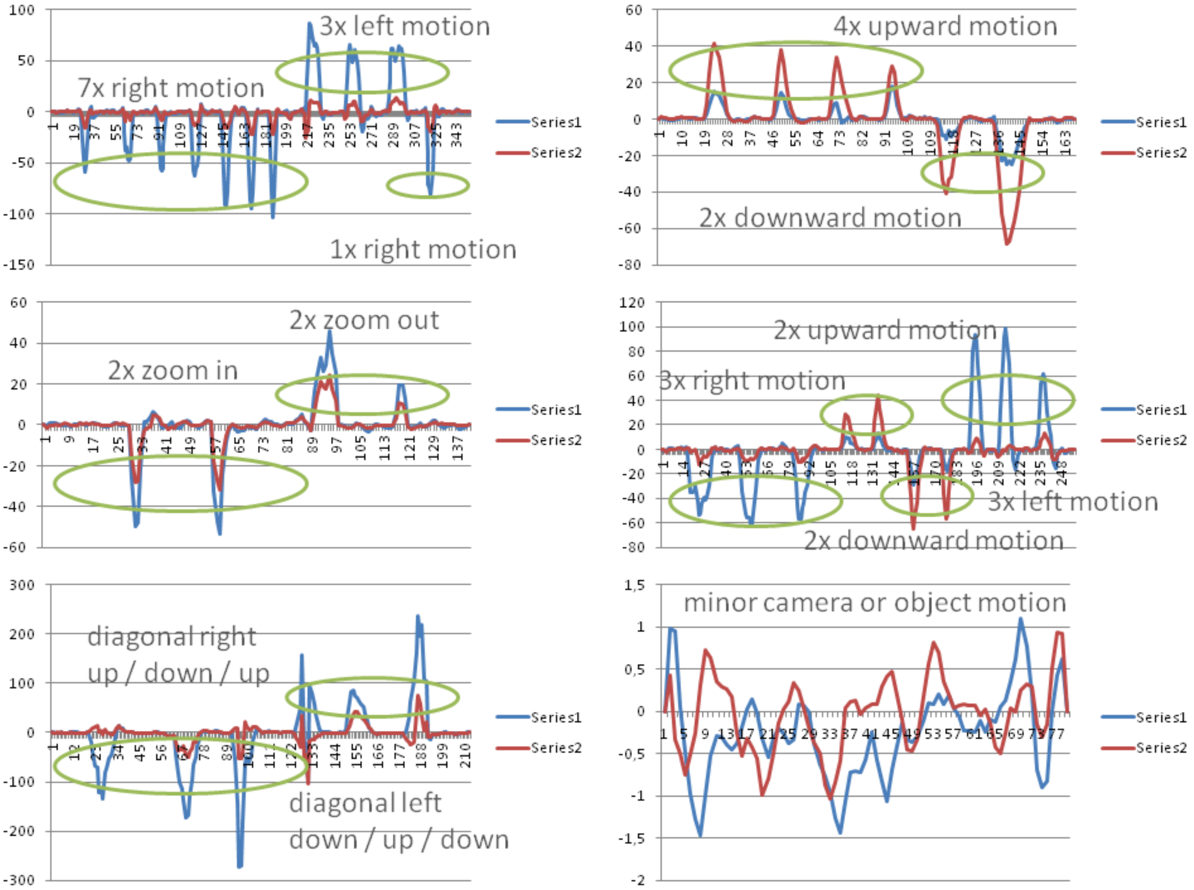


Figure 6: Indicative examples of the correlation between the values of vectors V_x' (x-translation scores, depicted as Series 1 in the diagrams) and V_y' (y-translation scores, depicted as Series 2 in the diagrams) for different types of camera activity. The comparison of the curvature of each vector against a set of experimentally defined rules and the correlation between the values of each vector results in the identification of eleven different types of camera activity for the detected video fragments.

ending frames of the sub-shot. These keyframes are used by the developed concept-based video labeling algorithm (see Section 4.2) for the segment-level semantic annotation of the video, and by the implemented thumbnail extraction method (see Section 4.1.3.3) that is responsible for selecting the most appropriate ones as representative thumbails of the video.

Details about the performance of this sub-shot segmentation algorithm, in terms of detection accuracy and time efficiency are given in Section 4.1.4.

4.1.3.2 Video sub-shot fragmentation based on the visual coherence Based on the findings of the first testing cycle of InVID (in WP7) regarding the performance of the motion-based sub-shot segmentation algorithm presented in the previous section, and driven by the established requirements concerning the time needed for analysis, we developed another approach for temporal fragmentation of UGVs aiming to significantly speed-up the analysis (thus heavily reducing the required processing time) while also targeting to improve its ability to define the appropriate video segments. This method decomposes a single-shot video into sub-shots by assessing the visual resemblance of neighboring frames of the video. For this, the visual content of each processed video frame is described with the help of the well-known Discrete Cosine Transform (DCT), which is an internal processing step when extracting the MPEG-7 Color Layout Descriptor (Kasutani & Yamada, 2001). The pipeline for computing the DCT-based representation of a video frame is illustrated in Fig. 7 and is also the one used by the implemented DCT-based Near Duplicate Detection technique reported in Section 4.2 of D3.1. According to this approach, the video frame is initially resized to $m \times m$ dimensions for increasing the resilience of the analysis against changes in the image aspect ratio and size (step 1 in Fig. 7). Following, the resized image is represented as a sum of cosine functions oscillating at different frequencies via a two-dimensional DCT (step 2). The outcome of this process is an $m \times m$ matrix (for illustration purposes, $m = 8$ in the depicted example) where the top-left element corresponds to the DC coefficient (zero-frequency) and every other element moving from left to right and from top to bottom corresponds to an increase in the horizontal and vertical frequency by a half cycle, respectively. Subsequently, the top-left $r \times r$ part ($r < m$) of the computed matrix ($r = 3$ in the depicted example) is kept, while high-frequency coefficients are discarded, thus removing information related to the visual details of the image (step 3). Finally, a matrix reshaping process is applied to piece together the rows of the extracted $r \times r$ sub-matrix to a single row vector (step 4), and the DC coefficient is then removed (step 5), forming a row vector of size $r^2 - 1$ that represents the image.

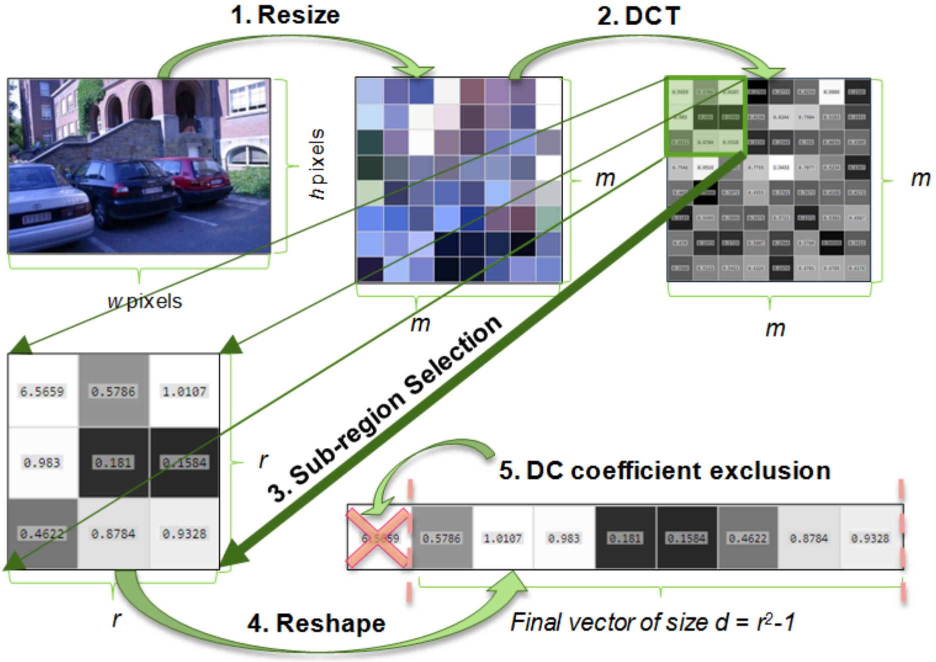


Figure 7: The steps of the applied analysis for extracting the DCT-based representation of the visual content of each processed video frame.

Using the above described DCT-based representation, the visual similarity between a pair of frames is estimated by computing the cosine similarity. Specifically, given a pair of video frames F_i and F_j with descriptor vectors D_i and D_j respectively, their visual resemblance $V_{i,j}$ is calculated by: $V_{i,j} = \frac{D_i \cdot D_j}{\|D_i\|\|D_j\|}$, where $\cdot$ denotes the dot product of the descriptor vectors and $\|\|$ denotes their Euclidean norm. However, subsequent frames of a video, even with the standar frame-rate of 30fps, usually exhibit high visual similarity, which is even bigger for videos of greater frame-rates that users are allowed to capture with modern smartphones or other devices (such as GoPro cameras which support video recoding up to 240fps). Guided by this fact, and similarly to the design and development of the previously described motion-based sub-shot segmentation approach, the aforementioned pair-wise similarity estimation is not applied for every pair of consecutive video frames, but only for neigboring ones selected via a frame-sampling strategy with a fixed-step equal to 33% of the video frame-rate (i.e. it keeps 3 equally distant frames per second). The result of this pair-wise similarity estimation process for the entire set of selected frames is a series of similarity scores which is post-processed for indicating: i) visually coherent video segments with minor activity (exhibiting high visual resemblance) and ii) parts of the video with dynamically changed visual content that denotes major activity (showing lower visual resemblance).

Specifically, the computed series of scores is smoothed with the help of a sliding mean average window of size 3, for reducing the effect of sudden, short-term changes in the visual content of the video (such as the ones introduced after camera flashlights or slight hand movement of the camera holder). Then, the turning points of the smoothed series are defined by computing its 2nd derivative. Each turning point signifies a change in the similarity tendency and therefore a sub-shot boundary. The latter implies that each video sub-shot is delimited by a pair of subsequent turning points in the smoothed series of similarity scores. An example of a smoothed series of similarity scores containing the detected sub-shot boundaries (see yellow vertical lines) is presented in Fig. 8.

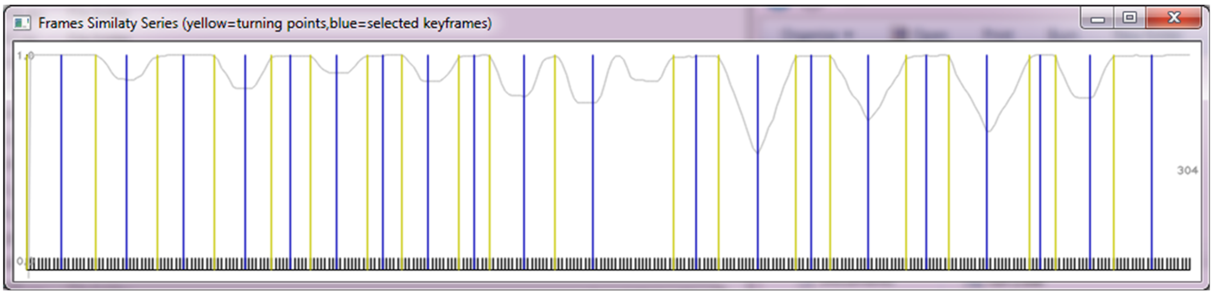


Figure 8: An example of the smoothed series of similarity scores (green curve), the identified sub-shot boundaries (yellow vertical lines) and the selected representative keyframe (blue vertical lines) for each one of them.

This figure also depicts (see blue vertical lines) the selected keyframe for each defined sub-shot. For the sub-shots with minor visual alteration this keyframe corresponds to the middle one, while for the sub-shots that depict some activity (caused either by some of the aforementioned camera activities and/or by the displacement of the shown visual objects) this keyframe corresponds to the most "dynamic" one (i.e. the most dissimilar to the previous neigboring frame from the set of sampled frames that constitute the sub-shot). Finally, two additional keyframes are selected for each sub-shot through a process that defines two sub-parts of the fragment, one being from the starting frame to the selected keyframe and another one being from the selected keyframe to the ending frame of the fragment, and keeps the middle frame of each sub-part. As before, these representative keyframes are further analyzed by the developed InVID technologies for concept-based video labeling and video thumbnail extraction.

Details regarding the precision and time performance of this method are reported in Section 4.1.4.

4.1.3.3 Video thumbnail extraction Drawing input by the algorithms for video temporal fragmentation, this method aims to define a small set of representative keyframes that will be used as video thumbnails in the User Interface of the InVID Dashboard and the Verification Application. In particular, given the collection of extracted keyframes from a video, it describes the visual content of each keyframe using one of the following approaches:

– by extracting low-level features through the above discussed DCT-based methodology;

– by extracting high-level features with the help of Deep Convolutional Neural Networks (DCNN) in a way similar to the algorithm proposed in (Yue-Hei Ng, Yang, & Davis, 2015), using the Caffe DCNN framework (Krizhevsky et al., 2012) and the pre-trained GoogLeNet model (Szegedy et al., 2015).

Following, it clusters the extracted feature vectors using the k-means algorithm, after setting the number of clusters equal to the amount of thumbnails that need to be extracted (this is the only user-defined parameter of the software, which by default equals to 3). Each of these clusters is then represented by the keyframe that is the closest to the center of the cluster, thus being the most visually characteristic one, and this keyframe is selected as one of the video thumbnails. As a final processing step, the implemented algorithm orders the list of extracted thumbnails, enabling the user to select the most appropriate one(s) if a smaller number of thumbnails (e.g. a single one of them) is needed compared to the extracted ones. For this, it computes the dispersion of each defined cluster and it sorts the list of thumbnails based on the descending ordering of the calculated dispersion values.

The time performance of this analysis component is very high, ensuring that no critical overhead load is imposed to the video fragmentation process. Specifically, the DCT-based approach for thumbnail extraction requires 3.5 msec. per keyframe, which means that for a video of 100 sub-shots only $1/3$ of a second is needed for thumbnail extraction. The DCNN-based method for thumbnails extraction demands 31 msec. per keyframe, so for the same video of 100 sub-shots the needed time for thumbnail extraction is approximately 3 seconds. An indicative example of the top-3 extracted thumbnails for the introductory uncut shot of the "Spectre" film by each one of the implemented approaches for the representation of the visual content is presented in Fig. 9 bellow. As shown in this figure, both algorithms selected as the most appropriate one (top-1) a keyframe from the same part of the video (the one that shows the carnival parade in the street of the town). However, the remaining selected thumbnails by the DCT-based method exhibit a visual correlation that is higher than the corresponding one for the DCNN-based technique, which indicates the usefulness of the extracted high-level descriptors in producing a better sub-clustering of the set of extracted keyframes, thus resulting in a better set of video thumbnails.

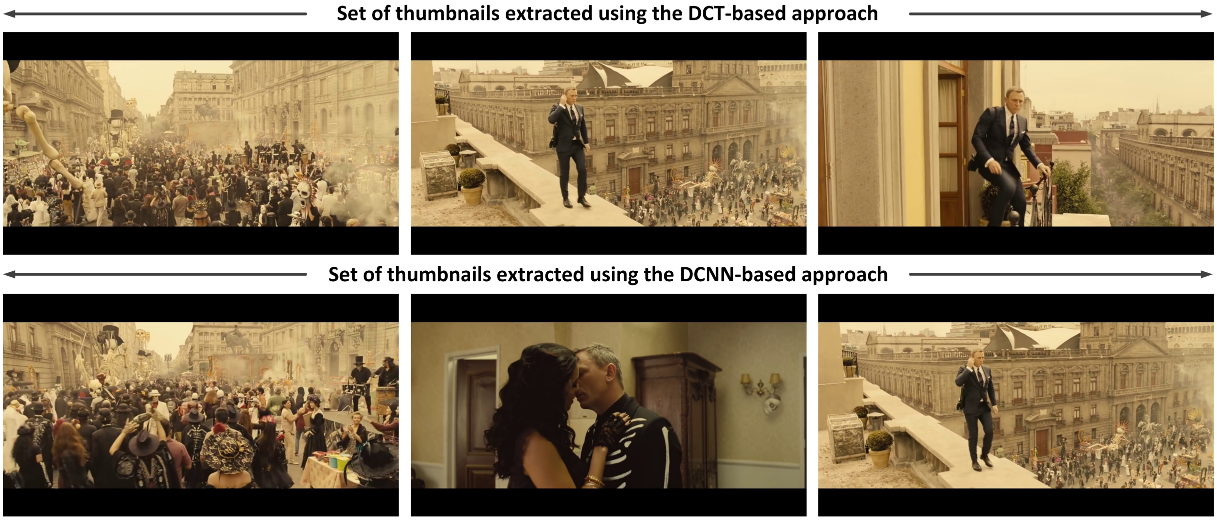

Set of thumbnails extracted using the DCT-based approach
Set of thumbnails extracted using the DCNN-based approach

Figure 9: The top-3 (from left to right) thumbnails extracted for the introductory part of "Spectre" using the two different approaches for representing the visual content of the extracted keyframes. The top row shows the selected thumbnails by the DCT-based method and the bottom row shows the corresponding thumbnails after using the DCNN-based method.

4.1.4 Evaluation

Driven by the lack of online available annotated datasets that can be used for evaluating the performance of the developed sub-shot segmentation approaches [13], we built our own ground-truth dataset. This dataset consists of 35 videos of 82 minutes total duration, which can be divided in the following video genres:

- 15 user-generated videos captured in CERTH's premises using an iPhone (denoted as "Recorded" in the sequel); these videos were created in a way which ensures the presence of all possible different types of camera activity, i.e. multi-directional camera motion and camera zoom in/out.
- 14 uncut shots (also known as "long takes") from known films found on YouTube (denoted as "Films" in the sequel); these videos have been created with the use of professional video capturing equipment under fully controlled recording conditions, so their visual content exhibits stability, continuity and professional quality.
- 5 user-generated videos found on YouTube (denoted as "UGVs" in the sequel); these videos are a typical example of the amateur videos distributed on social networks (e.g. Facebook, Twitter, etc.) and video sharing platforms (e.g. YouTube, Vimeo, etc.), including several types of visual distortion e.g. due to changes in the lighting conditions and short-term displacements of the recording device.
- 1 CCTV demo video found on YouTube (denoted as "CCTV" in the sequel); this is a video demo that presents the functionalities and recording capabilities of a CCTV camera, however it is relatively old, so its visual quality is low in parts of it due to the limited performance of the camera (e.g. highly blurred image when rapid focal change takes place).

The ground-truth for this dataset was created based on human-observation and through an annotation process that indicated: i) parts of the video with no or minor visual activity, ii) parts of the video

[13] Some datasets used in the works reported in Section 4.1.2 do exist, such as the TRECVid 2007 rushes summarization dataset (http://www-nlpir.nist.gov/projects/tv2007/tv2007.html#2.4), the UT Ego dataset (http://vision.cs.utexas.edu/projects/egocentric_data/UT_Egocentric_Dataset.html), the ADL dataset (http://people.csail.mit.edu/hpirsiav/codes/ADLdataset/adl.html) and the GTEA Gaze dataset (http://ai.stanford.edu/~alireza/GTEA_Gaze_Website/), but these datasets were designed for assessing the efficiency of methods targeting specific types of analysis, such as the detection of video rushes and the identification of everyday activities.

with contiguous camera activity of a particular type, (i.e. left camera pan, camera zoom in, right and upward camera movement), iii) parts of the video where both the camera and the captured object are moving (i.e. tracking a moving person with a moving camera), and iv) parts of the video where the visual content is modified due to the activity (e.g. entrance, movement or exit) of the dominant visual object(s). For the video fragments of the second and third category, the direction of the conducted movement was also registered and will be used in our future developments for supporting the automatic identification of several camera activities. The documentation of the dataset and the generated ground-truth data will be made publicly available in the next months via the InVID community in Zenodo [14].

The two developed sub-shot segmentation algorithms were compared against a number of other approaches that rely on the use of well-known visual features. Specifically, we built and evaluated:

– A method similar to (Pan et al., 2007), that assesses the visual similarity of neighboring video frames with the help of HSV histograms and the Chi Square metric (denoted as "HSV" in the sequel).

– A technique that partially overlaps the developed motion-based sub-shot segmentation algorithm and uses the ORB (oriented BRIEF) descriptor (Rublee, Rabaud, Konolige, & Bradski, 2011) (denoted as "ORB" in the sequel). For the pair of video frames F_a and F_b, this method extracts ORB descriptors for the keypoints of the frames and matches them through a Brute-Force, 2-NN searching strategy that computes the Hamming distance of the nearest neighbors. Then, the same distance ratio criterion is utilized for filtering-out erroneous matches, while the remaining ones (inliers) are used for measuring the visual similarity $S_{a,b}$ of the frames through the following formula: $S_{a,b} = max\{\frac{M_a}{D_a}, \frac{M_b}{D_b}\}$, where M_a is the number of matched descriptors of the frame F_a, D_a is the total number of extracted descriptors from the frame F_a, M_b is the number of matched descriptors of the frame F_b and D_b is the total number of extracted descriptors from the frame F_b.

– A variation of the algorithm in (Cooray et al., 2011), which computes the optical flow for a sparse feature set (denoted as "SOF" in the sequel). For this, we used the "goodFeaturesToTrack" algorithm of OpenCV to detect corners in the images and the iterative Lucas-Kanade method to estimate the optical flow. Then, we averaged the resulting motion vector to end up with a single vector that describes the camera movement. Finally, the similarity between neighboring video frames was estimated using the inverse of the magnitude of this vector.

– An alternative approach that estimates the visual resemblance of a pair of video frames with the help of DCNN-based features (denoted as "DCNN" in the sequel). For this, we used the Caffe DCNN framework (Krizhevsky et al., 2012) to extract the convolutional responses of the "inception_3a/output" layer of the pre-trained GoogLeNet model (Szegedy et al., 2015), and we applied a method similar to (Yue-Hei Ng et al., 2015) using max pooling (instead of VLAD aggregation), to speed-up the feature extraction process. Finally, the visual similarity is evaluated by computing the L2 distance between the formed DCNN vectors.

The findings regarding the performance of the studied sub-shot segmentation methodologies (in terms of Precision, Recall and F-Score values) are reported in the following Table 9. As a general comment, these data make clear that the fine-grained fragmentation of a single-shot video into sequences of frames showing either a visually static scene or a continuous and contiguous change of the visual content due to typical video recording activities, is a highly challenging task. Looking at the most global metric F-Score, the best performance is achieved by the developed algorithm that assesses the visual coherence of neighboring video frames with the help of DCT descriptors (described in Section 4.1.3.2), while the second best corresponds to the developed motion-based approach that relies on the extraction and matching of SURF descriptors (presented in Section 4.1.3.1). Then, simpler methods that assess the visual similarity of video frames with the help of HSV histograms or ORB local descriptors exhibit less competitive performance, similar to the one shown by the optical flow algorithm. Finally, the tested DCNN-based technique was proven to be the least effective one.

Analyzing a bit more the detection effectiveness of each of the evaluated techniques with the help of the computed Precision and Recall scores, we can see that the implemented DCT-based approach has, by far, the highest Recall score, which means that a large number of video fragments is defined by the analysis. However, the Precision score of this method indicates that only a small part of these fragments correspond to actual sub-shots of the ground-truth dataset, thus resulting in a medium level

[14] https://zenodo.org/communities/invid-h2020

of overall performance (expressed by the F-Score value). The most efficient algorithms in terms of Precision, seem to be the ones that rely on the extraction and matching of low- or mid-level features between neighboring frames (e.g. SURF, ORB and DCNN features), with the best one being the developed motion-based method that also identifies the camera activity through the estimation of the homography matrix. Competitive performance with the best one in terms of Recall scores is recorded for the implemented HSV-based and the SOF-based algorithms, however the Precision of both of these methods is lower than that of the DCT-based approach.

Table 9: The experimental results regarding the performance (in terms of Precision, Recall and F-Score) of the six tested sub-shot segmentation approaches for four different types of videos.

	Motion-based			Visual coherence			HSV		
	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
Recorded	0,833	0,851	0,842	0,709	0,957	0,814	0,551	0,753	0,636
Films	0,319	0,409	0,358	0,283	0,794	0,417	0,182	0,720	0,291
UGVs	0,330	0,521	0,404	0,193	0,782	0,309	0,162	0,790	0,269
CCTV	0,522	0,414	0,462	0,622	0,966	0,757	0,206	0,897	0,335
Total	**0,381**	0,487	0,428	0,296	**0,820**	**0,435**	0,196	0,744	0,311
	ORB			SOF			DCNN		
	Precision	Recall	F-Score	Precision	Recall	F-Score	Precision	Recall	F-Score
Recorded	0,496	0,681	0,574	0,628	0,755	0,686	1,000	0,043	0,082
Films	0,282	0,445	0,345	0,176	0,845	0,292	0,352	0,189	0,246
UGVs	0,253	0,311	0,279	0,235	0,780	0,362	0,469	0,252	0,328
CCTV	0,253	0,724	0,375	0,315	0,586	0,410	0,429	0,207	0,279
Total	0,301	0,464	0,365	0,204	0,815	0,326	0,378	0,181	0,246

Finally, by evaluating the performance of the tested technologies for the different types of video content we can see that the highest efficiency for almost all methods (besides the DCNN-based one) is presented for the generated videos in CERTH's premisses, while this performance is reduced for the other categories. This is reasonable due to the fact that these videos were captured in a way that minimizes the existence of camera shakes, blurs or changes in illumination that are common in amateur UGVs and induce the detection of false alarms by the sub-shot segmentation algorithms. Moreover, between the top-2 methods (i.e. the DCT-based one and the motion-based one) the algorithm that relies on the assessment of the visual coherence exhibits better performance in cases where a moving person is tracked by a moving camera (such as in films) and in cases where a static scene changes due to the activity of a visual object (such as in CCTV videos), while the motion-based one is slightly better in the case of amateur UGVs.

Concerning time-efficiency, the needed time for analysis by each tested method is reported in the following Table 10. The required processing time is expressed as a percentage of the video duration, with a value equal to 100% meaning that the analysis can be performed in real-time. The reported scores indicate that the developed DCT-based sub-shot segmentation algorithm is the fastest one, being more than 25 times faster than real-time analysis, while rather similar performance is observed for the HSV- and the ORB-based approaches. However, both of these methods exhibited significantly lower performance in terms of detection accuracy. The SOF- and the DCNN-based algorithms are 2 to 3 times slower, while the motion-based one that involves the extraction and matching of complex local descriptors (compared to the binary ORB descriptors) and the estimation of the homography between pairs of frames is by far the slowest one (i.e. being a bit faster than real-time analysis).

Table 10: The experimental results about the time-efficiency of the six tested sub-shot segmentation approaches in terms of the needed time for analysis (expressed as a percentage of the video duration).

Approach	Processing time (% of video duration)
Motion-based	92,556
Visual coherence	**3,677**
HSV	4,166
ORB	8,781
SOF	10,714
DCNN	13,367

Based on the findings concerning the detection accuracy and the time-efficiency of each tested approach we concluded that the best choice for the fragmentation of single-shot videos in the InVID platform is the developed DCT-based algorithm. This method is the best trade-off between accurate and fast analysis. Moreover, it was more effective when it was compared against the motion-based one with the help of the developed web application for video fragmentation and reverse keyframe search (see Section 4.1.6 bellow). Through the assessment of the visual coherence of different parts of the video with the help of the DCT-based approach, the users of the application were provided a rich set of representative keyframes in a very shot time after the submission of the video for analysis, and this set of keyframes supported the successful search and retrieval of near duplicates of the video from the web. Nevertheless, as shown by the computed values about the detection accuracy of this method, this is room for further improvement, mainly in terms of Precision, and this will be our main focus for the future developments of the project at this direction.

4.1.5 API

The InVID video fragmentation and thumbnail extraction module has been implemented as a REST service using the Python Bottle (version 0.11.5) framework. The service, that is hosted by CERTH, integrates the developed algorithms for video sub-shot segmentation and thumbnail extraction. Moreover, it includes existing CERTH technologies for temporal fragmentation of professional, multi-shot videos into shots (Apostolidis & Mezaris, 2014) and scenes (Sidiropoulos et al., 2011) (scenes being defined as groups of visually coherent and temporally aligned shots that correspond to the story-telling parts of the video).

The base URL of the service is: `http://multimedia2.iti.gr:8080` and the access to the service is permitted only to authorized users with a valid user key. For each registered user there is a limit regarding the total duration of video content sent for analysis during the last 10 hours of operation. As input, the service takes the URL of the video file that needs to be analyzed; this URL can link to a video file hosted in online repositories (both FTP and HTTP), or found in video/file sharing platforms and social networks [15]. As output, the service creates: i) a JSON file with the video fragmentation analysis results, ii) two collections of image files that correspond to the extracted keyframes for the detected shots and sub-shots of the video, and iii) a collection of image files that correspond to the extracted thumbnails for the video. For initiating the video fragmentation analysis for a given video, the user must commit an HTTP POST request on `http://multimedia2.iti.gr:8080/segmentation`. The body of the HTTP POST request is in JSON format and contains the following parameters:

- "video_url": the URL of the video to be processed
- "login", "password": optional parameters used for authentication checks in case of password-protected repositories
- "user_key": a unique 32-digits access key that allows access to the service
- "kf_num_sh": an optional argument that defines the number of extracted keyframes per video shot (default value is 3)
- "kf_num_sb": an optional argument that defines the number of extracted keyframes per video sub-shot (default value is 3)
- "thumb_num": an optional argument that defines the number of extracted thumbnails for the video (default value is 3)

The communication between the REST service and the user is synchronous only during the transmission of the call. Before turning into asynchronous, and thus enabling the submission of a new HTTP POST request from the same user, the service provides details (in JSON format) about the successful receipt of the analysis request and the assigned identifier to the video file, or informs the user concerning a number of different violated conditions (e.g. wrongly formatted analysis request, total video duration exceeded, or broken video URL) that prevented the initialization of the analysis.

[15] The currently supported online sources include YouTube, DailyMotion, Facebook, Twitter and Dropbox; nevertheless, failure of the video fetching process can be experienced based on the operation of the used third party video downloading software (i.e. the "you-get" downloader from `https://pypi.python.org/pypi/you-get`), and platform- or user-defined restrictions about the use of the video.

After the start of the video processing, the user is able to get information about the status of the analysis by committing an HTTP GET request on `http://multimedia2.iti.gr:8080/status/<video_id>`, where "video_id" is the automatically assigned identifier to the video file. Then, the service responds with a text message describing in details the conducted progress concerning the analysis of the video. Moreover, through a set of specific HTTP GET requests the user is able to get:

- the JSON file with the analysis results that include temporal information about the detected scenes, shots and sub-shots of the video (i.e. their time boundaries in seconds), the defined keyframes for each video shot and sub-shot (i.e. their timestamps in seconds), and the extracted thumbnails (i.e. their ordering and their timestamps in seconds);
- the set of extracted keyframes for the shots of the video (as a zipped file);
- the set of extracted keyframes for the sub-shots of the video (as a zipped file);
- the set of extracted thumbnails for the video (as a zipped file);
- the extracted keyframes for the shots of the video on a one-by-one basis;
- the extracted keyframes for the sub-shots of the video on a one-by-one basis.

Last but not least, the user of the service should be informed that the video files submitted to the web service for analysis and the corresponding analysis results are automatically deleted from the server approximately 48 to 72 hours after their analysis is completed.

4.1.6 Web application for reverse keyframe search

As presented in D6.2, the InVID Verification Application integrates a component for automatic near duplicate video detection (see Section 3.3.1 of D6.2) which utilizes the corresponding technologies reported in Section 4 of D3.1. However, wanting to offer to the users/journalists an alternative solution for this specific video verification process (which can also serve as a teaser for attracting users to the complete InVID Verification Application), we built an interactive web application that supports fast near duplicate video detection via keyframe extraction and reverse keyframe search. This web application is based on utilizing the results of the video fragmentation and keyframe/thumbnail extraction component described above.

The existence of other online tools that support the search and retrieval of near duplicates of an image or video is a strong indication that this type of analysis is very useful when trying to determine the originality of an online shared video. For example, the Youtube DataViewer of the Amnesty International [16] enables the users to find near duplicates of a YouTube video, while the Custom Reverse Image Search of the IntelTechniques [17] extends this search on other platforms, including Vimeo, Facebook, Vine, Instagram, LiveLeak and Backpage. However, both of these solutions perform reverse video search based on a limited set of randomly selected keyframes/thumbnails that has been associated to the video, thus excluding parts of the video that could enhance the reverse search or be of particular interest to the user. Moreover, the search is supported only for videos available online, thus making impossible the reverse video search for a video stored in the user's machine. In contrast, our web application extracts a dynamic number of keyframes in a way which ensures that all the visually discrete parts of the video are adequately represented through the extracted set of keyframes. Furthermore, as described in the following, it supports the direct analysis of both online available videos from several platforms and local copies of a video from the user's machine without requiring its prior upload to any video sharing platform. In this way, it assists users to quickly discover the temporal structure of a video, to extract detailed information about the video content and to use this data in their reverse video search queries. A slightly different approach is applied by Spotter [18], which is an online reverse video search engine. According to its creators, Spotter uses advanced Computer Vision and Machine Learning techniques to track any video on the web by using a video as query and searching on a constantly increasing database of pre-analyzed and indexed videos. This means that the efficiency of the search and thus, the quality of the analysis results heavily depend on the size of the indexed video content in the Spotter platform. A preliminary evaluation using a small number of videos (due to restrictions raised by the beta version of the tool) showed that the analysis depends on a manually extendable set of automatically

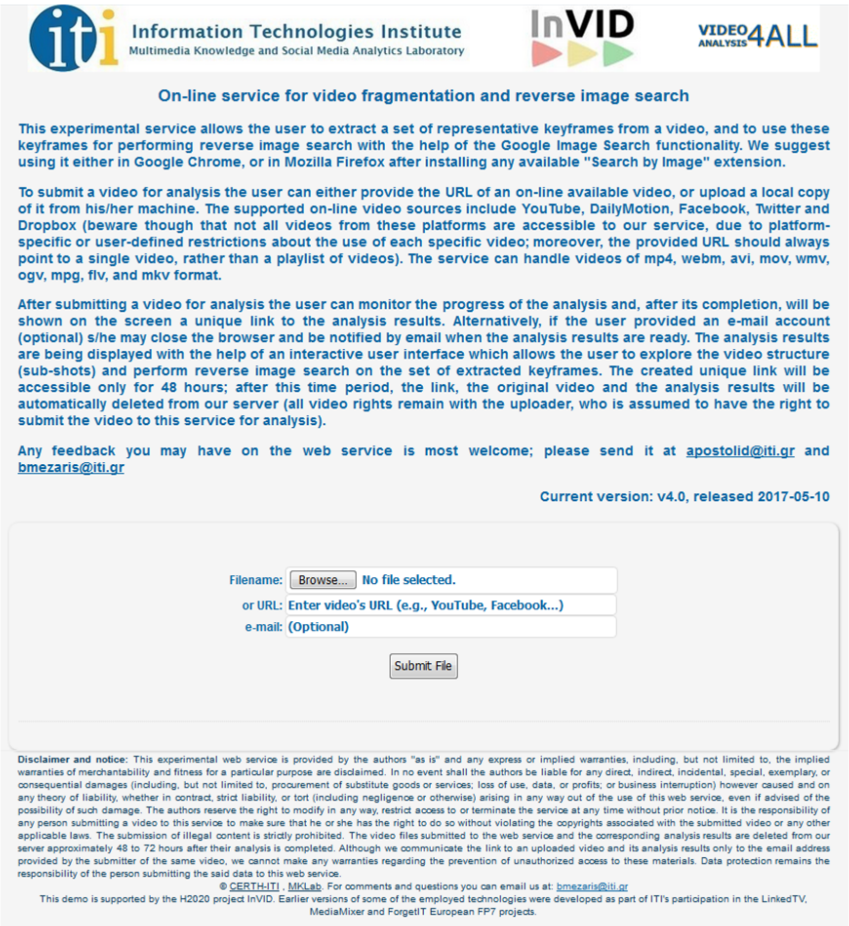
[16] `https://citizenevidence.amnestyusa.org/`

[17] `https://inteltechniques.com/osint/reverse.video.html`

[18] `https://spotter.tech/`

extracted keywords from the name of the video. In most cases it led to the retrieval of a reasonable number of occurrences of the video in various platforms (e.g. Facebook and Youtube) but it failed to find the most needed or the most associated media item with the video (be it the most shared tweet on the topic and the one that started all the buzz around the video, or the original video that was faked and reused), while in other cases the tool was not able to find any near duplicates of a video with many copies on YouTube. Motivated by the index-based methodology of Spotter, in InVID we will go for an integrated solution for reverse video search. In particular, the web application presented in this section and the near duplicate detection component reported in D3.1, will have a complementary role, in the sense that the created index of the analyzed videos via the near duplicate detection component of the InVID Verification Application will be an additional source of information when looking for near duplicates with the help of the developed web application for video fragmentation and reverse keyframe search. In this way, a more complete search that involves the internal (i.e. within the InVID platform) and external (i.e. within the Web) investigation for near duplicates of a given video will be enabled for the users of the application.

The developed web application is hosted by CERTH and can be accessed at `http://multimedia2.iti.gr/videofragmentation_v4/service/start.html`. As depicted in Fig. 10, its user interface contains: i) instructions of use and limitations of the service, ii) details about the generated analysis results and the video rights, and iii) an entry point for submitting a video for analysis. The latter can be done either via providing the URL of an online available video or by uploading a local copy of it from the user's machine. The supported online video sources include YouTube, DailyMotion, Facebook, Twitter and Dropbox. However, not all videos from these platforms are accessible to our service due to platform-specific or user-defined restrictions about the use of each specific video. Moreover, the provided URL should always point to a single video, rather than a playlist of videos. The supported video formats include "mp4, "webm", "avi", "mov", "wmv", "ogv", "mpg", "flv" and "mkv".


Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
InVID
VIDEO ANALYSIS 4ALL

On-line service for video fragmentation and reverse image search

This experimental service allows the user to extract a set of representative keyframes from a video, and to use these keyframes for performing reverse image search with the help of the Google Image Search functionality. We suggest using it either in Google Chrome, or in Mozilla Firefox after installing any available "Search by Image" extension.

To submit a video for analysis the user can either provide the URL of an on-line available video, or upload a local copy of it from his/her machine. The supported on-line video sources include YouTube, DailyMotion, Facebook, Twitter and Dropbox (beware though that not all videos from these platforms are accessible to our service, due to platform-specific or user-defined restrictions about the use of each specific video; moreover, the provided URL should always point to a single video, rather than a playlist of videos). The service can handle videos of mp4, webm, avi, mov, wmv, ogv, mpg, flv, and mkv format.

After submitting a video for analysis the user can monitor the progress of the analysis and, after its completion, will be shown on the screen a unique link to the analysis results. Alternatively, if the user provided an e-mail account (optional) s/he may close the browser and be notified by email when the analysis results are ready. The analysis results are being displayed with the help of an interactive user interface which allows the user to explore the video structure (sub-shots) and perform reverse image search on the set of extracted keyframes. The created unique link will be accessible only for 48 hours; after this time period, the link, the original video and the analysis results will be automatically deleted from our server (all video rights remain with the uploader, who is assumed to have the right to submit the video to this service for analysis).

Any feedback you may have on the web service is most welcome; please send it at apostolid@iti.gr and bmezaris@iti.gr

Current version: v4.0, released 2017-05-10

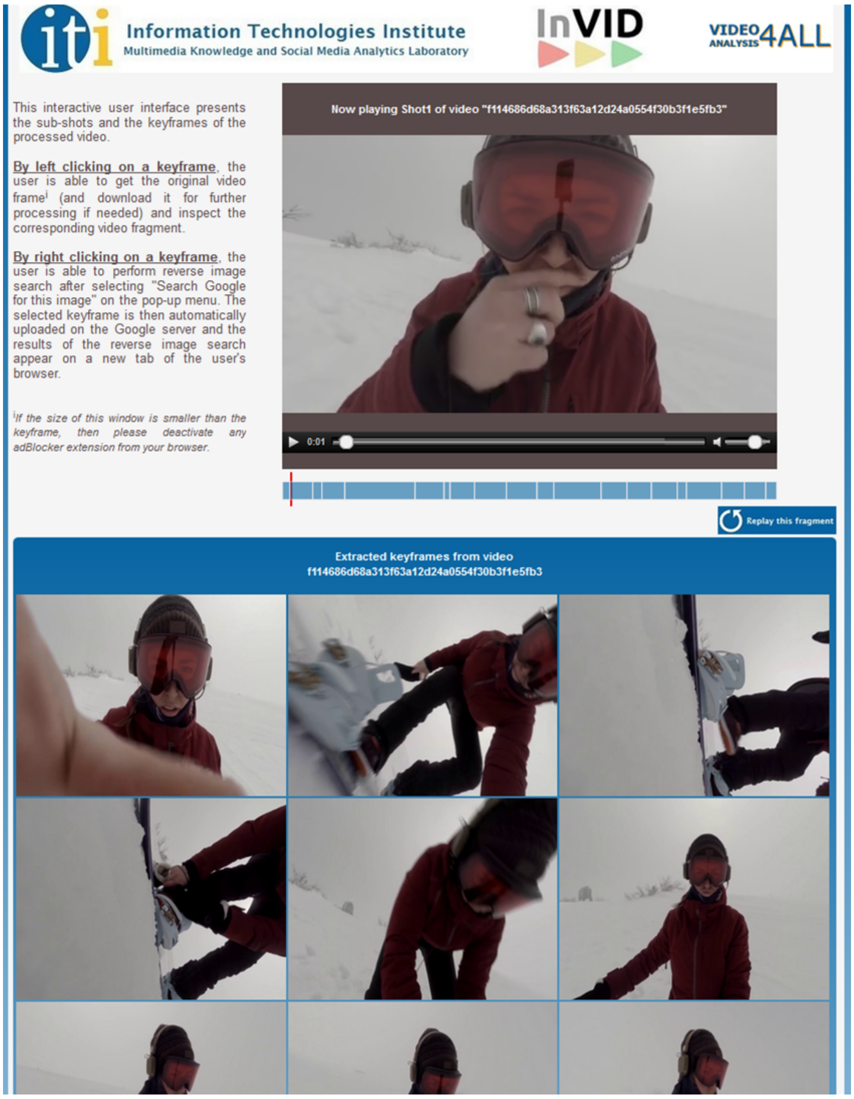
Filename: Browse... No file selected.
or URL: Enter video's URL (e.g., YouTube, Facebook...)
e-mail: (Optional)

Disclaimer and notice: This experimental web service is provided by the authors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the authors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this web service, even if advised of the possibility of such damage. The authors reserve the right to modify in any way, restrict access to or terminate the service at any time without prior notice. It is the responsibility of any person submitting a video to this service to make sure that he or she has the right to do so without violating the copyrights associated with the submitted video or any other applicable laws. The submission of illegal content is strictly prohibited. The video files submitted to the web service and the corresponding analysis results are deleted from our server approximately 48 to 72 hours after their analysis is completed. Although we communicate the link to an uploaded video and its analysis results only to the email address provided by the submitter of the same video, we cannot make any warranties regarding the prevention of unauthorized access to these materials. Data protection remains the responsibility of the person submitting the said data to this web service.

© CERTH-ITI , MKLab. For comments and questions you can email us at: bmezaris@iti.gr
This demo is supported by the H2020 project InVID. Earlier versions of some of the employed technologies were developed as part of ITI's participation in the LinkedTV, MediaMixer and ForgetIT European FP7 projects.

Figure 10: The start page of the developed web application for video keyframe extraction and reverse keyframe search.

After submitting a video for analysis the user is able to monitor the progress of the analysis and, after its completion, to get on the screen a unique link to the analysis results. Alternatively, if the user provided an e-mail account (optional) s/he may close the browser and be notified by e-mail when the analysis results are ready. The analysis is based on the video sub-shot segmentation algorithm described in Section 4.1.3.2 and the analysis results are being displayed with the help of the interactive user interface illustrated in Fig. 11. This interface allows the user to explore the video structure (sub-shots) and perform reverse image search on the set of extracted keyframes. The latter can be done simply by left clicking on any desired keyframe and selecting “Search Google for this Image”, an action that initiates a Google Image Search; the results of this search are served to the user in a new tab of his/her browser (see the example in Fig. 12). The created unique link is accessible only for 48 hours, while after this time period, the link, the original video and the analysis results are automatically deleted from our server. Last but not least, all video rights remain with the uploader, who is assumed to have the right to submit the video to this service for analysis.


Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
InVID
VIDEO ANALYSIS 4ALL

This interactive user interface presents the sub-shots and the keyframes of the processed video.

By left clicking on a keyframe, the user is able to get the original video frame[i] (and download it for further processing if needed) and inspect the corresponding video fragment.

By right clicking on a keyframe, the user is able to perform reverse image search after selecting "Search Google for this image" on the pop-up menu. The selected keyframe is then automatically uploaded on the Google server and the results of the reverse image search appear on a new tab of the user's browser.

[i] If the size of this window is smaller than the keyframe, then please deactivate any adBlocker extension from your browser.

Now playing Shot1 of video "f114686d68a313f63a12d24a0554f30b3f1e5fb3"

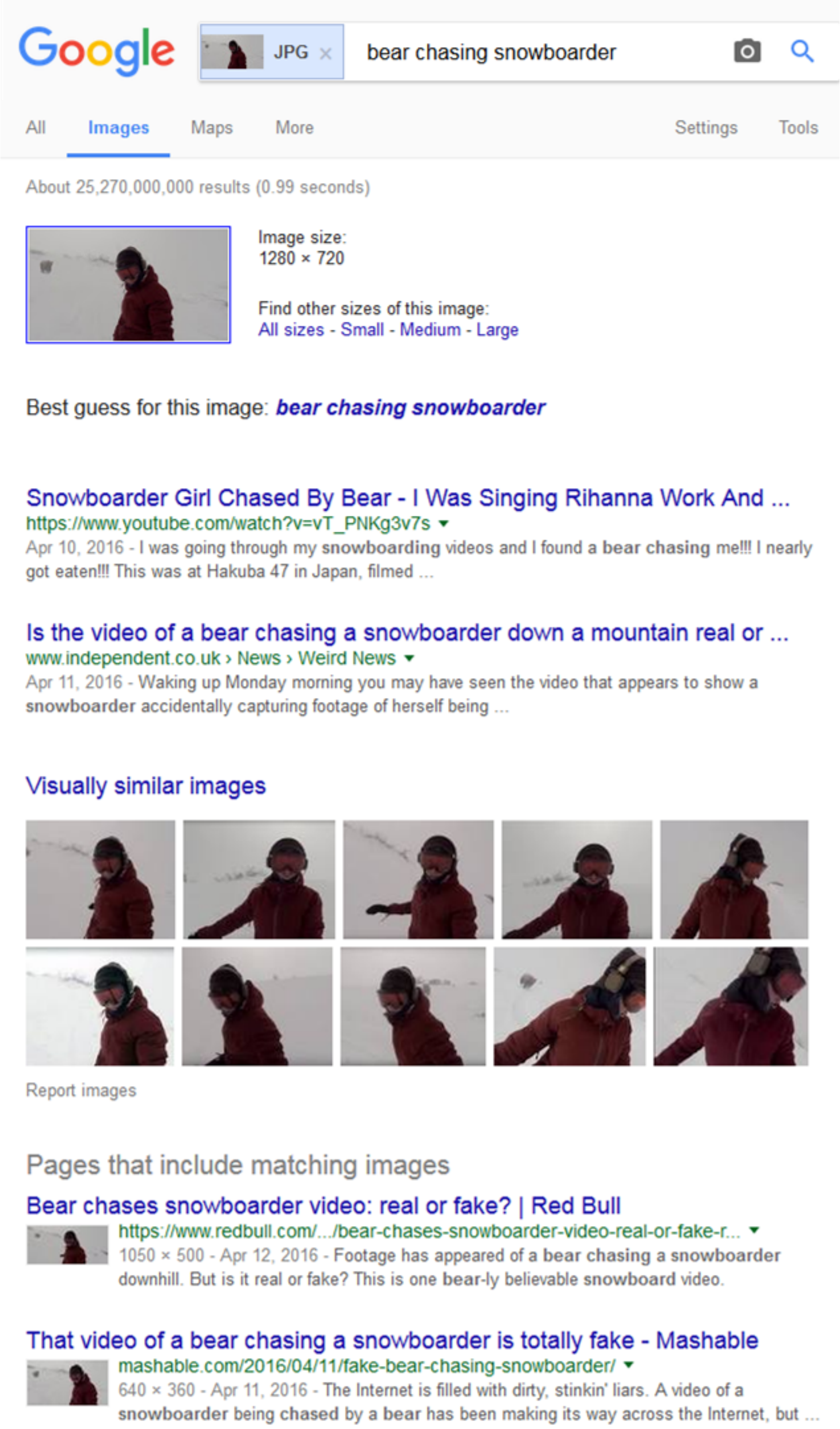
Extracted keyframes from video f114686d68a313f63a12d24a0554f30b3f1e5fb3

Figure 11: The interactive user interface with the extracted keyframes and the reverse keyframe search functionality.

This technology has been thoroughly evaluated through the first three testing cycles of the project (see Section 4.3 of D7.1). Based on the partners’ feedback, several improvements were performed, including:

– the development of a faster method for the fragmentation of the video (as mentioned in the beginning of Section 4.1.3.2);

– the analysis of online available videos (only locally available videos in the users’ machine could be



About 25,270,000,000 results (0.99 seconds)

Image size:
1280 × 720

Find other sizes of this image:
All sizes - Small - Medium - Large

Best guess for this image: *bear chasing snowboarder*

Snowboarder Girl Chased By Bear - I Was Singing Rihanna Work And ...
https://www.youtube.com/watch?v=vT_PNKg3v7s ▾
Apr 10, 2016 - I was going through my **snowboarding** videos and I found a **bear chasing** me!!! I nearly got eaten!!! This was at Hakuba 47 in Japan, filmed ...

Is the video of a bear chasing a snowboarder down a mountain real or ...
www.independent.co.uk › News › Weird News ▾
Apr 11, 2016 - Waking up Monday morning you may have seen the video that appears to show a **snowboarder** accidentally capturing footage of herself being ...

Visually similar images

Report images

Pages that include matching images

Bear chases snowboarder video: real or fake? | Red Bull
https://www.redbull.com/.../bear-chases-snowboarder-video-real-or-fake-r... ▾
1050 × 500 - Apr 12, 2016 - Footage has appeared of a **bear chasing** a **snowboarder** downhill. But is it real or fake? This is one **bear**-ly believable **snowboard** video.

That video of a bear chasing a snowboarder is totally fake - Mashable
mashable.com/2016/04/11/fake-bear-chasing-snowboarder/ ▾
640 × 360 - Apr 11, 2016 - The Internet is filled with dirty, stinkin' liars. A video of a **snowboarder** being **chased** by a **bear** has been making its way across the Internet, but ...

Figure 12: The analysis results of a Google Image Search for an extracted keyframe after analyzing a YouTube video with the developed web application. More than 180 links to the video were found, including 12 different copies of the original video.

submitted for analysis in the initial version of the application);

- the direct access to the analysis results (by making optional the provision and use of an e-mail account as means for receiving the unique link to the analysis results);
- the simplification of the reverse keyframe search process;
- updating its documentation by including more condensed and descriptive details about the use of the service and a statement about the video rights.

Nevertheless, further improvements are required, which are mostly related to the visual content of the extracted keyframes and, thus, their appropriateness for reverse keyframe search. In particular, the conducted tests indicated the need to: i) filter out blurred or blank frames with limited or no usability in terms of reverse keyframe search, ii) select more descriptive and representative keyframes from a video sub-shot, and iii) extract a larger number of keyframes from video sub-shots characterized by intense local or global motion. Moreover, a set of suggestions concerning the design, the functionality and the usability of the user interface was also provided by the testers of the web application. These suggestions will be taken under consideration for preparing the next version of the web application, which is currently under development and will be released within the first week of July.

The aforementioned requirements will guide or future developments on the tool, that will include: i) the integration of an algorithm for detecting and discarding blurred or blank keyframes, ii) the implementation of a method for assessing the “representativeness” of each keyframe and selecting the most desirable ones, iii) the extension of the utilized sub-shot segmentation approach in order to extract and exploit information about the dynamics (e.g. via estimating the intensity of motion activity) of each fragment, and iv) the improvement of the user interface according to the experts’ needs.

4.2 Concept-based Video Labeling

4.2.1 Motivation and foreseen use

With the help of the InVID Dashboard the user (i.e. a journalist) is able to gather a set of videos related to a certain topic of interest. This process includes the collection of various types of online available contextual information and metadata about these videos, such as subtitles, date of creation/publish, related posts and so on. However, there is a rich source of information about their content which is still unexploitable; their visual content itself. Driven by the need to extract human-interpretable metadata concerning the video content, we built a method that performs concept-based video labeling via detecting a set of high-level visual concepts (such as “car”, “building”, “demonstration”, “running”) in the video fragments defined by the video fragmentation component of the InVID platform. For this, it applies a fragment-level semantic concept detection, assigning one or more semantic concepts to each fragment (which is represented via a set of keyframes) based on a predefined list of visual concepts. In a typical process, the extracted keyframes for each defined video fragment (using for instance one of the video fragmentation approaches presented in Section 4.1) are passed through a pre-trained deep convolutional neural network (DCNN) that performs the final class label prediction directly, using typically a softmax or a hinge loss layer (Simonyan & Zisserman, 2014; Krizhevsky et al., 2012).

Nevertheless, the small number of labeled training examples is a common problem in video datasets, making it difficult to train a deep network from scratch without over-fitting its parameters on the training set (Snoek et al., 2015). For this reason, it is common to use transfer learning, i.e. to take a network that has been trained on a large-scale source dataset (e.g ImageNet (Russakovsky, Deng, & et al., 2015)) and fine-tune its parameters for the target dataset. Furthermore, the tasks in the target dataset may be related, and so their relations can be exploited to further improve the video concept detection accuracy. Concept correlations obtained by the ground-truth annotation can provide a source of information regarding the relations between tasks. Additionally, concepts, besides label relations, can be related in terms of their feature representation or the task parameters, i.e. the parameters of the binary classifier learned from the training data. Multi-task learning (MTL) refers to those methods that learn many tasks together at the same time.

Based on the above, we initially investigate the performance of three different approaches for transfer learning. Based on the findings of this evaluation, we build the InVID method for concept-based video labeling by appending an MTL-like loss to a neural network and minimizing the entire network end-to-end. In addition, we incorporate a label-based constraint related to the concept correlations. We refer to the proposed method as deep multi-task learning with label constraint (DMTL_LC) and we apply it on a transfer learning scenario. Specifically, we extend the two-sided neural network, proposed in (Yang & Hospedales, 2015) for MTL, in the following ways: i) we use the network jointly with a pre-trained network in order to perform transfer learning, instead of using it as a standalone network that takes as input hand-crafted or DCNN-based features; ii) we introduce a new label-based constraint that considers concept correlations.

Through the developed technique for concept-based video labeling, the user of the InVID Dashboard will be able to extract a rich set of metadata describing its semantic content and use these metadata for: i) indexing the collected group of videos, ii) searching this group based on high-level concepts, and iii) finding associations between relevant video fragments. The aforementioned activities assist the user to better manage, discover, understanding and exploit the collected multimodal information with the help of the InVID Dashboard around a particular topic of interest.

4.2.2 State of the art

MTL and transfer learning are two strategies to improve learning by sharing knowledge across different but related tasks or domains. Let us define two domains with their learning tasks: the source domain D_{so} with a set of learning tasks T_{so} (the set of concepts that need to be detected), and the target domain D_{ta} with a set of learning tasks T_{ta}. On the one hand, transfer learning aims to improve the learning in D_{ta} by using the knowledge in D_{so}, without considering potential improvements to the tasks of D_{so}. The latter is the focus of multi-domain learning (MDL). On the other hand, MTL methods learn the relations across the learning tasks T_{so} or T_{ta} together at the same time. It should be noted that the terms MTL and MDL are sometimes used interchangeably. However, it is useful to distinguish them clearly: MDL refers to shared knowledge about the same tasks across different domains, while MTL refers to shared knowledge about different tasks in the same domain. The latter is the focus of this work, so we do not further discuss methods that focus on MDL, such as (Long & Wang, 2015).

Noisy and incomplete annotations are common in video datasets (e.g TRECVID SIN (Over et al., 2013)), which makes it difficult to train a deep neural network from scratch (Snoek et al., 2015). Many works investigate which features within a pre-trained network are sufficiently generic, and develop approaches that effectively transfer this knowledge to new target datasets. The typical approach for transfer learning is to start with a DCNN trained in D_{so}, replace its classification layer with a new T_{ta}-dimensional classification layer and train it towards the D_{ta} domain (Chatfield et al., 2014; Yosinski et al., 2014; Girshick et al., 2014; Chaabouni, Benois-Pineau, & Amar, 2016). The way that the parameters of the source DCNN will be used has been examined in many works. For example, in (Oquab et al., 2014; Yosinski et al., 2014), the first H layers of the pre-trained DCNN are copied and remain frozen, and the rest of the layers are randomly initialized. In addition, (Yosinski et al., 2014) fine-tunes the H layers, instead of freezing them, which leads to improved accuracy. Fine-tuning begins with the parameter weights of the source-domain DCNN and modifies them in order to adjust the network to the target domain. A different approach was proposed in (Snoek et al., 2015; Markatopoulou et al., 2015; Oquab et al., 2014) that extends a pre-trained DCNN by one or more fully-connected layers placed on the bottom of the classification layer. Similar transfer learning techniques for DCNNs have also been applied on related domains for example to predict salient areas in natural video (Chaabouni et al., 2016).

MTL methods learn the relations across many tasks together at the same time. The main difference between MTL methods is the way they define task relatedness. Some methods identify shared features between different task and use regularization to model task relatedness (Argyriou, Evgeniou, & Pontil, 2007; Obozinski & Taskar, 2006; Mousavi et al., 2014). Others identify a shared subspace over the task parameters (Evgeniou & Pontil, 2004; Daumé, 2009; Argyriou, Evgeniou, & Pontil, 2008). The methods above make the strong assumption that all tasks are related; some newer methods consider the fact that some tasks may be unrelated. For example, the clustered MTL algorithm (CMTL) (Zhou, Chen, & Ye, 2011a) uses a clustering approach to assign to the same cluster parameters of tasks that lie nearby in terms of their L2 distance. Adaptive MTL (AMTL) (Sun, Chen, Liu, & Wu, 2015) decomposes the task parameters into a low-rank structure that captures task relations, and a group-sparse structure that detects outlier tasks. The GO-MTL algorithm (Kumar & Daume, 2012) (i.e, for Grouping and Overlap in Multi-Task Learning) and the online version of it (Markatopoulou, Mezaris, & Patras, 2016) use a dictionary-based method that allows two tasks from different groups to overlap by having one or more basis in common.

Deep learning is well suited for MTL; in (Yang & Hospedales, 2015) a two-sided neural network that addresses the MTL problem is proposed. Specifically, this method unifies several MTL methods that use a predictor matrix factorization approach, e.g $\boldsymbol{w}^{(t)} = \boldsymbol{L}\boldsymbol{s}^{(t)^{\mathsf{T}}}$ (Kumar & Daume, 2012), in order to learn their parameters using a two-sided neural network. $\boldsymbol{L}$ correspond to the parameter vectors of k latent tasks, while $\boldsymbol{s}^{(t)} \in \mathbb{R}^{1\times k}$ is a task-specific weight vector that contains the coefficients of the linear combination. MTL in deep learning architectures has also been proposed for facial landmark detection (Z. Zhang, Luo, Loy, & Tang, 2014) and human pose estimation (Ouyang, Chu, & Wang, 2014). In (Z. Zhang et al., 2014) the task of facial landmark detection is optimized with the assistance of an arbitrary number of related/auxiliary tasks. This is a special case of the conventional MTL that typically maximizes the performance of all tasks. In this work the two sided neural-network proposed by (Yang & Hospedales, 2015) is modified and extended, for devising a deep learning method suitable for transferring a network that has been originally trained on a large-scale image dataset for concept detection, to a target video dataset and a corresponding new set of target concepts.

4.2.3 Developed approach

4.2.3.1 Problem formulation A video concept detection system needs to learn a number of supervised learning tasks T_{ta}, one for each target concept. Each task t is associated with the training set available for this concept $X^{(t)} = (\boldsymbol{x}_i^{(t)}, y_i^{(t)})_{i=1}^{N_t}$, where $\boldsymbol{x}_i^{(t)} \in \mathbb{R}^d, y_i^{(t)} \in \{\pm 1\}$. When the training set is small, it is common to take a DCNN that has been trained on a large-scale source dataset for T_{so} tasks, and transfer its parameters on a target DCNN to be trained on the target dataset $X = \{X^{(t)}\}_{t=1}^{T_{ta}}$ for a different set of T_{ta} tasks. With respect to the target dataset, the task parameters of related tasks may share similar knowledge, but also concept correlations obtained by the ground-truth annotation provide another source of information regarding the relations between tasks. In this section, considering all the above, we apply three different fine-tuning strategies for transfer learning and we built on the outcomes of our experimental evaluations concerning the effectiveness of each approach, to develop a concept detection method that appends a GO-MTL-like loss to a neural network and incorporates a label-based constraint that considers concept correlations. We minimize the entire network end-to-end using stochastic gradient descent (SGD). We refer to the proposed method as deep multi-task learning with label constraint (DMTL_LC) and we apply it on a transfer learning scenario.

4.2.3.2 Fine-tuning strategies for Transfer Learning In this section we present three fine-tuning strategies (Fig. 13) that can be used for the problem of visual annotation, in order to effectively fine-tune DCNNs D_s that were trained on a large visual dataset for a new target video/image dataset. For this, let D_s denote a pre-trained DCNN, trained on C_s categories using a source dataset, and D_t denote the target DCNN, fine-tuned on C_t categories of a different target dataset. The three studied fine-tuning strategies are as follows:

- FT1-def: Default fine-tuning strategy: This is the typical strategy that modifies the last fully-connected layer of D_s to produce the desired number of outputs C_t, by replacing the last fully-connected layer with a new C_t-dimensional classification fully-connected layer.
- FT2-re: Re-initialization strategy: In this scenario, similar to FT1-def, the last fully-connected layer is replaced by a new C_t-dimensional classification layer. The weights of the last N layers, preceding the classification layer, are also re-initialized (i.e. reset and learned from scratch).
- FT3-ex: Extension strategy: Similar to the previous two strategies, the last fully-connected layer is replaced by a new C_t-dimensional classification fully-connected layer. Subsequently, the network is extended with E fully-connected layers of size L that are placed on the bottom of the modified classification layer. These additional layers are initialized and trained from scratch during fine-tuning, at the same rate as the modified classification layer. One example of a modified network after the insertion of one extension layer for two popular DCNN architectures, is presented in Fig. 14. Regarding the GoogLeNet architecture, which has two additional auxiliary classifiers, an extension layer was also inserted in each of them.

Each fine-tuned network D_t can be used in two different ways to annotate new test keyframes/images with semantic concepts. a) Direct classification: Each test keyframe/image is forward propagated by D_t and the network's output is used as the final class distribution assigned to the keyframe/image. b) D_t is used as feature generator: The training set is forward propagated by the network and the features extracted from one or more layers of D_t are used as feature vectors to subsequently train one supervised classifier (e.g Logistic Regression) per concept. Then, each test keyframe/image is firstly described by the DCNN-based features and subsequently these features serve as input to the trained classifiers.

The used dataset and the conducted experiments for assessing the performance of the aforementioned fine-tuning strategies are presented in Section 4.2.4.1.

4.2.3.3 Deep Multi-task Learning with Label Constraint: DMTL_LC Building on the outcomes of the study reported in the section above, we extend the most efficient transfer learning methodology to further improve its detection accuracy. Figure 15 presents the proposed approach for transferring a pre-trained DCNN network that consists of V_{so} layers (upper part) on a target DCNN to be trained to a target dataset (lower part). Starting with the DCNN trained on the source domain, the first H layers are copied to the target DCNN and fine-tuned on the target dataset. The remaining R layers are completely removed or randomly initialized; consequently, $H + R \leq V_{so}$. Subsequently, the target network can be

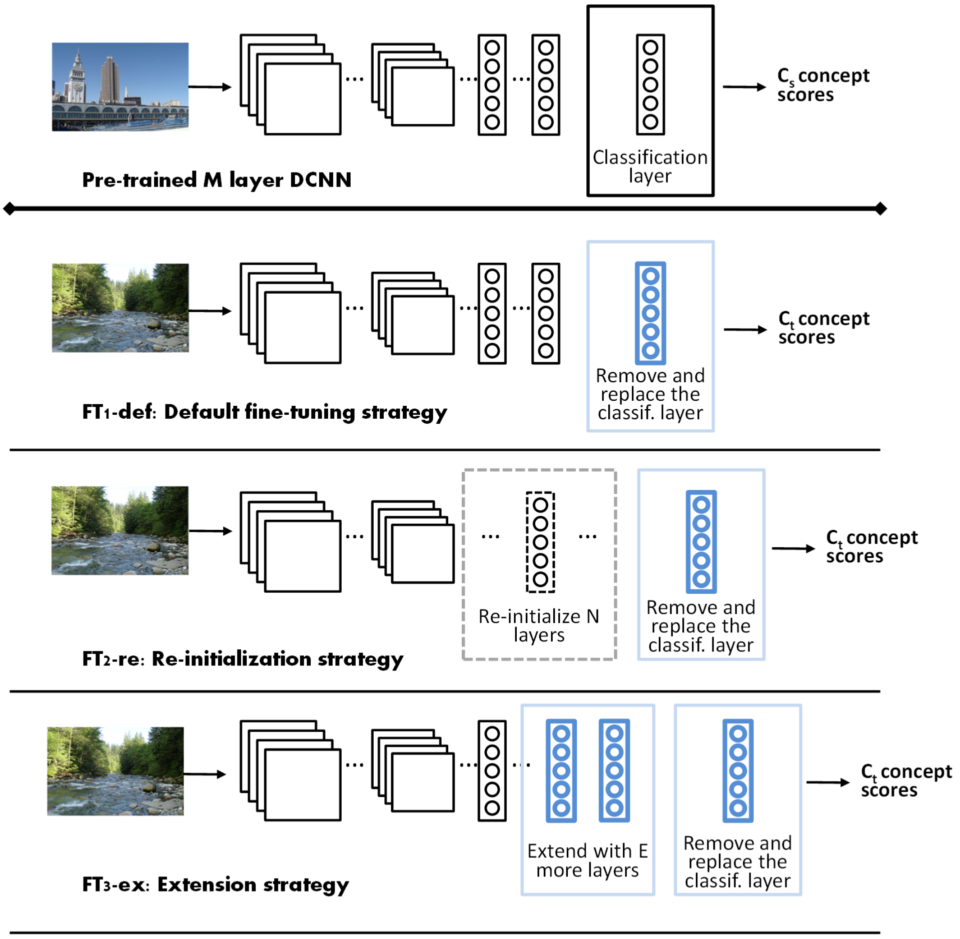

Pre-trained M layer DCNN
Classification layer
C_s concept scores
FT1-def: Default fine-tuning strategy
Remove and replace the classif. layer
C_t concept scores
FT2-re: Re-initialization strategy
Re-initialize N layers
Remove and replace the classif. layer
C_t concept scores
FT3-ex: Extension strategy
Extend with E more layers
Remove and replace the classif. layer
C_t concept scores

Figure 13: Fine-tuning strategies outline.

extended with $E \geq 0$ fully-connected layers, as presented in the previous section. Finally, the target network is trained using the DMTL_LC method.

The DMTL_LC algorithm unifies the GO-MTL algorithm (Kumar & Daume, 2012) in the target DCNN by using and extending the two-sided neural network proposed in (Yang & Hospedales, 2015) as follows: i) The two-sided neural network is placed on the top of the V_{ta}-th fully-connected layer (where $V_{ta} = H+R+E$ is the number of layers before the two-sided network), instead of using it as a standalone network that takes as input hand-crafted or DCNN-based features. ii) The two-sided network is extended with a new label-based constraint in order to incorporate statistical information of pairwise correlations between concepts that we can acquire from the ground-truth annotation.

Specifically, the upper side of the target DCNN in Fig. 15, contains a fully-connected layer $FC_{\boldsymbol{L}}$ that takes as input the output of the V_{ta}-th layer. $FC_{\boldsymbol{L}}$ consists of k neurons, each representing one latent task. The parameter matrix $\boldsymbol{L} \in \mathbb{R}^{d\times k}$ of this layer constitutes a shared knowledge basis for all task models T_{ta}. The concept related to each task t is represented by a semantic descriptor $\boldsymbol{z}^{(t)} \in \{0,1\}^{1\times T_{ta}}$, which is a binary vector of length T_{ta} that has zeros in every position except for position t. The lower side of the target DCNN contains a fully-connected layer $FC_{\boldsymbol{S}}$ that consists of k neurons and takes as input the semantic descriptor $\boldsymbol{z}^{(t)}$. Each row of the parameter matrix $\boldsymbol{S} \in \mathbb{R}^{T_{ta}\times k}$ of this layer contains a task-specific weight vector of the coefficients of the linear combination with the shared basis $\boldsymbol{L}$. This linear combination indicates for each concept which latent tasks describe it. The label-based constraint is placed on the top of the task-specific layer $FC_{\boldsymbol{S}}$. The network predicts a single output, which is equal to $\hat{y}^{(t)} = (\boldsymbol{y}^{(V_{ta})}\mathbf{L})(\boldsymbol{z}^{(t)}\mathbf{S})^{\mathsf{T}}$, where $\boldsymbol{y}^{(V_{ta})}$ is the output of the V_{ta} layer. The higher the output, the more likely that the concept learned w.r.t. task t is depicted in the input keyframe.

The above problem can be formulated by two separate objective functions:

$$\min_{(\mathbf{L},\mathbf{S},f\in F)} \frac{1}{T_{ta}} \sum_{t=1}^{T_{ta}} \left\{ \frac{1}{N_t} \sum_{i=1}^{N_t} \mathscr{L}\left(\hat{y}_i^{(t)}, y_i^{(t)}\right) \right\} \quad (2)$$

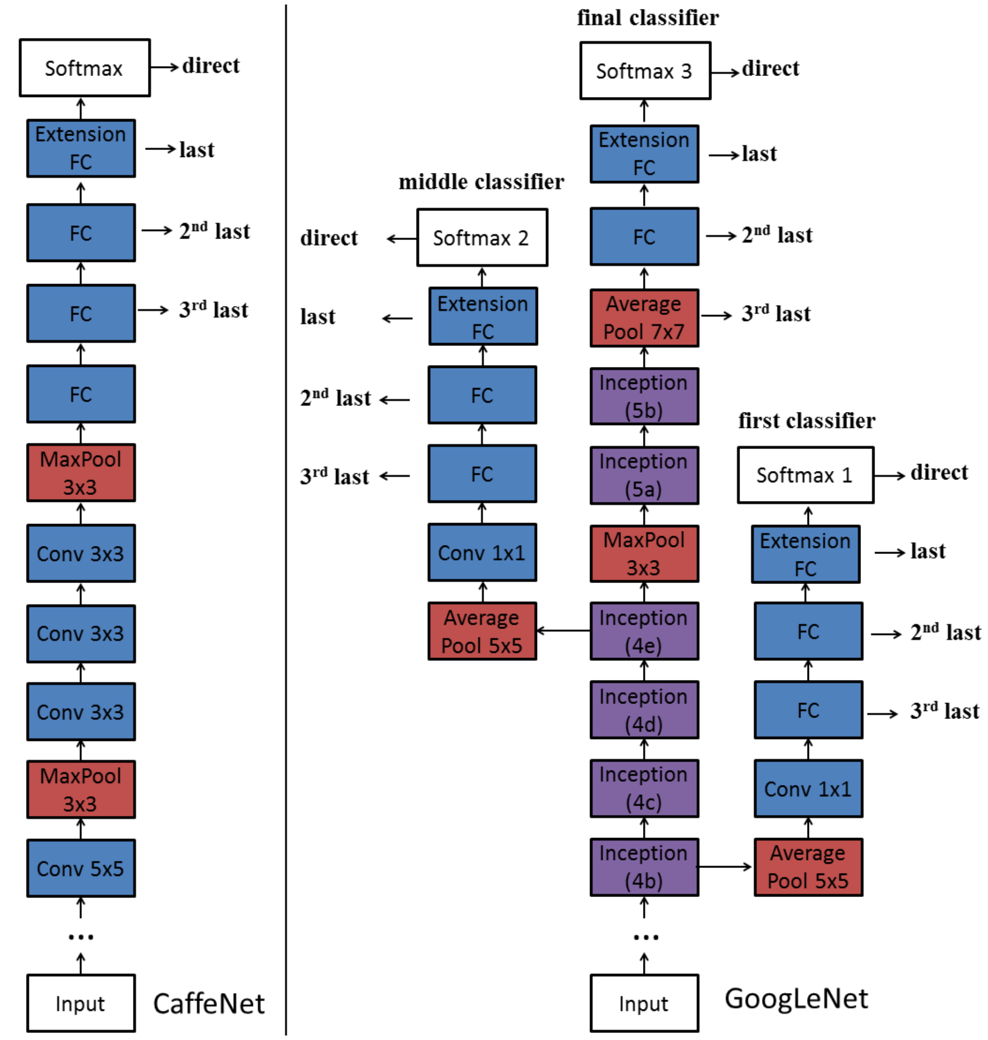


Figure 14: A simplified illustration of the CaffeNet (Krizhevsky et al., 2012) (left) and GoogLeNet (Szegedy et al., 2015) (right) architectures used after insertion of one extension layer. Each of the inception layers of GoogLeNet consists of six convolution layers and one pooling layer. The figure also presents the direct output of each network and the output of the last three layers that were used as features w.r.t. FT3-ex strategy. Similarly, the corresponding layers were used for the FT1-def and FT2-re strategies.

where $\hat{y}_i^{(t)} = (\boldsymbol{y}_i^{(v)}\mathbf{L})(\boldsymbol{z}^{(t)}\mathbf{S})^{\mathsf{T}}$ is the prediction w.r.t task t, and $\boldsymbol{y}_i^{(v)} = \alpha(\boldsymbol{W}^{(v)}\boldsymbol{y}_i^{(v-1)} + \boldsymbol{b}^{(v)})$ is the output of the v-th layer, with α referring to the layer's activation functions. E.g. $\alpha(x) = max(0,x)$ for the ReLU function.

In the above equation $\mathscr{L}$ refers to the loss function calculated between the prediction $\hat{y}_i^{(t)}$ and ground-truth annotation $y_i^{(t)}$. $f^{(v)} = \{\boldsymbol{W}^{(v)}, \boldsymbol{b}^{(v)}\}$ is the pair of the network parameters for the v-th layer and $F = \{f^{(v)}\}_{v=1}^{V_{ta}}$ is the set of network parameters for the first V_{ta} layers.

The second objective function that is placed on the top of the task-specific layer FC_S can be formulated as follows:

$$\min_{\mathbf{S}} \beta \left(\frac{1}{T_{ta}} \sum_{t=1}^{T_{ta}} \left\{ \frac{1}{N_t} \sum_{i=1}^{N_t} \mathscr{L}\left(\hat{\phi}^{(t)}, \phi^{(t)} \right) \right\} \right) \tag{3}$$

The role of this objective function is to approximate the correlation matrix $\boldsymbol{\Phi} \in [-1,1]^{T_{ta} \times T_{ta}}$. Each position of this matrix corresponds to the ϕ-correlation coefficient between two concepts regarding two different tasks t and t', calculated from the ground-truth annotation of the training set. Consequently, $\phi^{(t)} \in [-1,1]^{1 \times T_{ta}}$ refers to the t'th row of $\boldsymbol{\Phi}$ that contains the correlations of task t with all the other tasks. $\hat{\phi}^{(t)} \in \mathbb{R}^{1 \times T_{ta}}$, where $\hat{\phi}^{(t)} = (\boldsymbol{z}^{(t)}\mathbf{S})\mathbf{C}^{\mathsf{T}}$, is the network's prediction for this row. Finally, $\boldsymbol{C} \in \mathbb{R}^{T_{ta} \times k}$ is the weight matrix to train for approximating the correlation matrix. To train $\boldsymbol{C}$, back propagation can be

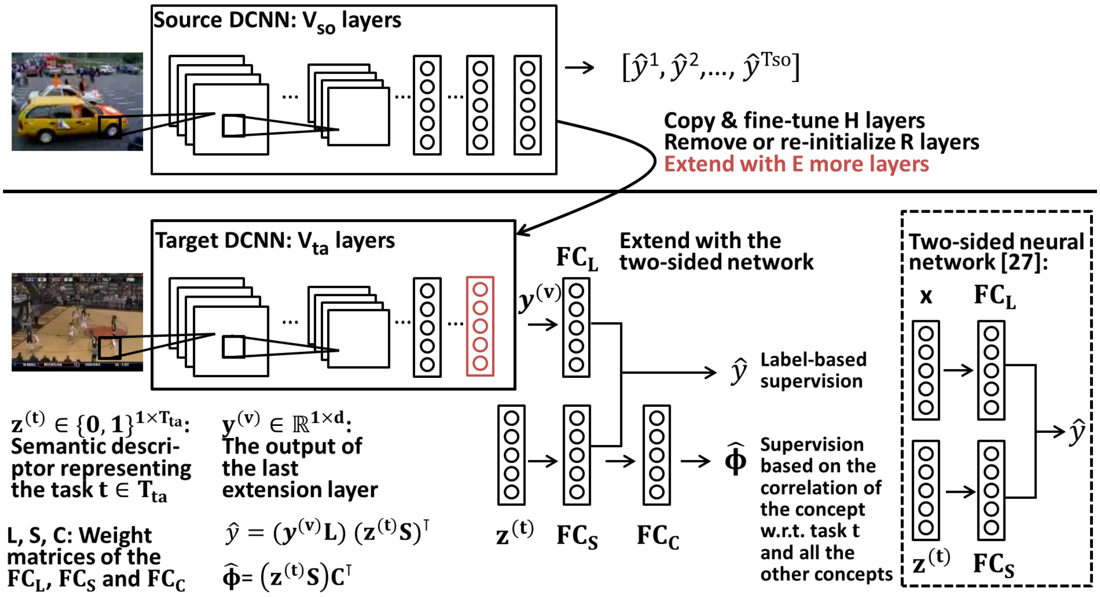

Source DCNN: V_{so} layers
$[\hat{y}^1, \hat{y}^2, ..., \hat{y}^{Tso}]$
Copy & fine-tune H layers
Remove or re-initialize R layers
Extend with E more layers
Target DCNN: V_{ta} layers
FC_L
Extend with the two-sided network
Two-sided neural network [27]:
x
FC_L
$y^{(v)}$
$\hat{y}$ Label-based supervision
$\hat{\Phi}$ Supervision based on the correlation of the concept w.r.t. task t and all the other concepts
$\hat{y}$
$z^{(t)} \in \{0,1\}^{1\times T_{ta}}$: Semantic descriptor representing the task t ∈ T_{ta}
$y^{(v)} \in \mathbb{R}^{1\times d}$: The output of the last extension layer
L, S, C: Weight matrices of the FC_L, FC_S and FC_C
$\hat{y} = (y^{(v)}L)\,(z^{(t)}S)^\intercal$
$\hat{\Phi} = (z^{(t)}S)C^\intercal$
$z^{(t)}$
FC_S
FC_C
$z^{(t)}$
FC_S

Figure 15: Transfer learning using the proposed DMTL_LC method.

performed by the loss $\mathscr{L}$ between $\hat{\phi}^{(t)}$ and $\phi^{(t)}$.

We use the sigmoid cross entropy loss given by the following equation: $\mathscr{L} = \phi log(\sigma(\hat{\phi})) + (1-\phi) log(1-\sigma(\hat{\phi}))$, where $\sigma(.)$ refers to the sigmoid function $\sigma(x) = 1/(1+exp(-x))$. We scale the target vector ϕ in [0,1] in order to deal with the negative values.

This second objective function takes the form of a constraint over the task-specific parameters **S** of the network. Specifically, the rows of the correlation matrix Φ of two correlated concepts will be similar and we want the corresponding rows of S to be similar, too. During training, this second loss (Eq. 3) gets added to the total loss of the network (Eq. 2) with a discount weight β. At inference time, this auxiliary constraint is discarded.

4.2.4 Evaluation

4.2.4.1 Evaluation of fine-tuning strategies for transfer learning

Dataset and experimental setup The TRECVID SIN task 2013 (Over et al., 2013) dataset and the PASCAL VOC-2012 (Everingham, Van Gool, Williams, Winn, & Zisserman, n.d.) dataset were utilized to train and evaluate the concept detection methods presented in the previous sections. The TRECVID SIN dataset consists of low-resolution videos, segmented into video shots; each shot is represented by one keyframe. The dataset is divided into a training and a test set (approx. 600 and 200 hours, respectively). The training set is partially annotated with 346 semantic concepts. The test set is evaluated on 38 concepts, i.e. a subset of the 346 concepts. The PASCAL VOC-2012 (Everingham et al., n.d.) dataset consists of images annotated with one object class label of the 20 available object classes. PASCAL VOC-2012 is divided into training, validation and test sets (consisting of 5717, 5823 and 10991 images, respectively). We used the training set to train the compared methods, and evaluated them on the validation set. We did not use the original test set because ground-truth annotations are not publicly available for it (the evaluation of a method on the test set is possible only through the evaluation server provided by the PASCAL VOC competition, submissions to which are restricted to two per week). The image/video indexing problem was examined; that is, given a concept, we measure how well the top retrieved images/video shots for this concept truly relate to it.

A set of experiments was developed in order to compare the three fine-tuning strategies presented in Section 4.2.3.2. Specifically, in all cases we discarded and replaced the classification fully-connected (fc) layer of the utilized pre-trained network, with a 345-dimensional fc classification layer for the 345 concepts of the TRECVID SIN dataset, or with a 20-dimensional classification layer for the 20 object categories of the PASCAL VOC-2012 dataset. We examined two values for parameter N of the FT2-re strategy; we refer to each configuration as FT2-re1 (for $N=1$) and FT2-re2 (for $N=2$). The FT3-ex strategy was examined for two settings of network extensions $E \in \{1,2\}$: i.e. extending the network by one or two fc layers, respectively, followed by ReLU (Rectified Linear Units) and Dropout layers. The size of each extension layer was examined for 7 different dimensions: $L \in \{64, 128, 256, 512, 1024, 2048, 4096\}$. We refer to these configurations as FT3-exE-L. The new layers' learning rate and momentum was set to 0.01 and $5e{-}4$, whereas the mini-batch size was restricted by our hardware resources and set to 128.

	final classifier					middle classifier		first classifier	
conf / layer	direct	last	2nd last	3rd last	fused	direct	fused	direct	fused
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
FT1-def	22.45	29.60	29.80	-	30.58	23.08	29.41	21.25	26.00
FT2-re1	20.88	28.44	28.43	-	29.58	22.51	28.55	20.37	25.16
FT2-re2	19.08	27.21	27.17	-	28.02	21.73	28.44	20.07	25.74
FT3-ex1-64	25.48	28.86	26.86	29.22	29.62	23.30	28.37	20.20	24.47
FT3-ex1-128	25.52	29.75	28.66	29.57	30.60	23.98	28.82	20.87	25.38
FT3-ex1-256	24.79	30.16	28.99	30.26	31.11	23.62	29.56	21.06	26.32
FT3-ex1-512	24.28	30.86	29.26	29.68	31.47	23.54	29.86	20.71	26.32
FT3-ex1-1024	24.03	31.02	28.78	29.35	31.55	23.43	29.90	20.53	26.57
FT3-ex1-2048	23.37	31.02	27.24	29.37	31.02	23.29	29.94	20.56	26.61
FT3-ex1-4096	23.07	30.91	28.98	29.61	**31.57**	22.85	29.64	20.82	26.26
FT3-ex2-64	16.44	17.51	19.62	19.95	20.09	11.43	15.12	10.65	13.33
FT3-ex2-128	23.87	26.19	26.73	26.05	27.02	18.70	23.64	14.87	19.95
FT3-ex2-256	24.46	28.94	28.69	28.68	29.57	22.68	26.98	18.75	23.10
FT3-ex2-512	23.95	29.44	29.07	28.94	30.14	22.72	28.22	20.20	24.79
FT3-ex2-1024	23.41	30.03	28.80	29.54	30.63	22.79	29.10	19.74	25.68
FT3-ex2-2048	23.38	30.74	28.98	28.21	30.61	22.29	29.34	19.57	26.23
FT3-ex2-4096	23.07	31.21	28.94	27.98	30.93	22.11	29.40	19.64	26.11

Table 11: MXinfAP (%) for GoogLeNet-5k-345-SIN. The best result per column is underlined. The globally best result per sub-table is bold and underlined.

	final classifier					middle classifier		first classifier	
conf / layer	direct	last	2nd last	3rd last	fused	direct	fused	direct	fused
	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)
FT1-def	82.39	86.75	86.74	-	88.01	81.10	84.25	78.96	79.06
FT2-re1	80.50	85.21	86.91	-	87.44	79.58	82.76	77.78	77.23
FT2-re2	77.73	78.81	83.13	-	83.11	75.28	77.34	71.99	69.65
FT3-ex1-64	79.74	82.86	86.41	86.26	86.92	76.36	82.72	72.32	77.51
FT3-ex1-128	80.47	85.50	88.26	86.56	88.12	78.57	84.12	74.01	78.76
FT3-ex1-256	81.43	85.81	88.33	86.73	**88.36**	79.31	84.48	75.29	79.12
FT3-ex1-512	81.65	85.91	87.84	86.90	88.33	79.99	84.76	76.25	79.69
FT3-ex1-1024	82.30	86.48	87.01	86.89	88.20	80.68	84.56	77.32	79.32
FT3-ex1-2048	82.51	86.93	86.80	86.96	88.23	81.15	84.51	77.97	79.62
FT3-ex1-4096	82.39	87.20	86.37	87.05	88.13	81.52	84.45	78.43	79.65
FT3-ex2-64	43.85	45.11	53.99	51.67	52.81	39.10	47.22	32.42	38.72
FT3-ex2-128	75.89	70.96	82.85	83.34	82.51	63.27	72.34	54.45	63.64
FT3-ex2-256	78.94	80.30	86.44	86.43	86.01	69.19	77.67	65.31	72.75
FT3-ex2-512	80.47	82.83	87.56	87.00	87.38	75.17	81.44	66.50	74.38
FT3-ex2-1024	81.47	84.54	86.81	86.53	87.58	76.99	82.85	71.09	76.74
FT3-ex2-2048	82.11	85.49	86.90	86.28	87.76	78.15	83.24	73.55	77.69
FT3-ex2-4096	80.50	83.83	85.82	84.71	86.64	77.49	81.79	74.66	78.21

Table 12: MAP % for GoogleNet-5k-VOC. The best result per column is underlined. The globally best result per sub-table is bold and underlined.

Experimental results Tables 11 and 12 present the results of the three fine-tuning strategies of Section 4.2.3.2 for the TRECVID SIN and PASCAL VOC dataset, respectively. For each dataset we fine-tuned the GoogLeNet-5k, which refers to a DCNN that we trained according to the 22-layer GoogLeNet architecture on the ImageNet "fall" 2011 dataset for 5055 categories. Specifically, GoogLeNet-5k was fine-tuned on the 345 TRECVID SIN concepts (i.e. all the available TRECVID SIN concepts, except for one which was discarded because only 5 positive samples are provided for it). We refer to it as GoogLeNet-5k-345-SIN. In addition, GoogLeNet-5k was fine-tuned on the positive examples of the PASCAL VOC-2012 training set. This network is labeled as GoogLeNet-5k-VOC. For the TRECVID SIN dataset we analyse our results in terms of mean extended inferred average precision (MXinfAP) (Yilmaz, Kanoulas, & Aslam, 2008), which is an approximation of the mean average precision suitable for the partial ground-truth that accompanies the TRECVID dataset (Over et al., 2013). Table 11 presents the MXinfAP of the GoogLeNet-5k-345-SIN and Table 12 presents the results in terms of MAP of the GoogLeNet-5k-VOC.

For each pair of utilized network and fine-tuning strategy we evaluate: i) The direct output of the network (Tables 11 and 12: col. (a)). ii) Logistic regression (LR) classifiers trained on DCNN-based features. Specifically, the output of each of the three last layers of each fine-tuned network was used as feature to train one LR model per concept (Tables 11 and 12: col. (b)-(d)). Furthermore, we present results for the late-fused output (arithmetic mean) of the LR classifiers built using the last three layers (Tables 11 and 12: col. (e)). Evaluations are also reported for the two auxiliary classifiers (Tables 11 and 12: col. (f)-(i)). The details of the two GoogLeNet architecture and the extracted features are also illustrated in Fig. 14. Based on the results reported in the aforementioned tables, we reach the following conclusions:

a) For both datasets, the FT3-ex strategy almost always outperforms the other two fine-tuning strategies (FT1-def, FT2-re) for specific (L, E) values.

b) With respect to the direct output, FT3-ex1-64 and FT3-ex1-128 constitute the top-two methods for the TRECVID SIN dataset. On the other hand, FT3-ex1-2048 and FT3-ex1-4096 are the top-two methods for the PASCAL VOC-2012 dataset. That is, the FT3-ex strategy with one extension layer is always the best solution, but the optimal dimension of the extension layer varies, depending on the target domain dataset.

c) The highest concept detection accuracy for each network is always reached when LR classifiers are trained on features extracted from the last and the second last fully connected layer for TRECVID SIN and PASCAL VOC-2012 dataset, respectively, using the FT3-ex strategy. That is, features extracted from the top layers are more accurate than layers positioned lower in the network, but the optimal layer varies, depending on the target domain dataset.

d) DCNN-based features significantly outperform the direct output alternative in the vast majority of cases. However, in a few cases the direct network output works comparably well. The choice between the two approaches should be based on the application that the DCNN will be used. E.g. real time applications' time and memory limitations would most probably render using DCNNs as feature extractors in conjunction with additional learning (LR or SVMs) prohibitive. Furthermore, we observe that the features extracted from the final classifier of GoogLeNet-based networks outperform the other two auxiliary classifiers, in most cases.

e) Using DCNN layers' responses as feature vectors, on the one hand, FT3-ex1-512 is in the top-five methods irrespective of the extracted feature and the used dataset. On the other hand, FT3-ex2-64 is always among the five worst fine-tuning methods. The rest of the FT3-ex configurations, present fluctuations of their performance across the different utilized DCNN-based features.

f) Finally, it is better to combine features extracted from many layers; specifically, performing late fusion on the output of the LR classifiers trained with each one of the last three fully connected layers almost always outperforms using a single such classifier irrespective of the employed network (Tables 11 and 12: col. (e)). The above conclusion was also reached for the auxiliary classifiers of GoogLeNet-based networks but for space-limitations we only present the fused output for each of these auxiliary classifiers (Tables 11 and 12: col. (g),(i)).

The interested reader can refer to (Pittaras, Markatopoulou, Mezaris, & et al., 2017) for an extensive experimental comparison with more DCNN architectures, different subsets of concepts and different DCNN and fine-tuning learning parameters.

4.2.4.2 Evaluation of the developed DMTL_LC approach for concept detection

Dataset and Experimental Setup Our experiments were performed on the TRECVID 2013 SIN dataset (Over et al., 2013) described in Section 4.2.4.1. In our experiments, we used the 8-layer AlexNet (Krizhevsky et al., 2012) that was trained on 1000 ImageNet categories (Russakovsky et al., 2015) as the source DCNN, and fine-tuned it on the 60 TRECVID SIN concepts. We evaluated all the methods on the test set using the subset of 38 concepts that were also evaluated as part of the TRECVID

2013 SIN task (Over et al., 2013). The video indexing problem was examined; that is, given a concept, we measure how well the top retrieved video shots for this concept truly relate to it. We analyze our results in terms of mean extended inferred average precision (MXinfAP) (Yilmaz et al., 2008).

A first set of experiments was ran, where we examined the three different fine-tuning strategies, in this case using the pre-trained AlexNet (Krizhevsky et al., 2012), to fine-tune a target-DCNN towards the 60 TRECVID SIN concepts (Over et al., 2013): i) The FT1-def approach (Fig. 15: H=7, R = E=0), that copies the first 7 layers (Chatfield et al., 2014; Yosinski et al., 2014; Girshick et al., 2014). ii) The FT3-ex approach (Fig. 15: H=7, R=0, E=1), that copies the first 7 layers and extends the network by one more layer (Snoek et al., 2015; Markatopoulou et al., 2015; Oquab et al., 2014). iii) The FT2-re approach (Fig. 15: H=6, R=1, E=0), that copies the first 6 layers and randomly initializes the 7th layer (Yosinski et al., 2014). For each approach we evaluated a) the typical transfer learning method (Default-TL), also evaluated in Section 4.2.4.1, which replaces the classification layer of AlexNet (Krizhevsky et al., 2012) with a new 60-dimensional layer; b) the proposed DMTL_LC method that uses a two-sided network and considers concept correlations. The H layers, in all cases, were copied and fine-tuned towards the target dataset. To train the proposed method, for each concept, a training set was assembled that included all positive annotated training examples for the given concept, and negatives to a maximum of 15:1 ratio. For the Default-TL method we used the positive examples for each concept following an one-vs-all strategy. Subsequently, we applied each of the fine-tuned networks on the TRECVID keyframes and we evaluated the direct output of each network that corresponds to the class label prediction for 60 categories (Table 13).

We also compared (Table 14) the proposed method with the following ones: i) Single-task learning (STL) using a) Logistic regression (LR), b) LSVM and c) kernel SVM with radial kernel (KSVM). ii) MTL using: a) AMTL (Sun et al., 2015), b) CMTL (Zhou et al., 2011a) and c) the two-sided neural network instantiated with the GO-MTL algorithm (Yang & Hospedales, 2015). We refer to the latter method as 2S-NN. STL refers to the typical approach of training one classifier e.g SVMs, per concept, with features extracted from one or more layers of DCNNs (Markatopoulou, Mezaris, & Patras, 2015; Chatfield et al., 2014), instead of performing the final class label prediction directly, using a softmax/hinge loss layer (Simonyan & Zisserman, 2014; Krizhevsky et al., 2012). To train the compared methods, we applied the pre-trained AlexNet on the TRECVID keyframes and we used as a feature the network's last fully-connected layer (fc8). Subsequently, we used the same training set of positive/negative examples as described above.

Regarding the proposed method, the value of k was set to 157 and the regularization parameter β in Eq. (3) was set to 0.3. These parameters are expected to depend on the dimensionality of the feature space and the number of examples, and according to preliminary experiments seem to work well for the employed dataset. The hinge loss was used in Eq. (2) and a ReLU function was placed on the top of S to encourage sparse models. We used stochastic gradient descent (SGD) with 0.9 momentum and cross-validated the learning rate between 10^{-5} and 10^{-2} by a multiplicative step-size $10^{0.5}$. The Caffe software (Jia et al., 2014) was used for training the DCNN networks on a Tesla K40 GPU. The LibLINEAR library (Fan, Chang, Hsieh, Wang, & Lin, 2008) was used as the source of learning LSVM and LR models and the LibSVM (Chang & Lin, 2011) for learning KSVMs. The MALSAR library (Zhou, Chen, & Ye, 2011b) was used for learning the CMTL (Zhou et al., 2011a) and AMTL (Sun et al., 2015).

Experimental Results Tables 13 and 14 present the results of our experiments in terms of MXinfAP. DMTL is an intermediate version of the proposed DMTL_LC that solves the objective function of DMTL (eq. 2) without using the label constraint of DMTL_LC (eq. 3). In Table 13 we examine the best way of using the layers of the pre-trained AlexNet by comparing three different fine-tuning processes. For completeness, we also report how these processes affect the typical way of transferring learning that replaces the classification layer of AlexNet with a new 60-dimensional classification layer (Default-TL). Based on these results, which refer to the direct output of the fine-tuned networks, we can see that the proposed DMTL_LC performs better than the Default-TL alternative independently of the utilized fine-tuning process, with only one exception in the case of the baseline fine-tuning. Furthermore, adding the label constraint (DMTL_LC) further improves the DMTL method for all of the fine-tuning processes. The proposed DMTL_LC is overall the best performing method, reaching a MXinfAP of 22.60% (Table 13: col(c)). This result is important, considering that the pre-trained AlexNet was used as the source DCNN; by incorporating in our DMTL_LC framework better performing DCNN architectures such as GoogLeNet (Szegedy et al., 2015) instead of AlexNet, further performance gains are expected.

In Table 14 we compare the proposed DMTL_LC method with different STL and MTL methods. The pre-trained AlexNet and the best Default-TL fine-tuned network of Table 13, i.e. Table 13: col. (b), are

Table 13: MXinfAP (%) for 38 concepts, for different fine-tuning processes of the pre-trained 8-layer AlexNet (Krizhevsky et al., 2012) towards the 60 TRECVID SIN concepts (Over et al., 2013): i) FT1-def (Chatfield et al., 2014), (Yosinski et al., 2014), (Girshick et al., 2014). ii) FT3-ex (Snoek et al., 2015), (Markatopoulou et al., 2015), (Oquab et al., 2014). iii) FT2-re (Yosinski et al., 2014). For each approach we evaluate a) the typical transfer learning method (Default-TL) that replaces the classification layer of AlexNet (Krizhevsky et al., 2012) with a new 60-dimensional layer; b) the proposed DMTL, DMTL_LC methods that use a two-sided network. The H AlexNet layers, in all cases, are copied and fine-tuned towards the target dataset.

fine-tuning process	(i) FT1-def (Chatfield et al., 2014), (Yosinski et al., 2014), (Girshick et al., 2014)	(ii) FT3-ex (Snoek et al., 2015), (Markatopoulou et al., 2015), (Oquab et al., 2014)			(iii) FT2-re (Yosinski et al., 2014)	
fine-tuning parameters	(a)	(b) #Neurons for the extension layer: 1096	(c) 2048	(d) 4096	(e) #Neurons for the re-initialization layer 1096	(f) 2048
DefaultTL-Softmax	**16.76**	16.22	15.53	14.79	16.24	16.68
DefaultTL-Hinge	13.26	19.91	19.89	18.76	19.20	15.30
Proposed-DMTL	12.71	15.82	14.89	19.93	18.39	19.47
Proposed-DMTL_LC	15.78	**20.13**	**22.60**	**20.84**	**22.54**	**21.47**

used as the source DCNNs. The proposed DMTL_LC fine-tunes each of these networks towards the 60 TRECVID SIN concepts. To train the other methods, the output of the last fully-connected layer of each source DCNN was used as a feature. Regarding the AlexNet source DCNN, the proposed DMTL_LC is the best performing method, reaching a MXinfAP of 22.60%. We also observe that fine-tuning is a procedure that significantly improves the precision of all the compared methods, by increasing the MXinfAP, when the best Default-TL is used as the source DCNN. As the Default-TL approach does not consider the correlations of the concepts and the relations across tasks, in contrast to the proposed DMTL_LC, the latter reaches the highest performance by fine-tuning once again the former network (MXinfAP equal to 25.04%).

4.2.5 API

The developed module for concept-based video labeling is included in the REST service described in Section 4.1.5, extending its functionality and making it an integrated web service for video fragmentation

Table 14: MXinfAP (%) for 38 concepts for different STL and MTL methods using two pre-trained DCNNs.

	Methods	AlexNet	AlexNet Default-TL (best from Table 13)
	Direct output	-	19.91
STL e.g (Markatopoulou et al., 2015), (Chatfield et al., 2014)	LR	18.57	22.34
	LSVM	20.59	22.21
	KSVM	18.81	21.79
MTL	AMTL (Sun et al., 2015)	20.44	22.21
	CMTL (Zhou et al., 2011a)	18.18	22.38
	2S-NN (Yang & Hospedales, 2015)	20.19	23.12
	Proposed DMTL_LC (Section 4.2.3.3)	**22.60**	**25.04**

and annotation. Building on the outcomes of the video fragmentation component of the service, the concept-based labeling module extracts information about the semantic content of the video at the shot- and sub-shot-level by detecting a number of high-level visual concepts after analyzing one representative keyframe per video fragment. The analysis is based on the algorithm presented in Section 4.2, and the video semantics are described with the help of 323 concepts selected from list of concepts defined in the TRECVID SIN task (Over et al., 2013).

For initiating the analysis for a given video, which now contains its fragmentation into the aforementioned three levels of temporal granularity (i.e. scenes, shots and sub-shots) and its shot- and sub-shot-level concept-based annotation, the user must commit an HTTP POST request on `http://multimedia2.iti.gr:8080/segmentation-annotation`. The body of the HTTP POST request is in JSON format and contains the parameters described in Section 4.1.5.

As before, the communication between the REST service and the user is synchronous only during the transmission of the call, while after getting the analysis request the service responds (in JSON format) informing about the successful receipt of the analysis request and the assigned identifier to the video file, or notifying the user concerning a number of different violated conditions (e.g. wrongly formatted analysis request, total video duration exceeded, or broken video URL) that prevented the initialization of the analysis. Following, the user is again able to get information about the status of the analysis by committing an HTTP GET request on `http://multimedia2.iti.gr:8080/status/<video_id>`, where "video_id" is the automatically assigned identifier to the video file. Finally, through a set of specific HTTP GET requests the user is able to get the extracted keyframes and thumbnails of the video (either on a one-by-one basis or as an entire collection), and the JSON file with the analysis results which, this time, include:

- temporal information about the detected scenes, shots and sub-shots of the video (i.e. their time boundaries in seconds);
- temporal information about the extracted keyframes for each shot and sub-shot (i.e. their timestamps in seconds);
- ordering and temporal information about the extracted thumbnals of the video (i.e. their timestamps in seconds);
- information about the semantic content of each video shot (i.e. a set of confidence scores regarding the existence of 323 concepts in the visual content of the shot and a list with the top-5 concepts);
- information about the semantic content of each video sub-shot (i.e. a set of confidence scores regarding the existence of 323 concepts in the visual content of the sub-shot and a list with the top-5 concepts).

5 Outlook and Next Steps

Our story detection work will seek to find further solutions for generating disambiguated story clusters automatically, looking both at issues with merged stories and split stories that were observed using the evaluation. Firstly, the upgrade of keyword extraction capabilities to Named Entity Keywords (NEKs) is expected to support this effort significantly, as keywords with different syntactic forms can be aligned to the same semantic entity. Furthermore, these entities will be typed and relations may be determined that hold between them (and stored in our Semantic Knowledge Base for future reference). It will be explored how this additional semantic knowledge could be applied to story labelling in order to support further disambiguation of stories as well as construction of more appropriate queries to social networks for related content for those stories. Having a semantic model to describe each news story can support more than just more relevant content retrieval from social networks, for example the location detection service we develop in WP3 could enable us to geolocate each story to a bounding box within which the events of the story are taking place and then dynamically search for tweets around that location while the story is 'active'.

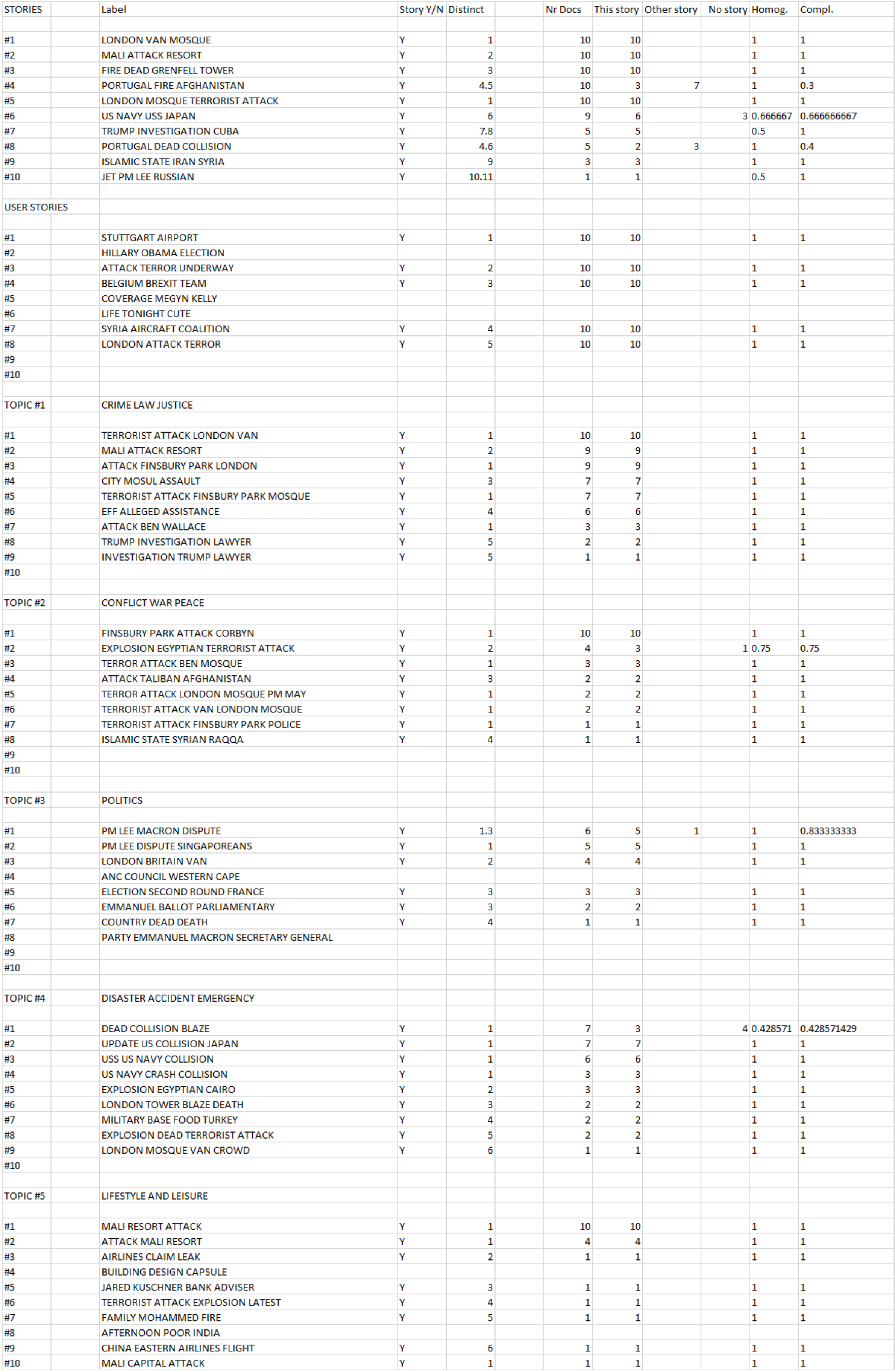
The social media extraction and annotation task will continue to expand its coverage of stories and sources. The next significant update will be to switch the dynamic input filter from keywords to story labels. We have shown in our evaluation that story labels provide enough accuracy for query formulation and retrieve relevant documents. Queries will be constructed for each of the news topics we have defined to ensure a greater breadth of stories are covered by the video content. We also plan to introduce support for detecting links to videos on other platforms within the collected documents, e.g. a tweet may link to a YouTube video rather than use a native platform for sharing the media. This is the only way to find e.g. Facebook Videos for a news story as Facebook does not support a public video search API. Finally, the metadata model will be extended with two new metrics which need to be calculated on the platform itself (as opposed to retrieved via API from the source) - reach and authoritativeness. Both will support a user in browsing the retrieved content according to their interests (popularity, verifiability).

With regards to video annotation, the fragmentation and thumbnail extraction service is live and being integrated with the platform. The reportings on the development and evaluation of the InVID algorithms for temporal fragmentation of single-shot videos and the extraction of representative keyframes/ thumbnails indicate that the definition of a precise and fine-grained segmentation of such videos is a challenging task. The experimental findings show that the implemented InVID solutions outperform other approaches, while significant progress has been made in terms of processing time. In particular, the implemented DCT-based approach is > 20 times faster than the initial, motion-based one (being also the fastest one among the compared techniques), and its response time was judged as satisfactory by the participants in the pilot testing of WP7. Nevertheless, there is still a lot of room for improvement in terms of detection accuracy. For this purpose we plan to extract and describe the motion activity over sequences of frames through a more intelligent method that indicates the optical flow in wisely selected parts of each frame, and to combine this information with data about the color of each frame (represented with the help of the DCT descriptors) via a fusion mechanism; the latter will allow the detection of different camera activities and the identification of visually distinct video fragments produced when a moving camera follows a moving object. Last but not least, based on the feedback from the pilot testing regarding the appropriateness and usability of the extracted keyframes/thumbnails of the video, we plan to extend the video fragmentation pipeline by integrating an analysis step for a) filtering-out blurred or blank keyframes, and ii) selecting keyframes that satisfy particular users' needs (e.g. keyframes showing intense activity or depicting an as wide as possible instance of a captured scene).

We also presented a developed deep multi-task learning method for video concept detection. The reported experiments reveal the usefulness of fine-tuning a deep network by directly learning the relations between many task models (one per concept) in combination with the concept correlations that can be captured from the ground-truth annotation. Based on these findings we will further extend our DMTL_LC method in order to scale for more concepts, and we will improve its detection accuracy. We also aim to reduce the needed processing time for both training and classification phase, by replacing the current two-sided network with a single-side one.

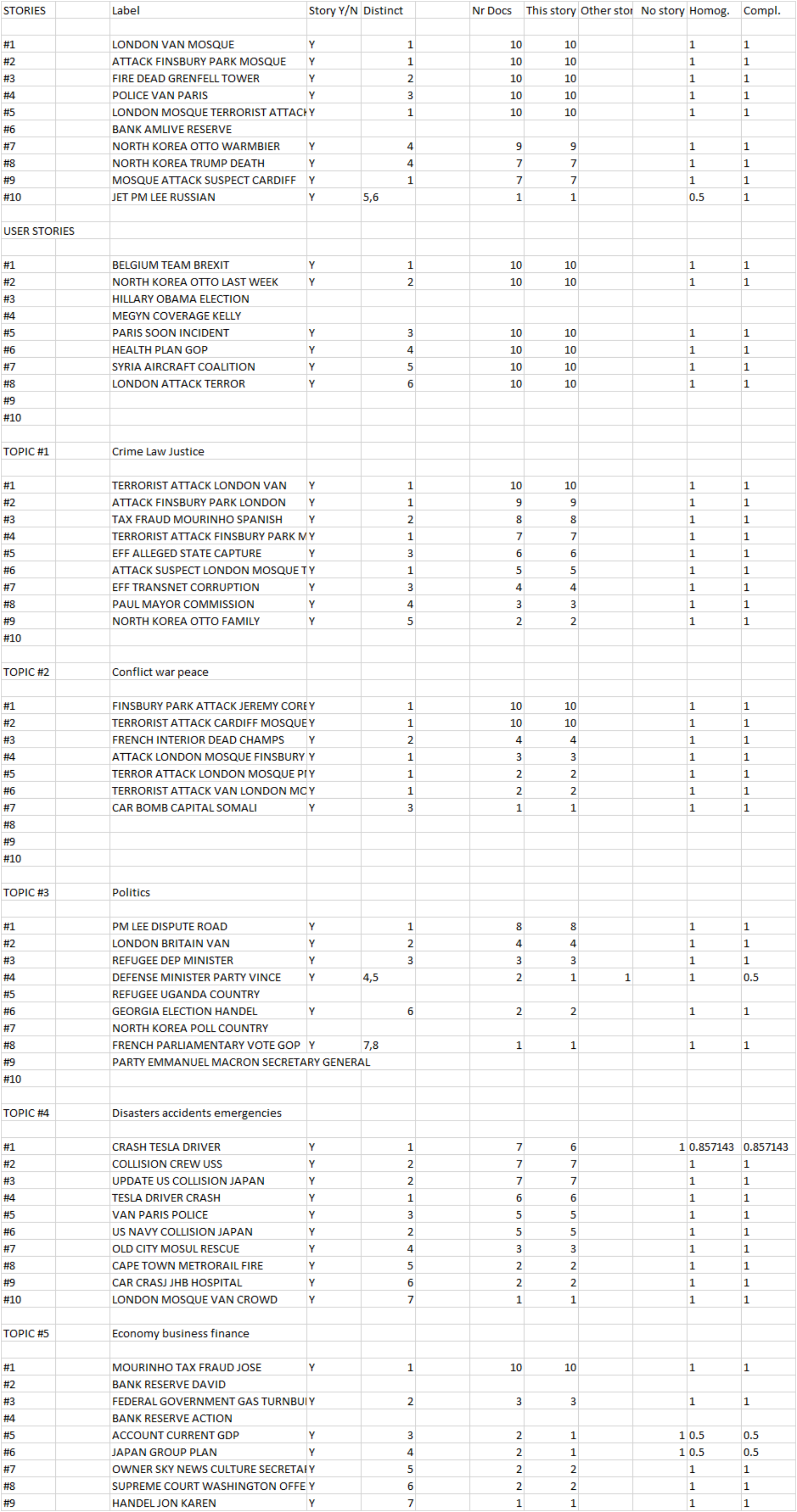
The evaluations presented in this deliverable serve as valuable benchmarks so that we can measure the further improvement in our approaches achieved by the next milestone, when we hope to report on further improved story detection, social media extraction and annotation, and video annotation - all supporting the journalist in their search for appropriate online video materials for news exploration and reporting.

6 Appendix A: Story detection evaluation



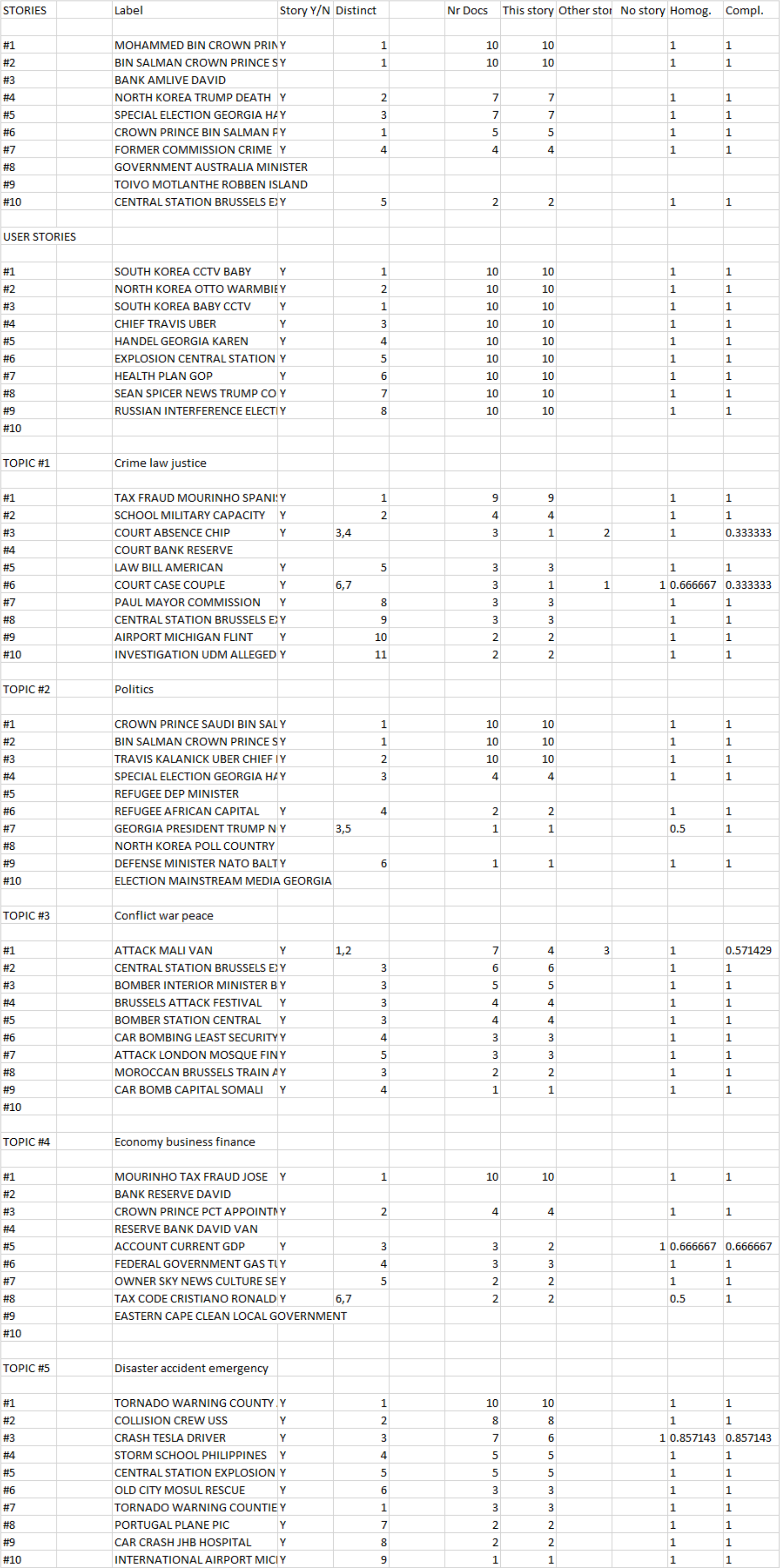
STORIES		Label	Story Y/N	Distinct		Nr Docs	This story	Other story	No story	Homog.	Compl.
#1		LONDON VAN MOSQUE	Y	1		10	10			1	1
#2		MALI ATTACK RESORT	Y	2		10	10			1	1
#3		FIRE DEAD GRENFELL TOWER	Y	3		10	10			1	1
#4		PORTUGAL FIRE AFGHANISTAN	Y	4.5		10	3	7		1	0.3
#5		LONDON MOSQUE TERRORIST ATTACK	Y	1		10	10			1	1
#6		US NAVY USS JAPAN	Y	6		9	6		3	0.666667	0.666666667
#7		TRUMP INVESTIGATION CUBA	Y	7.8		5	5			0.5	1
#8		PORTUGAL DEAD COLLISION	Y	4.6		5	2	3		1	0.4
#9		ISLAMIC STATE IRAN SYRIA	Y	9		3	3			1	1
#10		JET PM LEE RUSSIAN	Y	10.11		1	1			0.5	1
USER STORIES											
#1		STUTTGART AIRPORT	Y	1		10	10			1	1
#2		HILLARY OBAMA ELECTION									
#3		ATTACK TERROR UNDERWAY	Y	2		10	10			1	1
#4		BELGIUM BREXIT TEAM	Y	3		10	10			1	1
#5		COVERAGE MEGYN KELLY									
#6		LIFE TONIGHT CUTE									
#7		SYRIA AIRCRAFT COALITION	Y	4		10	10			1	1
#8		LONDON ATTACK TERROR	Y	5		10	10			1	1
#9											
#10											
TOPIC #1		CRIME LAW JUSTICE									
#1		TERRORIST ATTACK LONDON VAN	Y	1		10	10			1	1
#2		MALI ATTACK RESORT	Y	2		9	9			1	1
#3		ATTACK FINSBURY PARK LONDON	Y	1		9	9			1	1
#4		CITY MOSUL ASSAULT	Y	3		7	7			1	1
#5		TERRORIST ATTACK FINSBURY PARK MOSQUE	Y	1		7	7			1	1
#6		EFF ALLEGED ASSISTANCE	Y	4		6	6			1	1
#7		ATTACK BEN WALLACE	Y	1		3	3			1	1
#8		TRUMP INVESTIGATION LAWYER	Y	5		2	2			1	1
#9		INVESTIGATION TRUMP LAWYER	Y	5		1	1			1	1
#10											
TOPIC #2		CONFLICT WAR PEACE									
#1		FINSBURY PARK ATTACK CORBYN	Y	1		10	10			1	1
#2		EXPLOSION EGYPTIAN TERRORIST ATTACK	Y	2		4	3		1	0.75	0.75
#3		TERROR ATTACK BEN MOSQUE	Y	1		3	3			1	1
#4		ATTACK TALIBAN AFGHANISTAN	Y	3		2	2			1	1
#5		TERROR ATTACK LONDON MOSQUE PM MAY	Y	1		2	2			1	1
#6		TERRORIST ATTACK VAN LONDON MOSQUE	Y	1		2	2			1	1
#7		TERRORIST ATTACK FINSBURY PARK POLICE	Y	1		1	1			1	1
#8		ISLAMIC STATE SYRIAN RAQQA	Y	4		1	1			1	1
#9											
#10											
TOPIC #3		POLITICS									
#1		PM LEE MACRON DISPUTE	Y	1.3		6	5	1		1	0.833333333
#2		PM LEE DISPUTE SINGAPOREANS	Y	1		5	5			1	1
#3		LONDON BRITAIN VAN	Y	2		4	4			1	1
#4		ANC COUNCIL WESTERN CAPE									
#5		ELECTION SECOND ROUND FRANCE	Y	3		3	3			1	1
#6		EMMANUEL BALLOT PARLIAMENTARY	Y	3		2	2			1	1
#7		COUNTRY DEAD DEATH	Y	4		1	1			1	1
#8		PARTY EMMANUEL MACRON SECRETARY GENERAL									
#9											
#10											
TOPIC #4		DISASTER ACCIDENT EMERGENCY									
#1		DEAD COLLISION BLAZE	Y	1		7	3		4	0.428571	0.428571429
#2		UPDATE US COLLISION JAPAN	Y	1		7	7			1	1
#3		USS US NAVY COLLISION	Y	1		6	6			1	1
#4		US NAVY CRASH COLLISION	Y	1		3	3			1	1
#5		EXPLOSION EGYPTIAN CAIRO	Y	2		3	3			1	1
#6		LONDON TOWER BLAZE DEATH	Y	3		2	2			1	1
#7		MILITARY BASE FOOD TURKEY	Y	4		2	2			1	1
#8		EXPLOSION DEAD TERRORIST ATTACK	Y	5		2	2			1	1
#9		LONDON MOSQUE VAN CROWD	Y	6		1	1			1	1
#10											
TOPIC #5		LIFESTYLE AND LEISURE									
#1		MALI RESORT ATTACK	Y	1		10	10			1	1
#2		ATTACK MALI RESORT	Y	1		4	4			1	1
#3		AIRLINES CLAIM LEAK	Y	2		1	1			1	1
#4		BUILDING DESIGN CAPSULE									
#5		JARED KUSCHNER BANK ADVISER	Y	3		1	1			1	1
#6		TERRORIST ATTACK EXPLOSION LATEST	Y	4		1	1			1	1
#7		FAMILY MOHAMMED FIRE	Y	5		1	1			1	1
#8		AFTERNOON POOR INDIA									
#9		CHINA EASTERN AIRLINES FLIGHT	Y	6		1	1			1	1
#10		MALI CAPITAL ATTACK	Y	1		1	1			1	1

Figure 16: Story detection on 19 June 2017.



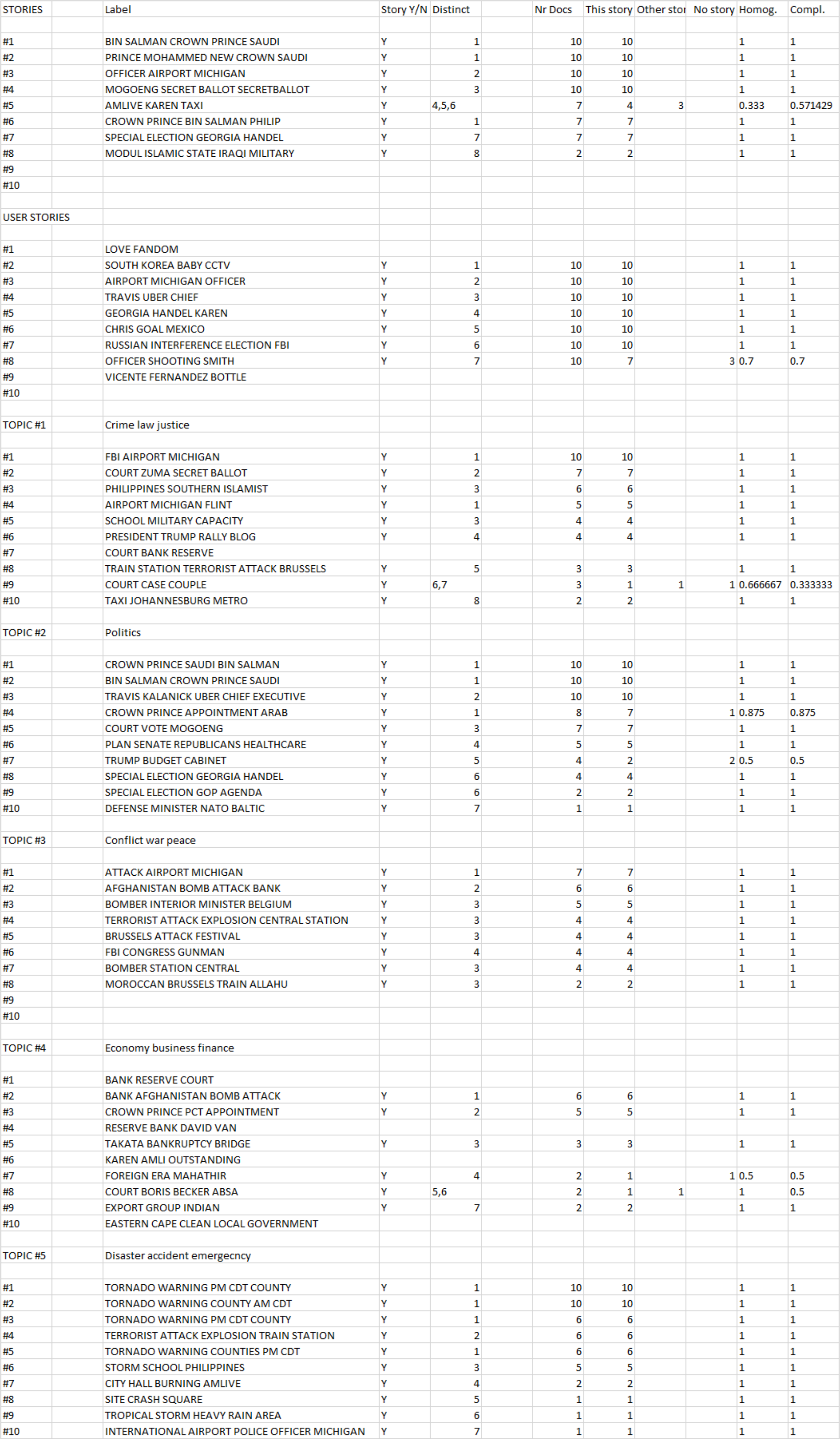
STORIES		Label	Story Y/N	Distinct		Nr Docs	This story	Other stor	No story	Homog.	Compl.
#1		LONDON VAN MOSQUE	Y	1		10	10			1	1
#2		ATTACK FINSBURY PARK MOSQUE	Y	1		10	10			1	1
#3		FIRE DEAD GRENFELL TOWER	Y	2		10	10			1	1
#4		POLICE VAN PARIS	Y	3		10	10			1	1
#5		LONDON MOSQUE TERRORIST ATTACK	Y	1		10	10			1	1
#6		BANK AMLIVE RESERVE									
#7		NORTH KOREA OTTO WARMBIER	Y	4		9	9			1	1
#8		NORTH KOREA TRUMP DEATH	Y	4		7	7			1	1
#9		MOSQUE ATTACK SUSPECT CARDIFF	Y	1		7	7			1	1
#10		JET PM LEE RUSSIAN	Y	5,6		1	1			0.5	1
USER STORIES											
#1		BELGIUM TEAM BREXIT	Y	1		10	10			1	1
#2		NORTH KOREA OTTO LAST WEEK	Y	2		10	10			1	1
#3		HILLARY OBAMA ELECTION									
#4		MEGYN COVERAGE KELLY									
#5		PARIS SOON INCIDENT	Y	3		10	10			1	1
#6		HEALTH PLAN GOP	Y	4		10	10			1	1
#7		SYRIA AIRCRAFT COALITION	Y	5		10	10			1	1
#8		LONDON ATTACK TERROR	Y	6		10	10			1	1
#9											
#10											
TOPIC #1		Crime Law Justice									
#1		TERRORIST ATTACK LONDON VAN	Y	1		10	10			1	1
#2		ATTACK FINSBURY PARK LONDON	Y	1		9	9			1	1
#3		TAX FRAUD MOURINHO SPANISH	Y	2		8	8			1	1
#4		TERRORIST ATTACK FINSBURY PARK M	Y	1		7	7			1	1
#5		EFF ALLEGED STATE CAPTURE	Y	3		6	6			1	1
#6		ATTACK SUSPECT LONDON MOSQUE T	Y	1		5	5			1	1
#7		EFF TRANSNET CORRUPTION	Y	3		4	4			1	1
#8		PAUL MAYOR COMMISSION	Y	4		3	3			1	1
#9		NORTH KOREA OTTO FAMILY	Y	5		2	2			1	1
#10											
TOPIC #2		Conflict war peace									
#1		FINSBURY PARK ATTACK JEREMY CORE	Y	1		10	10			1	1
#2		TERRORIST ATTACK CARDIFF MOSQUE	Y	1		10	10			1	1
#3		FRENCH INTERIOR DEAD CHAMPS	Y	2		4	4			1	1
#4		ATTACK LONDON MOSQUE FINSBURY	Y	1		3	3			1	1
#5		TERROR ATTACK LONDON MOSQUE PI	Y	1		2	2			1	1
#6		TERRORIST ATTACK VAN LONDON MC	Y	1		2	2			1	1
#7		CAR BOMB CAPITAL SOMALI	Y	3		1	1			1	1
#8											
#9											
#10											
TOPIC #3		Politics									
#1		PM LEE DISPUTE ROAD	Y	1		8	8			1	1
#2		LONDON BRITAIN VAN	Y	2		4	4			1	1
#3		REFUGEE DEP MINISTER	Y	3		3	3			1	1
#4		DEFENSE MINISTER PARTY VINCE	Y	4,5		2	1	1		1	0.5
#5		REFUGEE UGANDA COUNTRY									
#6		GEORGIA ELECTION HANDEL	Y	6		2	2			1	1
#7		NORTH KOREA POLL COUNTRY									
#8		FRENCH PARLIAMENTARY VOTE GOP	Y	7,8		1	1			1	1
#9		PARTY EMMANUEL MACRON SECRETARY GENERAL									
#10											
TOPIC #4		Disasters accidents emergencies									
#1		CRASH TESLA DRIVER	Y	1		7	6		1	0.857143	0.857143
#2		COLLISION CREW USS	Y	2		7	7			1	1
#3		UPDATE US COLLISION JAPAN	Y	2		7	7			1	1
#4		TESLA DRIVER CRASH	Y	1		6	6			1	1
#5		VAN PARIS POLICE	Y	3		5	5			1	1
#6		US NAVY COLLISION JAPAN	Y	2		5	5			1	1
#7		OLD CITY MOSUL RESCUE	Y	4		3	3			1	1
#8		CAPE TOWN METRORAIL FIRE	Y	5		2	2			1	1
#9		CAR CRASJ JHB HOSPITAL	Y	6		2	2			1	1
#10		LONDON MOSQUE VAN CROWD	Y	7		1	1			1	1
TOPIC #5		Economy business finance									
#1		MOURINHO TAX FRAUD JOSE	Y	1		10	10			1	1
#2		BANK RESERVE DAVID									
#3		FEDERAL GOVERNMENT GAS TURNBUI	Y	2		3	3			1	1
#4		BANK RESERVE ACTION									
#5		ACCOUNT CURRENT GDP	Y	3		2	1		1	0.5	0.5
#6		JAPAN GROUP PLAN	Y	4		2	1		1	0.5	0.5
#7		OWNER SKY NEWS CULTURE SECRETAI	Y	5		2	2			1	1
#8		SUPREME COURT WASHINGTON OFFE	Y	6		2	2			1	1
#9		HANDEL JON KAREN	Y	7		1	1			1	1

Figure 17: Story detection on 20 June 2017.



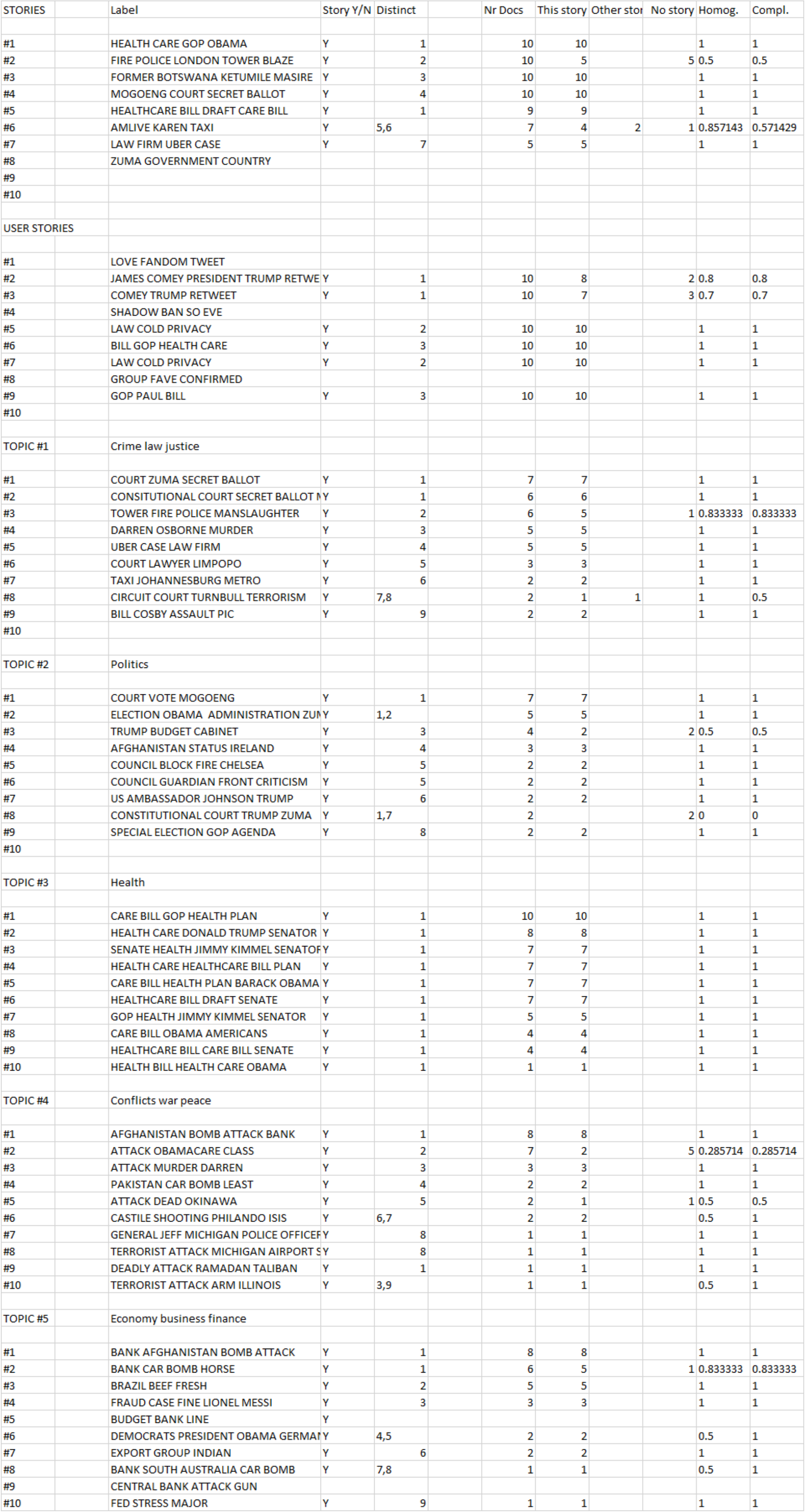
STORIES		Label	Story Y/N	Distinct		Nr Docs	This story	Other stor	No story	Homog.	Compl.
#1		MOHAMMED BIN CROWN PRIN	Y	1		10	10			1	1
#2		BIN SALMAN CROWN PRINCE S	Y	1		10	10			1	1
#3		BANK AMLIVE DAVID									
#4		NORTH KOREA TRUMP DEATH	Y	2		7	7			1	1
#5		SPECIAL ELECTION GEORGIA HA	Y	3		7	7			1	1
#6		CROWN PRINCE BIN SALMAN F	Y	1		5	5			1	1
#7		FORMER COMMISSION CRIME	Y	4		4	4			1	1
#8		GOVERNMENT AUSTRALIA MINISTER									
#9		TOIVO MOTLANTHE ROBBEN ISLAND									
#10		CENTRAL STATION BRUSSELS E)	Y	5		2	2			1	1
USER STORIES											
#1		SOUTH KOREA CCTV BABY	Y	1		10	10			1	1
#2		NORTH KOREA OTTO WARMBIE	Y	2		10	10			1	1
#3		SOUTH KOREA BABY CCTV	Y	1		10	10			1	1
#4		CHIEF TRAVIS UBER	Y	3		10	10			1	1
#5		HANDEL GEORGIA KAREN	Y	4		10	10			1	1
#6		EXPLOSION CENTRAL STATION	Y	5		10	10			1	1
#7		HEALTH PLAN GOP	Y	6		10	10			1	1
#8		SEAN SPICER NEWS TRUMP CO	Y	7		10	10			1	1
#9		RUSSIAN INTERFERENCE ELECTI	Y	8		10	10			1	1
#10											
TOPIC #1		Crime law justice									
#1		TAX FRAUD MOURINHO SPANI	Y	1		9	9			1	1
#2		SCHOOL MILITARY CAPACITY	Y	2		4	4			1	1
#3		COURT ABSENCE CHIP	Y	3,4		3	1	2		1	0.333333
#4		COURT BANK RESERVE									
#5		LAW BILL AMERICAN	Y	5		3	3			1	1
#6		COURT CASE COUPLE	Y	6,7		3	1	1	1	0.666667	0.333333
#7		PAUL MAYOR COMMISSION	Y	8		3	3			1	1
#8		CENTRAL STATION BRUSSELS E)	Y	9		3	3			1	1
#9		AIRPORT MICHIGAN FLINT	Y	10		2	2			1	1
#10		INVESTIGATION UDM ALLEGED	Y	11		2	2			1	1
TOPIC #2		Politics									
#1		CROWN PRINCE SAUDI BIN SAL	Y	1		10	10			1	1
#2		BIN SALMAN CROWN PRINCE S	Y	1		10	10			1	1
#3		TRAVIS KALANICK UBER CHIEF I	Y	2		10	10			1	1
#4		SPECIAL ELECTION GEORGIA HA	Y	3		4	4			1	1
#5		REFUGEE DEP MINISTER									
#6		REFUGEE AFRICAN CAPITAL	Y	4		2	2			1	1
#7		GEORGIA PRESIDENT TRUMP N	Y	3,5		1	1			0.5	1
#8		NORTH KOREA POLL COUNTRY									
#9		DEFENSE MINISTER NATO BALT	Y	6		1	1			1	1
#10		ELECTION MAINSTREAM MEDIA GEORGIA									
TOPIC #3		Conflict war peace									
#1		ATTACK MALI VAN	Y	1,2		7	4	3		1	0.571429
#2		CENTRAL STATION BRUSSELS E)	Y	3		6	6			1	1
#3		BOMBER INTERIOR MINISTER B	Y	3		5	5			1	1
#4		BRUSSELS ATTACK FESTIVAL	Y	3		4	4			1	1
#5		BOMBER STATION CENTRAL	Y	3		4	4			1	1
#6		CAR BOMBING LEAST SECURITY	Y	4		3	3			1	1
#7		ATTACK LONDON MOSQUE FIN	Y	5		3	3			1	1
#8		MOROCCAN BRUSSELS TRAIN A	Y	3		2	2			1	1
#9		CAR BOMB CAPITAL SOMALI	Y	4		1	1			1	1
#10											
TOPIC #4		Economy business finance									
#1		MOURINHO TAX FRAUD JOSE	Y	1		10	10			1	1
#2		BANK RESERVE DAVID									
#3		CROWN PRINCE PCT APPOINTN	Y	2		4	4			1	1
#4		RESERVE BANK DAVID VAN									
#5		ACCOUNT CURRENT GDP	Y	3		3	2		1	0.666667	0.666667
#6		FEDERAL GOVERNMENT GAS TL	Y	4		3	3			1	1
#7		OWNER SKY NEWS CULTURE SE	Y	5		2	2			1	1
#8		TAX CODE CRISTIANO RONALD	Y	6,7		2	2			0.5	1
#9		EASTERN CAPE CLEAN LOCAL GOVERNMENT									
#10											
TOPIC #5		Disaster accident emergency									
#1		TORNADO WARNING COUNTY	Y	1		10	10			1	1
#2		COLLISION CREW USS	Y	2		8	8			1	1
#3		CRASH TESLA DRIVER	Y	3		7	6		1	0.857143	0.857143
#4		STORM SCHOOL PHILIPPINES	Y	4		5	5			1	1
#5		CENTRAL STATION EXPLOSION	Y	5		5	5			1	1
#6		OLD CITY MOSUL RESCUE	Y	6		3	3			1	1
#7		TORNADO WARNING COUNTIE	Y	1		3	3			1	1
#8		PORTUGAL PLANE PIC	Y	7		2	2			1	1
#9		CAR CRASH JHB HOSPITAL	Y	8		2	2			1	1
#10		INTERNATIONAL AIRPORT MICI	Y	9		1	1			1	1

Figure 18: Story detection on 21 June 2017.



STORIES		Label	Story Y/N	Distinct		Nr Docs	This story	Other stor	No story	Homog.	Compl.
#1		BIN SALMAN CROWN PRINCE SAUDI	Y	1		10	10			1	1
#2		PRINCE MOHAMMED NEW CROWN SAUDI	Y	1		10	10			1	1
#3		OFFICER AIRPORT MICHIGAN	Y	2		10	10			1	1
#4		MOGOENG SECRET BALLOT SECRETBALLOT	Y	3		10	10			1	1
#5		AMLIVE KAREN TAXI	Y	4,5,6		7	4	3		0.333	0.571429
#6		CROWN PRINCE BIN SALMAN PHILIP	Y	1		7	7			1	1
#7		SPECIAL ELECTION GEORGIA HANDEL	Y	7		7	7			1	1
#8		MODUL ISLAMIC STATE IRAQI MILITARY	Y	8		2	2			1	1
#9											
#10											
USER STORIES											
#1		LOVE FANDOM									
#2		SOUTH KOREA BABY CCTV	Y	1		10	10			1	1
#3		AIRPORT MICHIGAN OFFICER	Y	2		10	10			1	1
#4		TRAVIS UBER CHIEF	Y	3		10	10			1	1
#5		GEORGIA HANDEL KAREN	Y	4		10	10			1	1
#6		CHRIS GOAL MEXICO	Y	5		10	10			1	1
#7		RUSSIAN INTERFERENCE ELECTION FBI	Y	6		10	10			1	1
#8		OFFICER SHOOTING SMITH	Y	7		10	7		3	0.7	0.7
#9		VICENTE FERNANDEZ BOTTLE									
#10											
TOPIC #1		Crime law justice									
#1		FBI AIRPORT MICHIGAN	Y	1		10	10			1	1
#2		COURT ZUMA SECRET BALLOT	Y	2		7	7			1	1
#3		PHILIPPINES SOUTHERN ISLAMIST	Y	3		6	6			1	1
#4		AIRPORT MICHIGAN FLINT	Y	1		5	5			1	1
#5		SCHOOL MILITARY CAPACITY	Y	3		4	4			1	1
#6		PRESIDENT TRUMP RALLY BLOG	Y	4		4	4			1	1
#7		COURT BANK RESERVE									
#8		TRAIN STATION TERRORIST ATTACK BRUSSELS	Y	5		3	3			1	1
#9		COURT CASE COUPLE	Y	6,7		3	1	1	1	0.666667	0.333333
#10		TAXI JOHANNESBURG METRO	Y	8		2	2			1	1
TOPIC #2		Politics									
#1		CROWN PRINCE SAUDI BIN SALMAN	Y	1		10	10			1	1
#2		BIN SALMAN CROWN PRINCE SAUDI	Y	1		10	10			1	1
#3		TRAVIS KALANICK UBER CHIEF EXECUTIVE	Y	2		10	10			1	1
#4		CROWN PRINCE APPOINTMENT ARAB	Y	1		8	7		1	0.875	0.875
#5		COURT VOTE MOGOENG	Y	3		7	7			1	1
#6		PLAN SENATE REPUBLICANS HEALTHCARE	Y	4		5	5			1	1
#7		TRUMP BUDGET CABINET	Y	5		4	2		2	0.5	0.5
#8		SPECIAL ELECTION GEORGIA HANDEL	Y	6		4	4			1	1
#9		SPECIAL ELECTION GOP AGENDA	Y	6		2	2			1	1
#10		DEFENSE MINISTER NATO BALTIC	Y	7		1	1			1	1
TOPIC #3		Conflict war peace									
#1		ATTACK AIRPORT MICHIGAN	Y	1		7	7			1	1
#2		AFGHANISTAN BOMB ATTACK BANK	Y	2		6	6			1	1
#3		BOMBER INTERIOR MINISTER BELGIUM	Y	3		5	5			1	1
#4		TERRORIST ATTACK EXPLOSION CENTRAL STATION	Y	3		4	4			1	1
#5		BRUSSELS ATTACK FESTIVAL	Y	3		4	4			1	1
#6		FBI CONGRESS GUNMAN	Y	4		4	4			1	1
#7		BOMBER STATION CENTRAL	Y	3		4	4			1	1
#8		MOROCCAN BRUSSELS TRAIN ALLAHU	Y	3		2	2			1	1
#9											
#10											
TOPIC #4		Economy business finance									
#1		BANK RESERVE COURT									
#2		BANK AFGHANISTAN BOMB ATTACK	Y	1		6	6			1	1
#3		CROWN PRINCE PCT APPOINTMENT	Y	2		5	5			1	1
#4		RESERVE BANK DAVID VAN									
#5		TAKATA BANKRUPTCY BRIDGE	Y	3		3	3			1	1
#6		KAREN AMLI OUTSTANDING									
#7		FOREIGN ERA MAHATHIR	Y	4		2	1		1	0.5	0.5
#8		COURT BORIS BECKER ABSA	Y	5,6		2	1	1		1	0.5
#9		EXPORT GROUP INDIAN	Y	7		2	2			1	1
#10		EASTERN CAPE CLEAN LOCAL GOVERNMENT									
TOPIC #5		Disaster accident emergecncy									
#1		TORNADO WARNING PM CDT COUNTY	Y	1		10	10			1	1
#2		TORNADO WARNING COUNTY AM CDT	Y	1		10	10			1	1
#3		TORNADO WARNING PM CDT COUNTY	Y	1		6	6			1	1
#4		TERRORIST ATTACK EXPLOSION TRAIN STATION	Y	2		6	6			1	1
#5		TORNADO WARNING COUNTIES PM CDT	Y	1		6	6			1	1
#6		STORM SCHOOL PHILIPPINES	Y	3		5	5			1	1
#7		CITY HALL BURNING AMLIVE	Y	4		2	2			1	1
#8		SITE CRASH SQUARE	Y	5		1	1			1	1
#9		TROPICAL STORM HEAVY RAIN AREA	Y	6		1	1			1	1
#10		INTERNATIONAL AIRPORT POLICE OFFICER MICHIGAN	Y	7		1	1			1	1

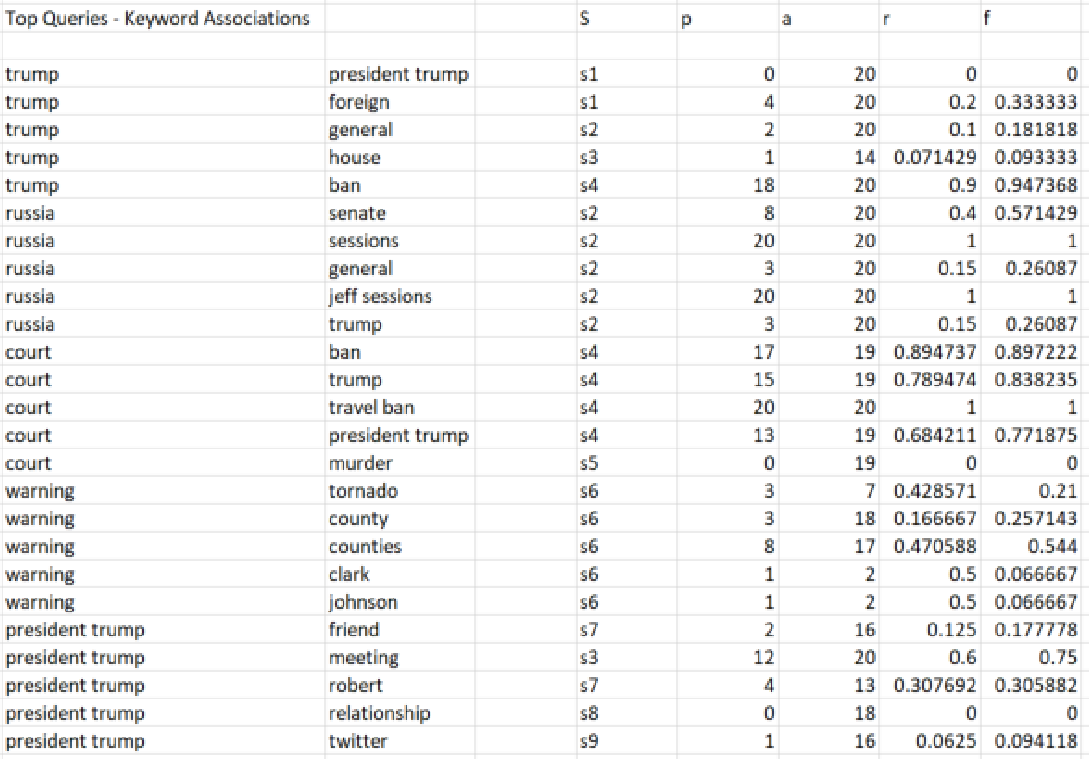
Figure 19: Story detection on 22 June 2017.



STORIES		Label	Story Y/N	Distinct		Nr Docs	This story	Other stor	No story	Homog.	Compl.
#1		HEALTH CARE GOP OBAMA	Y	1		10	10			1	1
#2		FIRE POLICE LONDON TOWER BLAZE	Y	2		10	5		5	0.5	0.5
#3		FORMER BOTSWANA KETUMILE MASIRE	Y	3		10	10			1	1
#4		MOGOENG COURT SECRET BALLOT	Y	4		10	10			1	1
#5		HEALTHCARE BILL DRAFT CARE BILL	Y	1		9	9			1	1
#6		AMLIVE KAREN TAXI	Y	5,6		7	4	2	1	0.857143	0.571429
#7		LAW FIRM UBER CASE	Y	7		5	5			1	1
#8		ZUMA GOVERNMENT COUNTRY									
#9											
#10											
USER STORIES											
#1		LOVE FANDOM TWEET									
#2		JAMES COMEY PRESIDENT TRUMP RETWE	Y	1		10	8		2	0.8	0.8
#3		COMEY TRUMP RETWEET	Y	1		10	7		3	0.7	0.7
#4		SHADOW BAN SO EVE									
#5		LAW COLD PRIVACY	Y	2		10	10			1	1
#6		BILL GOP HEALTH CARE	Y	3		10	10			1	1
#7		LAW COLD PRIVACY	Y	2		10	10			1	1
#8		GROUP FAVE CONFIRMED									
#9		GOP PAUL BILL	Y	3		10	10			1	1
#10											
TOPIC #1		Crime law justice									
#1		COURT ZUMA SECRET BALLOT	Y	1		7	7			1	1
#2		CONSITUTIONAL COURT SECRET BALLOT N	Y	1		6	6			1	1
#3		TOWER FIRE POLICE MANSLAUGHTER	Y	2		6	5		1	0.833333	0.833333
#4		DARREN OSBORNE MURDER	Y	3		5	5			1	1
#5		UBER CASE LAW FIRM	Y	4		5	5			1	1
#6		COURT LAWYER LIMPOPO	Y	5		3	3			1	1
#7		TAXI JOHANNESBURG METRO	Y	6		2	2			1	1
#8		CIRCUIT COURT TURNBULL TERRORISM	Y	7,8		2	1	1		1	0.5
#9		BILL COSBY ASSAULT PIC	Y	9		2	2			1	1
#10											
TOPIC #2		Politics									
#1		COURT VOTE MOGOENG	Y	1		7	7			1	1
#2		ELECTION OBAMA ADMINISTRATION ZUN	Y	1,2		5	5			1	1
#3		TRUMP BUDGET CABINET	Y	3		4	2		2	0.5	0.5
#4		AFGHANISTAN STATUS IRELAND	Y	4		3	3			1	1
#5		COUNCIL BLOCK FIRE CHELSEA	Y	5		2	2			1	1
#6		COUNCIL GUARDIAN FRONT CRITICISM	Y	5		2	2			1	1
#7		US AMBASSADOR JOHNSON TRUMP	Y	6		2	2			1	1
#8		CONSTITUTIONAL COURT TRUMP ZUMA	Y	1,7		2			2	0	0
#9		SPECIAL ELECTION GOP AGENDA	Y	8		2	2			1	1
#10											
TOPIC #3		Health									
#1		CARE BILL GOP HEALTH PLAN	Y	1		10	10			1	1
#2		HEALTH CARE DONALD TRUMP SENATOR	Y	1		8	8			1	1
#3		SENATE HEALTH JIMMY KIMMEL SENATOR	Y	1		7	7			1	1
#4		HEALTH CARE HEALTHCARE BILL PLAN	Y	1		7	7			1	1
#5		CARE BILL HEALTH PLAN BARACK OBAMA	Y	1		7	7			1	1
#6		HEALTHCARE BILL DRAFT SENATE	Y	1		7	7			1	1
#7		GOP HEALTH JIMMY KIMMEL SENATOR	Y	1		5	5			1	1
#8		CARE BILL OBAMA AMERICANS	Y	1		4	4			1	1
#9		HEALTHCARE BILL CARE BILL SENATE	Y	1		4	4			1	1
#10		HEALTH BILL HEALTH CARE OBAMA	Y	1		1	1			1	1
TOPIC #4		Conflicts war peace									
#1		AFGHANISTAN BOMB ATTACK BANK	Y	1		8	8			1	1
#2		ATTACK OBAMACARE CLASS	Y	2		7	2		5	0.285714	0.285714
#3		ATTACK MURDER DARREN	Y	3		3	3			1	1
#4		PAKISTAN CAR BOMB LEAST	Y	4		2	2			1	1
#5		ATTACK DEAD OKINAWA	Y	5		2	1		1	0.5	0.5
#6		CASTILE SHOOTING PHILANDO ISIS	Y	6,7		2	2			0.5	1
#7		GENERAL JEFF MICHIGAN POLICE OFFICER	Y	8		1	1			1	1
#8		TERRORIST ATTACK MICHIGAN AIRPORT S	Y	8		1	1			1	1
#9		DEADLY ATTACK RAMADAN TALIBAN	Y	1		1	1			1	1
#10		TERRORIST ATTACK ARM ILLINOIS	Y	3,9		1	1			0.5	1
TOPIC #5		Economy business finance									
#1		BANK AFGHANISTAN BOMB ATTACK	Y	1		8	8			1	1
#2		BANK CAR BOMB HORSE	Y	1		6	5		1	0.833333	0.833333
#3		BRAZIL BEEF FRESH	Y	2		5	5			1	1
#4		FRAUD CASE FINE LIONEL MESSI	Y	3		3	3			1	1
#5		BUDGET BANK LINE	Y								
#6		DEMOCRATS PRESIDENT OBAMA GERMAN	Y	4,5		2	2			0.5	1
#7		EXPORT GROUP INDIAN	Y	6		2	2			1	1
#8		BANK SOUTH AUSTRALIA CAR BOMB	Y	7,8		1	1			0.5	1
#9		CENTRAL BANK ATTACK GUN									
#10		FED STRESS MAJOR	Y	9		1	1			1	1

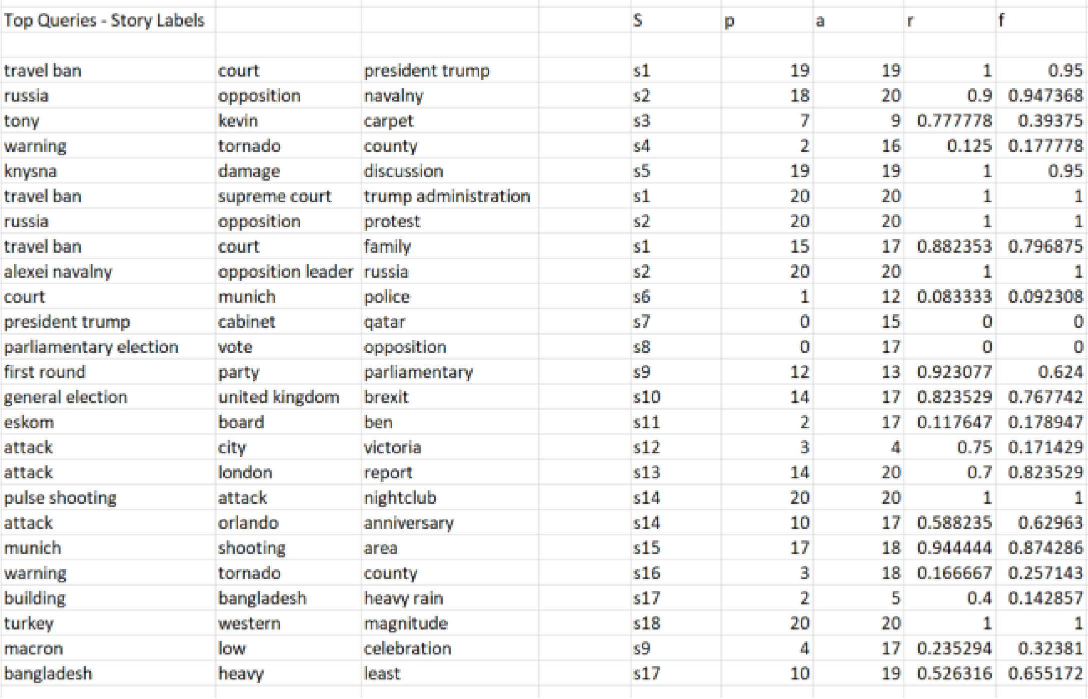
Figure 20: Story detection on 23 June 2017.

7 Appendix B: Social media extraction evaluation



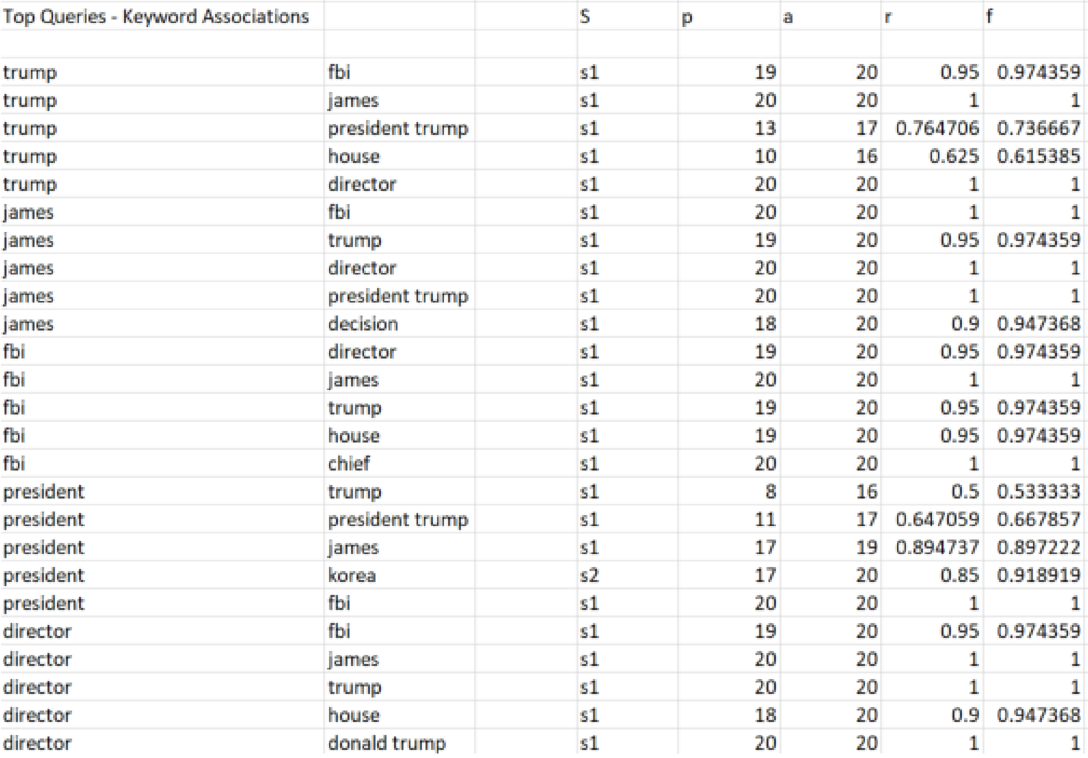
Top Queries - Keyword Associations			S	p	a	r	f
trump	president trump		s1	0	20	0	0
trump	foreign		s1	4	20	0.2	0.333333
trump	general		s2	2	20	0.1	0.181818
trump	house		s3	1	14	0.071429	0.093333
trump	ban		s4	18	20	0.9	0.947368
russia	senate		s2	8	20	0.4	0.571429
russia	sessions		s2	20	20	1	1
russia	general		s2	3	20	0.15	0.26087
russia	jeff sessions		s2	20	20	1	1
russia	trump		s2	3	20	0.15	0.26087
court	ban		s4	17	19	0.894737	0.897222
court	trump		s4	15	19	0.789474	0.838235
court	travel ban		s4	20	20	1	1
court	president trump		s4	13	19	0.684211	0.771875
court	murder		s5	0	19	0	0
warning	tornado		s6	3	7	0.428571	0.21
warning	county		s6	3	18	0.166667	0.257143
warning	counties		s6	8	17	0.470588	0.544
warning	clark		s6	1	2	0.5	0.066667
warning	johnson		s6	1	2	0.5	0.066667
president trump	friend		s7	2	16	0.125	0.177778
president trump	meeting		s3	12	20	0.6	0.75
president trump	robert		s7	4	13	0.307692	0.305882
president trump	relationship		s8	0	18	0	0
president trump	twitter		s9	1	16	0.0625	0.094118

Figure 21: Queries and their evaluation for the current approach on 13 June 2017.



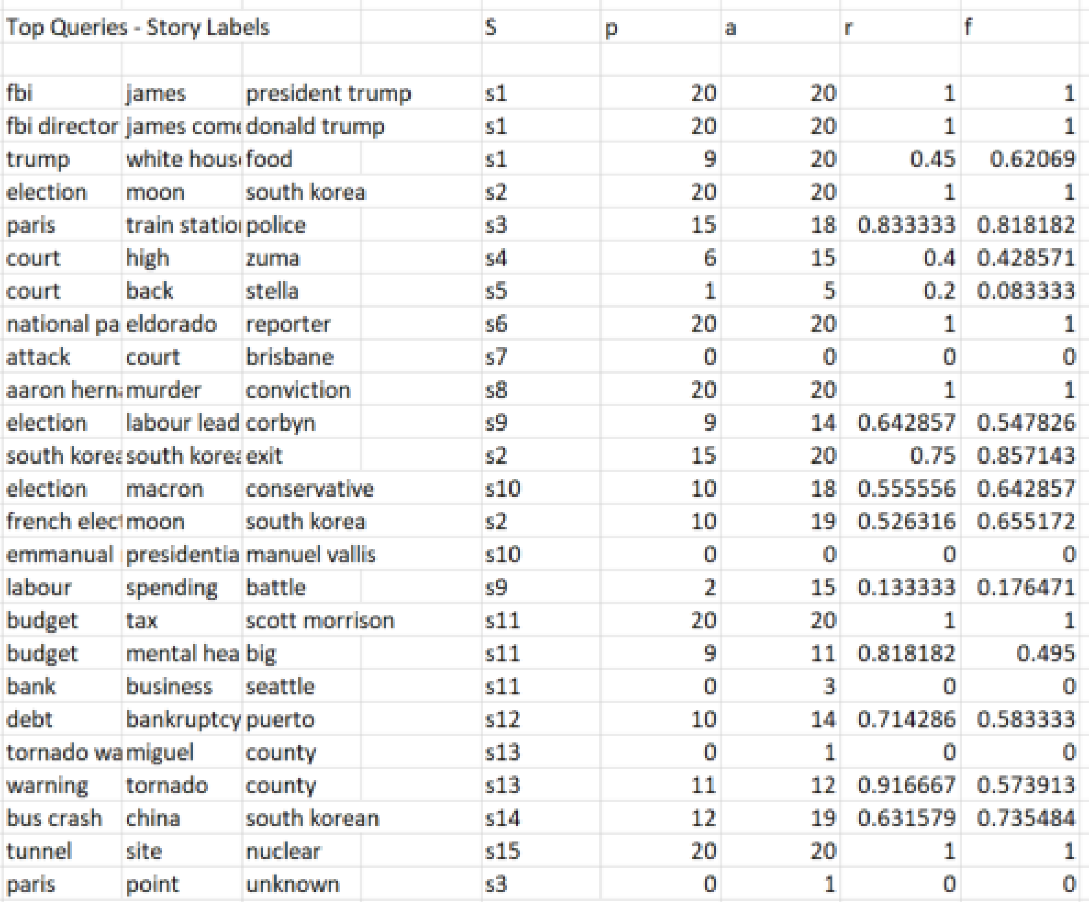
Top Queries - Story Labels				S	p	a	r	f
travel ban	court	president trump		s1	19	19	1	0.95
russia	opposition	navalny		s2	18	20	0.9	0.947368
tony	kevin	carpet		s3	7	9	0.777778	0.39375
warning	tornado	county		s4	2	16	0.125	0.177778
knysna	damage	discussion		s5	19	19	1	0.95
travel ban	supreme court	trump administration		s1	20	20	1	1
russia	opposition	protest		s2	20	20	1	1
travel ban	court	family		s1	15	17	0.882353	0.796875
alexei navalny	opposition leader	russia		s2	20	20	1	1
court	munich	police		s6	1	12	0.083333	0.092308
president trump	cabinet	qatar		s7	0	15	0	0
parliamentary election	vote	opposition		s8	0	17	0	0
first round	party	parliamentary		s9	12	13	0.923077	0.624
general election	united kingdom	brexit		s10	14	17	0.823529	0.767742
eskom	board	ben		s11	2	17	0.117647	0.178947
attack	city	victoria		s12	3	4	0.75	0.171429
attack	london	report		s13	14	20	0.7	0.823529
pulse shooting	attack	nightclub		s14	20	20	1	1
attack	orlando	anniversary		s14	10	17	0.588235	0.62963
munich	shooting	area		s15	17	18	0.944444	0.874286
warning	tornado	county		s16	3	18	0.166667	0.257143
building	bangladesh	heavy rain		s17	2	5	0.4	0.142857
turkey	western	magnitude		s18	20	20	1	1
macron	low	celebration		s9	4	17	0.235294	0.32381
bangladesh	heavy	least		s17	10	19	0.526316	0.655172

Figure 22: Queries and their evaluation for the proposed approach on 13 June 2017.



Top Queries - Keyword Associations		S	p	a	r	f
trump	fbi	s1	19	20	0.95	0.974359
trump	james	s1	20	20	1	1
trump	president trump	s1	13	17	0.764706	0.736667
trump	house	s1	10	16	0.625	0.615385
trump	director	s1	20	20	1	1
james	fbi	s1	20	20	1	1
james	trump	s1	19	20	0.95	0.974359
james	director	s1	20	20	1	1
james	president trump	s1	20	20	1	1
james	decision	s1	18	20	0.9	0.947368
fbi	director	s1	19	20	0.95	0.974359
fbi	james	s1	20	20	1	1
fbi	trump	s1	19	20	0.95	0.974359
fbi	house	s1	19	20	0.95	0.974359
fbi	chief	s1	20	20	1	1
president	trump	s1	8	16	0.5	0.533333
president	president trump	s1	11	17	0.647059	0.667857
president	james	s1	17	19	0.894737	0.897222
president	korea	s2	17	20	0.85	0.918919
president	fbi	s1	20	20	1	1
director	fbi	s1	19	20	0.95	0.974359
director	james	s1	20	20	1	1
director	trump	s1	20	20	1	1
director	house	s1	18	20	0.9	0.947368
director	donald trump	s1	20	20	1	1

Figure 23: Queries and their evaluation for the current approach on 10 May 2017.



Top Queries - Story Labels				S	p	a	r	f
fbi	james	president trump		s1	20	20	1	1
fbi director	james come	donald trump		s1	20	20	1	1
trump	white hous	food		s1	9	20	0.45	0.62069
election	moon	south korea		s2	20	20	1	1
paris	train statio	police		s3	15	18	0.833333	0.818182
court	high	zuma		s4	6	15	0.4	0.428571
court	back	stella		s5	1	5	0.2	0.083333
national pa	eldorado	reporter		s6	20	20	1	1
attack	court	brisbane		s7	0	0	0	0
aaron hern	murder	conviction		s8	20	20	1	1
election	labour lead	corbyn		s9	9	14	0.642857	0.547826
south kore	south kore	exit		s2	15	20	0.75	0.857143
election	macron	conservative		s10	10	18	0.555556	0.642857
french elec	moon	south korea		s2	10	19	0.526316	0.655172
emmanual	presidentia	manuel vallis		s10	0	0	0	0
labour	spending	battle		s9	2	15	0.133333	0.176471
budget	tax	scott morrison		s11	20	20	1	1
budget	mental hea	big		s11	9	11	0.818182	0.495
bank	business	seattle		s11	0	3	0	0
debt	bankruptcy	puerto		s12	10	14	0.714286	0.583333
tornado wa	miguel	county		s13	0	1	0	0
warning	tornado	county		s13	11	12	0.916667	0.573913
bus crash	china	south korean		s14	12	19	0.631579	0.735484
tunnel	site	nuclear		s15	20	20	1	1
paris	point	unknown		s3	0	1	0	0

Figure 24: Queries and their evaluation for the proposed approach on 10 May 2017.

References

Abdollahian, G., Taskiran, C. M., Pizlo, Z., & Delp, E. J. (2010, Jan). Camera motion-based analysis of user generated video. *IEEE Transactions on Multimedia*, *12*(1), 28-41. doi: 10.1109/TMM.2009.2036286

Aiello, L., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., ... Jaimes, A. (2013, October). Sensing Trending Topics in Twitter. *IEEE Transactions on Multimedia*, *15*(6), 1268–1282. doi: 10.1109/TMM.2013.2265080

Apostolidis, E., & Mezaris, V. (2014, May). Fast shot segmentation combining global and local visual descriptors. In *Proceedings of the 2014 ieee international conference on acoustics, speech and signal processing* (p. 6583-6587).

Argyriou, A., Evgeniou, T., & Pontil, M. (2007). Multi-task feature learning. *Advances in Neural Information Processing Systems (NIPS 2007)*.

Argyriou, A., Evgeniou, T., & Pontil, M. (2008). Convex multi-task feature learning. *Machine Learning*, *73*(3), 243-272.

Bai, L., Hu, Y., Lao, S., Smeaton, A. F., & O'Connor, N. E. (2010, August). Automatic summarization of rushes video using bipartite graphs. *Multimedia Tools Appl.*, *49*(1), 63–80. Retrieved from `http://dx.doi.org/10.1007/s11042-009-0398-1` doi: 10.1007/s11042-009-0398-1

Bay, H., Ess, A., Tuytelaars, T., & Gool, L. V. (2008, June). Speeded-up robust features (surf). *Computer Vision and Image Understanding*, *110*(3), 346–359. Retrieved from `http://dx.doi.org/10.1016/j.cviu.2007.09.014` doi: 10.1016/j.cviu.2007.09.014

Benois-Pineau, J., Lovell, B. C., & Andrews, R. J. (2013). Motion estimation in colour image sequences. In C. Fernandez-Maloigne (Ed.), *Advanced color image processing and analysis* (pp. 377–395). New York, NY: Springer New York. Retrieved from `http://dx.doi.org/10.1007/978-1-4419-6190-7_11` doi: 10.1007/978-1-4419-6190-7_11

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003, March). Latent Dirichlet Allocation. *J. Mach. Learn. Res.*, *3*, 993–1022. Retrieved 2016-01-14, from `http://dl.acm.org/citation.cfm?id=944919.944937`

Brigadir, I., Greene, D., & Cunningham, P. (2014). Adaptive representations for tracking breaking news on twitter. *CoRR*, *abs/1403.2923*. Retrieved from `http://arxiv.org/abs/1403.2923`

Burnside, G., Milioris, D., & Jacquet, P. (2014, April). One Day in Twitter: Topic Detection Via Joint Complexity.. Retrieved 2016-01-14, from `https://hal-polytechnique.archives-ouvertes.fr/hal-00967776`

Cameron, M. A., Power, R., Robinson, B., & Yin, J. (2012). Emergency situation awareness from twitter for crisis management. In *Proceedings of the 21st international conference on world wide web* (pp. 695–698).

Cataldi, M., Di Caro, L., & Schifanella, C. (2010). Emerging Topic Detection on Twitter Based on Temporal and Social Terms Evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining* (pp. 4:1–4:10). New York, NY, USA: ACM. Retrieved 2016-01-14, from `http://doi.acm.org/10.1145/1814245.1814249` doi: 10.1145/1814245.1814249

Chaabouni, S., Benois-Pineau, J., & Amar, C. B. (2016, Sept). Transfer learning with deep networks for saliency prediction in natural video. In *Ieee international conference on image processing (icip)* (p. 1604-1608).

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Trans. on Intelligent Systems and Technology*, *2*, 27:1–27:27.

Chatfield, K., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In *British machine vision conference.*

Chu, W.-T., Chuang, P.-C., & Yu, J.-Y. (2010). Video copy detection based on bag of trajectory and two-level approximate sequence. In *Matching, proceedings of ippr conference on computer vision, graphics, and image processing conference.*

Cooray, S. H., Bredin, H., Xu, L.-Q., & O'Connor, N. E. (2009). An interactive and multi-level framework for summarising user generated videos. In *Proceedings of the 17th acm international conference on multimedia* (pp. 685–688). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1631272.1631388` doi: 10.1145/1631272.1631388

Cooray, S. H., Lee, H., & O'Connor, N. E. (2011). A user-centric system for home movie summarisation. In *Proceedings of the 17th international conference on advances in multimedia modeling - volume part i* (pp. 424–434). Berlin, Heidelberg: Springer-Verlag. Retrieved from `http://dl.acm.org/citation.cfm?id=1949994.1950040`

Cooray, S. H., & O'Connor, N. E. (2010, Nov). Identifying an efficient and robust sub-shot segmentation method for home movie summarisation. In *2010 10th international conference on intelligent systems design and applications* (p. 1287-1292). doi: 10.1109/ISDA.2010.5687086

Cricri, F., Dabov, K., Curcio, I. D. D., Mate, S., & Gabbouj, M. (2011, Dec). Multimodal event detection in user generated videos. In *2011 ieee international symposium on multimedia* (p. 263-270). doi: 10.1109/ISM.2011.49

Daumé, H., III. (2009). Bayesian multitask learning with latent hierarchies. In *the 25th conf. on uncertainty in artificial intelligence (uai 2009)* (pp. 135–142). Quebec, Canada: AUAI Press.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by latent semantic analysis. *JAsIs*, *41*(6), 391–407.

Dhingra, B., Zhou, Z., Fitzpatrick, D., Muehl, M., & Cohen, W. W. (2016). Tweet2vec: Character-based distributed representations for social media. *CoRR*, *abs/1605.03481*. Retrieved from http://arxiv.org/abs/1605.03481

dos Santos, C. N., & Zadrozny, B. (2014). Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31th international conference on machine learning, ICML 2014, beijing, china, 21-26 june 2014* (pp. 1818–1826). Retrieved from http://jmlr.org/proceedings/papers/v32/santos14.html

Dumont, E., Merialdo, B., Essid, S., Bailer, W., Rehatschek, H., Byrne, D., ... Piatrik, T. (2008, 10). Rushes video summarization using a collaborative approach. In *TRECVID 2008, ACM International Conference on Multimedia Information Retrieval 2008, October 27-November 01, 2008, Vancouver, BC, Canada.* Vancouver, CANADA. Retrieved from http://www.eurecom.fr/publication/2576 doi: http://doi.acm.org/10.1145/1463563.1463579

Durik, M., & Benois-Pineau, J. (2001). Robust motion characterisation for video indexing based on mpeg2 optical flow. In *International workshop on content-based multimedia indexing, cbmi01* (p. 57-64).

Elbagoury, A., Ibrahim, R., Farahat, A., Kamel, M., & Karray, F. (2015, April). Exemplar-Based Topic Detection in Twitter Streams. In *Ninth International AAAI Conference on Web and Social Media.* Retrieved 2016-01-14, from http://www.aaai.org/ocs/index.php/ICWSM/ICWSM15/paper/view/10533

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (n.d.). *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.*

Evgeniou, T., & Pontil, M. (2004). Regularized multi–task learning. In *the 10th acm sigkdd int. conf. on knowledge discovery and data mining (kdd 2004)* (pp. 109–117). Seattle, WA.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., & Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874.

Fischler, M. A., & Bolles, R. C. (1981, June). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *ACM Communications*, *24*(6), 381–395. Retrieved from http://doi.acm.org/10.1145/358669.358692 doi: 10.1145/358669.358692

Fujiki, T., Nanno, T., Suzuki, Y., & Okumura, M. (2004). Identification of bursts in a document stream. In *First international workshop on knowledge discovery in data streams (in conjunction with ecml/pkdd 2004)* (pp. 55–64).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *the ieee conf. on computer vision and pattern recognition (cvpr 2014).*

González-Díaz, I., Martínez-Cortés, T., Gallardo-Antolín, A., & Díaz-de María, F. (2015, January). Temporal segmentation and keyframe selection methods for user-generated video search-based annotation. *Expert Syst. Appl.*, *42*(1), 488–502. Retrieved from http://dx.doi.org/10.1016/j.eswa.2014.08.001 doi: 10.1016/j.eswa.2014.08.001

Grana, C., & Cucchiara, R. (2006). Sub-shot summarization for MPEG-7 based fast browsing. In *Post-proceedings of the second italian research conference on digital library management systems (IRCDL 2006), padova, 27th january 2006* (pp. 80–84).

Guo, Y., Xu, Q., Sun, S., Luo, X., & Sbert, M. (2016). Selecting video key frames based on relative entropy and the extreme studentized deviate test. *Entropy*, *18*(3), 73. Retrieved from http://dblp.uni-trier.de/db/journals/entropy/entropy18.html#GuoXSLS16a

He, Q., Chang, K., & Lim, E.-P. (2007). Analyzing feature trajectories for event detection. In *Proceedings of the 30th annual international acm sigir conference on research and development in information retrieval* (pp. 207–214).

Hu, M., Liu, S., Wei, F., Wu, Y., Stasko, J., & Ma, K.-L. (2012). Breaking news on twitter. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 2751–2754). ACM.

Ifrim, G., Shi, B., & Brigadir, I. (2014). Event detection in twitter using aggressive filtering and hierarchical tweet clustering. In *Snow-dc@ www* (pp. 33–40).

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*.

Kang, H.-W., & Hua, X.-S. (2005). To learn representativeness of video frames. In *Proceedings of the 13th annual acm international conference on multimedia* (pp. 423–426). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1101149.1101242` doi: 10.1145/1101149.1101242

Karaman, S., Benois-Pineau, J., Dovgalecs, V., Mégret, R., Pinquier, J., André-Obrecht, R., ... Dartigues, J.-F. (2014). Hierarchical hidden markov model in detecting activities of daily living in wearable videos for studies of dementia. *Multimedia Tools and Applications*, *69*(3), 743–771. Retrieved from `http://dx.doi.org/10.1007/s11042-012-1117-x` doi: 10.1007/s11042-012-1117-x

Karaman, S., Benois-Pineau, J., Megret, R., Dovgalecs, V., Dartigues, J. F., & Gaestel, Y. (2010, Aug). Human daily activities indexing in videos from wearable cameras for monitoring of patients with dementia diseases. In *2010 20th international conference on pattern recognition* (p. 4113-4116). doi: 10.1109/ICPR.2010.999

Kasutani, E., & Yamada, A. (2001). The mpeg-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. In *Proceedings 2001 international conference on image processing (cat. no.01ch37205)* (Vol. 1, p. 674-677 vol.1). doi: 10.1109/ICIP.2001.959135

Katsios, G., Vakulenko, S., Krithara, A., & Paliouras, G. (2015). Towards open domain event extraction from twitter: Revealing entity relations. In *Proceedings of the 4th derive workshop co-located with the 12th extended semantic web conference (ESWC 2015), protoroz, slovenia, may 2015.* (pp. 35–46).

Kelm, P., Schmiedeke, S., & Sikora, T. (2009, May). Feature-based video key frame extraction for low quality video sequences. In *2009 10th workshop on image analysis for multimedia interactive services* (p. 25-28). doi: 10.1109/WIAMIS.2009.5031423

Kim, J.-G., Chang, H. S., Kim, J., & Kim, H.-M. (2000). Efficient camera motion characterization for mpeg video indexing. In *2000 ieee international conference on multimedia and expo. icme2000. proceedings. latest advances in the fast changing world of multimedia (cat. no.00th8532)* (Vol. 2, p. 1171-1174 vol.2). doi: 10.1109/ICME.2000.871569

Kim, Y., Jernite, Y., Sontag, D., & Rush, A. M. (2015). Character-aware neural language models. *CoRR*, *abs/1508.06615*. Retrieved from `http://arxiv.org/abs/1508.06615`

Koprinska, I., & Carrato, S. (1998). Video segmentation of mpeg compressed data. In *1998 ieee international conference on electronics, circuits and systems. surfing the waves of science and technology (cat. no.98ex196)* (Vol. 2, p. 243-246 vol.2). doi: 10.1109/ICECS.1998.814872

Krämer, P., & Benois-Pineau, J. (2005). Camera motion detection in the rough indexing paradigm. In *Trecvid 2005 workshop.* Gaithersburg, MD, USA.

Krestel, R., Werkmeister, T., Wiradarma, T. P., & Kasneci, G. (2015). Tweet-Recommender: Finding Relevant Tweets for News Articles. In *Proceedings of the 24th International Conference on World Wide Web Companion* (pp. 53–54). International World Wide Web Conferences Steering Committee.

Krizhevsky, A., Ilya, S., & Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NIPS 2012)*, 1097–1105.

Kumar, A., & Daume, H. (2012). Learning task grouping and overlap in multi-task learning. In *the 29th acm int. conf. on machine learning (icml 2012)* (pp. 1383–1390). Edinburgh, Scotland.

Lan, D.-J., Ma, Y.-F., & Zhang, H.-J. (2003, July). A novel motion-based representation for video mining. In *Multimedia and expo, 2003. icme '03. proceedings. 2003 international conference on* (Vol. 3, p. III-469-72 vol.3). doi: 10.1109/ICME.2003.1221350

Landauer, T. K., Foltz, P. W., & Laham, D. (1998, January). An introduction to latent semantic analysis. *Discourse Processes*, *25*(2-3), 259–284. Retrieved 2016-01-14, from `http://dx.doi.org/10.1080/01638539809545028` doi: 10.1080/01638539809545028

Leban, G., Fortuna, B., Brank, J., & Grobelnik, M. (2014). Cross-lingual detection of world events from news articles. In *Proceedings of the ISWC 2014 posters & demonstrations track a track within the 13th international semantic web conference, ISWC 2014, riva del garda, italy, october 21, 2014.* (pp. 21–24). Retrieved from `http://ceur-ws.org/Vol-1272/paper_19.pdf`

Leetaru, K., & Schrodt, P. A. (2013). Gdelt: Global data on events, location, and tone, 1979–2012. In *Isa annual convention* (Vol. 2, p. 4).

Lei, S., Xie, G., & Yan, G. (2014). A novel key-frame extraction approach for both video summary and video index. *The Scientific World Journal*, *2014*.

Lendvai, P., & Declerck, T. (2015, 10). Similarity-based cross-media retrieval for events. In R. Bergmann, S. Grg, & G. Mller (Eds.), *Proceedings of the lwa 2015 workshops: Kdml, fgwm, ir, and fgdb.* CEURS.

Liu, Y., Liu, Y., Ren, T., & Chan, K. (2008). Rushes video summarization using audio-visual information and sequence alignment. In *Proceedings of the 2nd acm trecvid video summarization workshop* (pp. 114–118). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1463563.1463584 doi: 10.1145/1463563.1463584

Long, M., & Wang, J. (2015). Learning multiple tasks with deep relationship networks. *CoRR*, *abs/1506.02117*.

Lu, Z., & Grauman, K. (2013). Story-driven summarization for egocentric video. In *Proceedings of the 2013 ieee conference on computer vision and pattern recognition* (pp. 2714–2721). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dx.doi.org/10.1109/CVPR.2013.350 doi: 10.1109/CVPR.2013.350

Luo, J., Papin, C., & Costello, K. (2009, February). Towards extracting semantically meaningful key frames from personal video clips: From humans to computers. *IEEE Trans. Cir. and Sys. for Video Technol.*, *19*(2), 289–301. Retrieved from http://dx.doi.org/10.1109/TCSVT.2008.2009241 doi: 10.1109/TCSVT.2008.2009241

Markatopoulou, F., Mezaris, V., & Patras, I. (2015). Cascade of classifiers based on binary, non-binary and deep convolutional network descriptors for video concept detection. In *the ieee int. conf. on image processing (icip 2015)* (p. 1786-1790). Quebec, Canada.

Markatopoulou, F., Mezaris, V., & Patras, I. (2016, Jan). Ordering of visual descriptors in a classifier cascade towards improved video concept detection. In *22nd int. conf. multimedia modeling (mmm 2016) part i* (Vol. 9516, p. 874-885). Miami, FL, USA: Springer International Publishing.

Markatopoulou et al., F. (2015). Iti-certh in trecvid 2015. In *Trecvid 2015.*

Martin, C., Corney, D., & Goker, A. (2015). Mining Newsworthy Topics from Social Media. In M. M. Gaber, M. Cocea, N. Wiratunga, & A. Goker (Eds.), *Advances in Social Media Analysis* (pp. 21–43). Springer International Publishing. Retrieved 2016-01-14, from http://link.springer.com/chapter/10.1007/978-3-319-18458-6_2 (DOI: 10.1007/978-3-319-18458-6_2)

Martin, C., & Göker, A. (n.d.). Real-time topic detection with bursty n-grams: Rgus submission to the 2014 snow challenge.

Mei, T., Tang, L.-X., Tang, J., & Hua, X.-S. (2013, July). Near-lossless semantic video summarization and its applications to video analysis. *ACM Trans. Multimedia Comput. Commun. Appl.*, *9*(3), 16:1–16:23. Retrieved from http://doi.acm.org/10.1145/2487268.2487269 doi: 10.1145/2487268.2487269

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*, *abs/1301.3781*. Retrieved from http://arxiv.org/abs/1301.3781

Mohanta, P. P., Saha, S. K., & Chanda, B. (2008, Dec). Detection of representative frames of a shot using multivariate wald-wolfowitz test. In *2008 19th international conference on pattern recognition* (p. 1-4). doi: 10.1109/ICPR.2008.4761403

Moran, S., McCreadie, R., Macdonald, C., & Ounis, I. (2016). Enhancing first story detection using word embeddings. In *Proceedings of the 39th international acm sigir conference on research and development in information retrieval* (pp. 821–824). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2911451.2914719 doi: 10.1145/2911451.2914719

Mousavi, H., Srinivas, U., Monga, V., Suo, Y., Dao, M., & Tran, T. (2014). Multi-task image classification via collaborative, hierarchical spike-and-slab priors. In *the ieee int. conf. on image processing (icip 2014)* (p. 4236-4240). Paris, France.

Ngo, C.-W., Ma, Y.-F., & Zhang, H.-J. (2005, Feb). Video summarization and scene detection by graph modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, *15*(2), 296-305. doi: 10.1109/TCSVT.2004.841694

Ngo, C.-W., Pong, T.-C., & Zhang, H.-J. (2003, March). Motion analysis and segmentation through spatio-temporal slices processing. *IEEE Transactions on Image Processing*, *12*(3), 341-355. doi: 10.1109/TIP.2003.809020

Nitta, N., & Babaguchi, N. (2013). [invited paper] content analysis for home videos. *ITE Transactions on Media Technology and Applications*, *1*(2), 91-100. doi: 10.3169/mta.1.91

Obozinski, G., & Taskar, B. (2006). Multi-task feature selection. In *the 23rd int. conf. on machine learning (icml 2006). workshop of structural knowledge transfer for machine learning.* Pittsburgh, Pennsylvania.

Odobez, J., & Bouthemy, P. (1995). Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, *6*(4), 348 - 365. Retrieved from `http://www.sciencedirect.com/science/article/pii/S1047320385710292` doi: http://dx.doi.org/10.1006/jvci.1995.1029

Ojutkangas, O., Peltola, J., & Järvinen, S. (2012). Location based abstraction of user generated mobile videos. In L. Atzori, J. Delgado, & D. Giusto (Eds.), *Mobile multimedia communications: 7th international icst conference, mobimedia 2011, cagliari, italy, september 5-7, 2011, revised selected papers* (pp. 295–306). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from `http://dx.doi.org/10.1007/978-3-642-30419-4_25` doi: 10.1007/978-3-642-30419-4_25

Omidyeganeh, M., Ghaemmaghami, S., & Shirmohammadi, S. (2011, Oct). Video keyframe analysis using a segment-based statistical metric in a visually sensitive parametric space. *IEEE Transactions on Image Processing*, *20*(10), 2730-2737. doi: 10.1109/TIP.2011.2143421

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In *the ieee conf. on computer vision and pattern recognition (cvpr 2014).* Columbus, Ohio.

Osborne, M., Petrovic, S., McCreadie, R., Macdonald, C., & Ounis, I. (2012). Bieber no more: First story detection using Twitter and Wikipedia. In *Proceedings of the Workshop on Time-aware Information Access. TAIA* (Vol. 12).

Ouyang, W., Chu, X., & Wang, X. (2014). Multi-source deep learning for human pose estimation. In *the ieee conf. on computer vision and pattern recognition (CVPR 2014)* (p. 2337-2344). Columbus, Ohio: IEEE.

Over et al., P. (2013). Trecvid 2013-an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Trecvid 2013.*

Pan, C.-M., Chuang, Y.-Y., & Hsu, W. H. (2007). Ntu trecvid-2007 fast rushes summarization system. In *Proceedings of the international workshop on trecvid video summarization* (pp. 74–78). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1290031.1290045` doi: 10.1145/1290031.1290045

Papadopoulos, S., Corney, D., & Aiello, L. M. (2014). Snow 2014 data challenge: Assessing the performance of news topic detection methods in social media. In *Snow-dc@ www* (pp. 1–8).

Petersohn, C. (2009, Nov). Temporal video structuring for preservation and annotation of video content. In *2009 16th ieee international conference on image processing (icip)* (p. 93-96). doi: 10.1109/ICIP.2009.5414114

Petkos, G., Papadopoulos, S., & Kompatsiaris, Y. (2014). Two-level message clustering for topic detection in twitter. In *Snow-dc@ www* (pp. 49–56).

Petrovi, S., Osborne, M., & Lavrenko, V. (2012). Using paraphrases for improving first story detection in news and Twitter. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 338–346). Association for Computational Linguistics.

Phuvipadawat, S., & Murata, T. (2010, August). Breaking News Detection and Tracking in Twitter. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 3, pp. 120–123). doi: 10.1109/WI-IAT.2010.205

Pittaras, N., Markatopoulou, F., Mezaris, V., & et al. (2017). Comparison of fine-tuning and extension strategies for deep convolutional neural networks. In *Multimedia modeling: 23rd international conference, mmm 2017, reykjavik, iceland, january 4-6* (pp. 102–114). Cham: Springer.

Popescu, A.-M., Pennacchiotti, M., & Paranjpe, D. (2011). Extracting events and event descriptions from twitter. In *Proceedings of the 20th international conference companion on World wide web* (pp. 105–106). ACM.

Pouliquen, B., Steinberger, R., & Deguernel, O. (2008). Story tracking: linking similar news over time and across languages. In *Proceedings of the workshop on multi-source multilingual information extraction and summarization* (pp. 49–56).

Ritter, A., Etzioni, O., Clark, S., & others. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1104–1112). ACM.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In *Computer vision (iccv), 2011 ieee international conference on* (pp. 2564–2571).

Rupnik, J., Muhic, A., Leban, G., Skraba, P., Fortuna, B., & Grobelnik, M. (2015). News Across Languages-Cross-Lingual Document Similarity and Event Tracking. *arXiv preprint arXiv:1512.07046*.

Russakovsky, O., Deng, J., & et al., H. S. (2015). ImageNet Large Scale Visual Recognition Challenge. *Int. Journal of Computer Vision (IJCV 2015)*, *115*(3), 211-252. doi: 10.1007/s11263-015-0816-y

Sidiropoulos, P., Mezaris, V., Kompatsiaris, I., Meinedo, H., Bugalho, M., & Trancoso, I. (2011, August). Temporal video segmentation to scenes using high-level audiovisual features. *IEEE Trans. Cir. and Sys. for Video Technol.*, *21*(8), 1163–1177. Retrieved from http://dx.doi.org/10.1109/TCSVT.2011.2138830 doi: 10.1109/TCSVT.2011.2138830

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv technical report*.

Snoek, C., Fontijne, D., van de Sande, K. E., & Stokman, H. e. a. (2015). Qualcomm research and university of amsterdam at trecvid 2015: Recognizing concepts, objects, and events in video. In *Trecvid 2015.*

Steiner, T., van Hooland, S., & Summers, E. (2013). MJ No More: Using Concurrent Wikipedia Edit Spikes with Social Network Plausibility Checks for Breaking News Detection. In *Proceedings of the 22nd International Conference on World Wide Web* (pp. 791–794). Geneva, Switzerland. Retrieved 2016-01-14, from http://dl.acm.org/citation.cfm?id=2487788.2488049

Stokes, N., & Carthy, J. (2001). Combining Semantic and Syntactic Document Classifiers to Improve First Story Detection. In *SIGIR 2001: Proceedings of the 24th ACM SIGIR Conference, September 9-13, 2001, New Orleans, Louisiana, USA* (pp. 424–425). Retrieved from http://doi.acm.org/10.1145/383952.384068 doi: 10.1145/383952.384068

Sun, G., Chen, Y., Liu, X., & Wu, E. (2015). Adaptive multi-task learning for fine-grained categorization. In *the ieee int. conf. on image processing (icip 2015)* (p. 996-1000). Quebec, Canada.

Szegedy et al., C. (2015). Going deeper with convolutions. In *the ieee conf. on computer vision and pattern recognition (cvpr 2015).* Boston, Massachusetts. Retrieved from http://arxiv.org/abs/1409.4842

Tang, L.-X., Mei, T., & Hua, X.-S. (2009). Near-lossless video summarization. In *Proceedings of the 17th acm international conference on multimedia* (pp. 351–360). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/1631272.1631321 doi: 10.1145/1631272.1631321

Treetasanatavorn, S., Heuer, J., Rauschenbach, U., Illgner, K., & Kaup, A. (2004, Oct). Temporal video segmentation using global motion estimation and discrete curve evolution. In *Image processing, 2004. icip '04. 2004 international conference on* (Vol. 1, p. 385-388 Vol. 1). doi: 10.1109/ICIP.2004.1418771

Vakulenko, S., Nixon, L. J. B., & Lupu, M. (2017). Character-based neural embeddings for tweet clustering. *CoRR*, *abs/1703.05123*. Retrieved from http://arxiv.org/abs/1703.05123

Van Canneyt, S., Feys, M., Schockaert, S., Demeester, T., Develder, C., & Dhoedt, B. (2014). Detecting newsworthy topics in Twitter. In *Data Challenge, Proceedings* (pp. 1–8). Seoul, Korea.

Wang, G., Seo, B., & Zimmermann, R. (2012). Motch: An automatic motion type characterization system for sensor-rich videos. In *Proceedings of the 20th acm international conference on multimedia* (pp. 1319–1320). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2393347.2396462 doi: 10.1145/2393347.2396462

Weng, J., & Lee, B.-S. (2011). Event detection in twitter. *ICWSM*, *11*, 401–408.

Wold, H. M., & Vikre, L. C. (2015). Online News Detection on Twitter.

Wu, Z., Chen, L., & Giles, C. L. (2015). Storybase: Towards Building a Knowledge Base for News Events. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, ACL 2015, July 26-31, 2015, Beijing, China, System Demonstrations* (pp. 133–138). Retrieved from http://aclweb.org/anthology/P/P15/P15-4023.pdf

Xu, J., Mukherjee, L., Li, Y., Warner, J., Rehg, J. M., & Singh, V. (2015). Gaze-enabled egocentric video summarization via constrained submodular maximization. In *Cvpr* (p. 2235-2244). IEEE Computer Society. Retrieved from http://dblp.uni-trier.de/db/conf/cvpr/cvpr2015.html#XuMLWRS15

Yang, Y., & Hospedales, T. M. (2015). A unified perspective on multi-domain and multi-task learning. In *the int. conf. on learning representations (iclr 2015).* San Diego, California.

Yilmaz, E., Kanoulas, E., & Aslam, J. A. (2008). A simple and efficient sampling method for estimating ap and ndcg. In *the 31st acm int. conf. on research and development in information retrieval (sigir 2008)* (pp. 603–610). Singapore.

Yin, J., Karimi, S., Lampert, A., Cameron, M., Robinson, B., & Power, R. (2015). Using social media to enhance emergency situation awareness. In *Twenty-fourth international joint conference on* *artificial intelligence.*

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems (NIPS 2014)*, 3320–3328.

Yue-Hei Ng, J., Yang, F., & Davis, L. S. (2015). Exploiting local features from deep networks for image retrieval. In *Proc. of the conf. on computer vision and pattern recognition workshops* (pp. 53–61).

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Proceedings of the 28th international conference on neural information processing systems* (pp. 649–657). Cambridge, MA, USA: MIT Press. Retrieved from `http://dl.acm.org/citation.cfm?id=2969239.2969312`

Zhang, Z., Luo, P., Loy, C. C., & Tang, X. (2014). Facial landmark detection by deep multi-task learning. In *the 13th europ. conf. on computer vision (eccv 2014)* (pp. 94–108). Zurich, Switzerland: Springer.

Zhou, J., Chen, J., & Ye, J. (2011a). Clustered multi-task learning via alternating structure optimization. *Advances in Neural Information Processing Systems (NIPS 2011)*.

Zhou, J., Chen, J., & Ye, J. (2011b). Malsar: Multi-task learning via structural regularization. *Technical report*.

Zimmermann, A. (2014). On the cutting edge of event detection from social streams–a non-exhaustive survey.