Tag-based Video Retrieval by Embedding Semantic Content in a Continuous Word Space

Arnav Agharwal Rama Kovvuri Ram Nevatia
University of Southern California
{agharwal,nkovvuri,nevatia}@usc.edu

Cees G.M. Snoek
University of Amsterdam
cgmsnoek@uva.nl

Abstract

Content-based event retrieval in unconstrained web videos, based on query tags, is a hard problem due to large intra-class variances, and limited vocabulary and accuracy of the video concept detectors, creating a "semantic query gap". We present a technique to overcome this gap by using continuous word space representations to explicitly compute query and detector concept similarity. This not only allows for fast query-video similarity computation with implicit query expansion, but leads to a compact video representation, which allows implementation of a real-time retrieval system that can fit several thousand videos in a few hundred megabytes of memory. We evaluate the effectiveness of our representation on the challenging NIST MEDTest 2014 dataset.

1. Introduction

Content-based video retrieval from a large database of unconstrained web videos is challenging due to large intra-class variations and difficulty of extracting semantic content from them. Exemplar-based methods have shown some success [24, 25] for video retrieval; here a set of positive videos along with a larger number of "background" videos is used to learn a distribution of features and concepts that can be used to measure similarity. We focus on concept tag-based retrieval as providing video exemplars is cumbersome, if not impossible, in many situations.

A key requirement for tag-based retrieval is for the multimedia system to extract semantic concepts, such as objects and actions, from videos. However, given the state-of-the-art, we can expect such extraction to be noisy. Additionally, though we can train detectors for tens of thousands of concepts, the vocabulary used for natural language retrieval may be significantly larger. Thus, we are faced with a "dictionary gap".

Early approaches for content-based retrieval find application in the image domain, where exemplar images were used as queries [20]. Recently, attempts were made to address the zero-shot image retrieval problem, where image queries were replaced with text queries [6, 22, 11], raising the issue of the "semantic query gap". These approaches propose automatic image captioning methods that, as a by-product, allow text queries to be used for image retrieval. This has been made possible due to datasets [27, 13], that have fine-grained sentence captions for images.

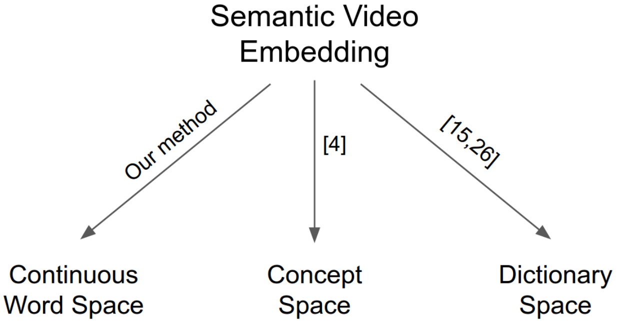

Figure 1. Methods classified based on semantic video embedding

In the absence of such datasets for videos, content-based retrieval methods rely on outputs from a semantic concept detector bank to close the "semantic query gap". While [14] uses exemplar videos as queries, the approaches [4, 26, 10, 2, 9, 8, 15] address the zero-shot retrieval problem. Based on the semantic video representation extracted from the detector bank (Figure 1), we consider the following three types of methods:

Concept Space (CoS) The response vector of the concept detector bank is used for semantic representation of a video. Thus, a video is represented as a point in a space that has dimensionality equal to the number of concept detectors.

Dictionary Space (DiS) Each concept in the detector bank is associated with one, or more words in a dictionary that are deemed semantically related to it, based on some similarity measure associated with the dictionary. The score of each detector is distributed over the "similar" words, possibly using the similarity scores. Thus, the detector bank response vector is mapped to a vector of scores in a space where each dimension corresponds to a word in the dictionary. Such an embedding is typically sparse, since the size of a dictionary is many times larger than the number of concepts in the detector bank.

Continuous Word Space (CWS) Each concept in the detector bank may be associated with a point in some continuous word space. Such a point may not necessarily correspond to a word in a dictionary, but may be a "blend" of continuous word vectors corresponding to different words. For each detector concept, the response score, combined with its continuous word space vector, is used to represent the video as a point in the continuous word space.

We propose a retrieval technique, falling under CWS embedding scheme, to implicitly map query concepts with the closest concepts in the detector vocabulary (*e.g.* "cake" may be mapped to "food" in absence of a cake detector), and aggregate their scores to help fill the "semantic query gap". Our method allows for retrieval by a single tag or a set of tags; more tags are likely to be descriptive and discriminative. It yields a compact video representation in a continuous semantic space, which can fit several hundred thousand videos in a few gigabytes of memory. We implemented a real-time, interactive retrieval system that can run on a thin client.

We also extend the method of [4], which is originally based on a CoS embedding scheme, for application to query tag-based retrieval for videos in the wild. To represent DiS methods, we implement the approach in [26], modifying low level details for fair comparison with the proposed CWS and CoS schemes. We evaluate the three approaches on the challenging NIST MEDTest 2014 [18] dataset.

In our CWS approach, since query tags map to the video embedding space, query-video similarity can be computed as a dot product in a low-dimensional space. The mapping offers multiple advantages. Embedding videos in a continuous word space:

1. results in implicit query expansion (mapping query tags to multiple, semantically related tags)

2. leads to compact video representation due to low dimensionality of the space, which is a significant gain, as the detector bank may contain a large number of concepts

3. allows fast scoring for each video

2. Related Work

In this section, we present video retrieval approaches that are relevant to our method.

Exemplar-based video retrieval

A video retrieval approach using few video exemplars as queries has been presented in [14]. Each video (in a query and the database) is represented as a vector of concept detector bank responses (CoS embedding). It evaluates multiple similarity measures to compute similarity between query and database videos, as well as early and late fusion schemes to obtain ranked lists in case of multiple video queries. Presenting K-shot retrieval experiments, the work concludes that for $K \leq 9$, the retrieval pipeline (using cross-correlation similarity metric and late fusion aggregation) outperforms supervised classifiers, trained using the K query videos as positive exemplars.

Query tag-based video retrieval

Semantic tag-based retrieval methods have been proposed in [4, 26, 10, 2, 9, 8, 15].

The approaches in [26, 15] can be classified as DiS methods. In [26], both videos and queries are embedded as sparse vectors in a high-dimensional vocabulary space, where each dimension corresponds to a concept in a text corpus. The detector scores are propagated to the top closest vocabulary matches for each concept in the detector bank, using a corpus similarity measure.

The method of [15] requires a set of "source" videos, tagged with concepts supplied by human annotators. The tags are propagated to a test video using similarity scores computed between "source"-test video pairs, based on a mid-level video representation. Once a test video is represented as a set of tags, the idea is to use standard document retrieval techniques for text queries. While their method shows promising results, it suffers from the drawback that it relies on a set of human-annotated videos. The "source" set used for evaluation contains a set of carefully annotated videos, with tags semantically related to the domain of application. Thus, it is not clear how the method will perform with noisy, and possibly unrelated, web-scale annotations.

The work in [4] belongs to the class of CoS embeddings. While it does not directly address the problem of query tag-based retrieval, it trained action classifiers on a subset of classes from the "UCF-101" [23] dataset, and used the names of held-out classes as queries for unsupervised retrieval. The response of the trained detectors is used as the mid-level concept bank video representations, and a continuous word space is used to select and aggregate scores of the top K detectors semantically closest to the query. The approach exhibits favorable performance over supervised classifiers trained using up to 5 positive exemplars. Since the concepts in the detector bank are few, and semantically correlated with the set of held out classes (sports actions), it is not obvious if the method will successfully apply to our domain, where the detector bank consists of thousands of potentially unrelated concepts with noisy responses.

The papers [9, 8] have proposed re-ranking approaches to improve retrieval results. The work [9] combines multiple ranked lists generated using different modalities from retrieval, exploiting both semantic, as well as low-level features. Authors in [8] propose a re-ranking scheme for multimodal data inspired from curriculum learning methods proposed in [1, 12]. Since re-ranking approaches post-process initial ranked lists, they may be considered as “add-ons” to the methods evaluated in this paper.

The image retrieval method of [21] comes closest to our proposed CWS approach. They learn a mapping from images to a semantic word space using a neural network, and use the embedding to distinguish between “seen” and “unseen” classes using novelty detection. An image labeled to belong to a seen class is classified using detectors trained on image-based features that were trained using exemplars. The unseen class images are classified by ranking according to distance from the continuous word space embedding of the class name.

In contrast, our CWS embedding uses a natural mapping function to embed concept detector bank responses, by combining them with their corresponding continuous word space embeddings.

3. Technical Approach

The approaches in each of the three dimensions studied in this paper begin with extraction of mid-level semantic features over videos. We detail below our “continuous word space” embedding method, and an extension of the “concept space” embedding approach of [4]. We also include an overview of the “dictionary space” embedding method of [26].

3.1. Continuous Word Space (CWS) Embedding

We begin by mapping feature bank concepts to their respective continuous word space vectors. Each video is then mapped to the continuous word space by computing a sum of these concept bank vectors, weighted by the corresponding detector responses.

User query tags are mapped to the continuous word space, and aggregated to form a single query vector. Finally, videos are scored using a similarity measure in the common space. The framework is illustrated in Figure 2.

3.1.1 Mid-level semantic video representation

We represent a video V_i as a vector of concept bank confidence scores, $f_i = (w_{i1}, w_{i2}, \ldots, w_{iN})^T$, corresponding to semantic concepts $\{c_1, c_2, \ldots, c_N\}$. Owing to semantic interpretation, such a representation is more useful for event-level inference than standard bag-of-words.

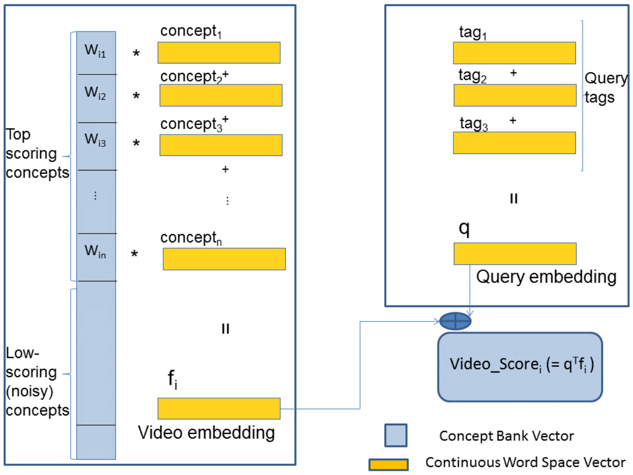

Figure 2. Video and Query Embedding framework

The video feature is assumed to be an L_1-normalized histogram f_i of these concept scores, which are scaled confidence outputs of detectors trained on the corresponding concepts.

3.1.2 Concepts and Queries to Continuous Word Space

We map each semantic feature concept c_i to its corresponding continous word representation vector $v(c_i)$. Since it is possible that no (case-insensitive) exact match exists in the vocabulary, matches may be computed by heuristic matching techniques (Section 4.4).

A user query can contain one, or more tags. Each tag is mapped to its corresponding continuous word space representation, if it exists. The query vector q is an L_2-normalized sum of these vectors, which is equivalent to the Boolean AND operator over input tags. More sophisticated schema (such as in [7]) may be desgined for query representation and combination, but this is not explored in this paper.

3.1.3 Embedding Function

We interpret a video as a text document, with the feature response vector being a histogram of word occurence frequency. The video feature vector is mapped to continuous word space by computing sum of concept vectors (mapped in Section 3.1.2), weighted by their corresponding feature responses. To avoid including noisy responses in the representation, only a thresholded set of top responses is used. If we denote f_i^c to be the embedding for video v_i, then

$$f_i^c = \sum_{c_k \in C_i} w_{ik}^{'} * v(c_k)$$

where C_i is the set of top responses for that video. The weights $w_{ij}^{'}$ are equivalent to w_{ij} up to scale, to ensure $\|f_i^c\|_2 = 1$.

Since the number of concepts in a concept bank can be large, and if detectors are trained with reasonable accuracy, it is expected that the bulk of the feature histogram will be distributed among a few concepts. Aggregating high-confidence detector responses helps in suppressing the remaining noisy detector responses.

3.1.4 Scoring videos

Videos are scored using the dot product similarity measure. This measure has the advantage of implicitly performing query expansion, i.e., responses of feature concepts semantically related to the query tags will be automatically aggregated, *e.g.*, if a user queries for "vehicle", then responses in video V_i, for related tags (assuming they are among the thresholded concepts), such as "car", or "bus", will be aggregated as follows:

$$\begin{aligned} q^T f_i^c &= v(\text{vehicle})^T \sum_{c_k \in C_i} w_{ik}^{'} * v(c_k) \\ &= v(\text{vehicle})^T [w_{i\text{car}}^{'} * v(\text{car}) + w_{i\text{bus}}^{'} * v(\text{bus})] \\ &\quad + v(\text{vehicle})^T \sum_{c_i \in {C_i \setminus \atop \{\text{car,bus}\}}} w_i^{'} * v(q)^T v(c_i) \\ &\approx v(\text{vehicle})^T [w_{i\text{car}}^{'} * v(\text{car}) + w_{i\text{bus}}^{'} * v(\text{bus})] + 0 \end{aligned}$$

since unrelated concepts are expected to be nearly orthogonal.

3.2. Concept Space (CoS) Embedding

The authors in [4] applied their retrieval method to the "UCF-101" [23] dataset. The goal of their method was to transfer knowledge from detectors trained on a known set of classes, to create a detector for an unseen class. This was achieved by aggregating detector scores of the known classes, weighted by their semantic similarity to the unseen class, using a continuous word space. We briefly describe the key components of their approach, and our modifications that extend its application to the domain of query tag-based retrieval for unconstrained web videos:

Mid-level representation In [4] classifiers were trained for a held out set of classes. These classifiers were then applied to the target class, and their confidence scores were concatenated to obtain a video representation. We replace these classifiers by our detector bank, as described in Section 3.1.1. As in their work, we use this as the final video representation, yielding a concept space video embedding.

Unseen class representation In [4], a class name is represented by its continuous word representation, trained on Wikipedia articles. Each detector concept is similarly mapped to this space. The representations are used to rank concepts with respect to the query by computing the dot product similarity score. We extend this idea, and replace the unseen class with the set of query tags.

Scoring videos The query-video similarity score is computed as a sum of the top K detector responses, weighted by the class-concept similarity score obtained in the previous step. We leave this unmodified in our extension.

3.3. Dictionary Space (DiS) Embedding

For completeness, we provide a high-level overview of the method presented in [26], which is evaluated in this work. The authors first trained multiple concept detector banks and applied them to each video, to obtain a set of mid-level concept space representations. In a given detector bank, for each concept, they obtain the top K similar words in a corpus, using a similarity score derived from the corpus. The confidence score of a concept detector for each video is distributed to the top K similar words, weighted by the corresponding similarity score. Thus, each detector bank response is mapped to a sparse vector in the dictionary space.

The tags in a query (which are a subset of the word corpus) are used to create a sparse vector in the dictionary space. The dimensions corresponding to each query tag are set to 1.0, while the rest are set to 0.0. For each detector bank, a similarity score with the query vector is computed as a dot product in this space. These similarity scores are fused to obtain the final query-video similarity score.

4. Experiments

In this section, we present the setup for tag-based retrieval experiments on the challenging NIST MEDTEST 2014 dataset [18]. We evaluate the performance of our continuous word space embedding approach, our concept space embedding extension of [4], and our implementation of the dictionary space method of [26].

4.1. Dataset

The MED dataset [18] provided by NIST consists of unconstrained Web videos downloaded from the Internet. The 2014 release consists of 20 complex high-level events, such as "Bike trick" and "Wedding shower" (a complete list is given in Table 1).

The dataset consists of two splits: 1) EventKit (or EK) includes some training videos (not relevant to our method), along with a description of each event, consisting of the sequence of actions, activities, scenes, and objects typically associated with it; and 2) MEDTest 2014 contains 27,269 test videos. This split has around 20 positive video samples, and 5,000 background (negative) videos, per event.

4.2. Concept Bank

In our concept bank, each concept is a node (internal or leaf) in the ImageNet hierarchy [3], which itself is a subgraph of the WordNet [17] hierarchy. It has $N \approx 15,000$ noun phrases, that consist of scenes, objects, people, and activities. Training images for each detector were obtained from the corresponding nodes in the ImageNet hierarchy. The detector bank is a deep convolutional neural network with multi-class softmax outputs.

The histogram f_i for video V_i is computed by average pooling of L_1-normalized frame-level responses computed on frames sampled at equally spaced intervals, and then renormalizing. Average pooling is preferred as it exhibits stable performance for noisy concept detectors [14].

For fair comparison, we use this concept bank for both the "concept space" ([4] uses "UCF-101" class names), and "dictionary space" ([26] uses detectors trained in-house) approaches.

4.3. Continuous Word Space

We use the word2vec [16] framework for obtaining continuous word space mapping of concepts. The word representation is obtained as a by-product of training a skip-gram neural network language model. The word2vec mapping used in our pipeline was trained on the Google News dataset. This dataset yields a 300,000 word vocabulary, and each word is mapped to a 300-dimensional vector. We used this corpus due its large vocabulary size.

The authors show in [16] that summing up vectors corresponding to each word in a sentence leads to a discriminative sentence representation, assuming that it contains a small number of words. By using only the top detector responses (Section 3.1.3), our embedding function is consistent with the above assumption, and is likely to produce a discriminative video image. We empirically threshold, and aggregate top 30% detector responses in the histogram, as it yields good retrieval results with our concept bank.

In [4], the authors use the word2vec representation trained on Wikipedia articles, while in [26], the Gigaword corpus [5] is used. For comparable results, we use the Google News corpus trained word2vec representation for evaluating both approaches.

4.4. Vector Mapping for Concept Bank

Since concepts may not have precise matches in the continuous word space vocabulary, we find a match in the following order:

1. Use the exact match, if available.
2. If it is a compound word with no exact matches, or a multi-word phrase, we search for exact matches for the constituent atomic words. If one or more atomic words match, then the original concept is mapped to an L_2-normalized sum of the corresponding vectors.
3. If no match found in previous step, the concept is discarded, and mapped to a zero vector.

4.5. Evaluation

For each event, we pick nouns from the supplied description in the EventKit, and use them as query tags to evaluate retrieval performance. Since retrieval performance depends on availability and quality of relevant detectors, rankings will vary with different query tags for the same event. Thus, we report the best performing tag (highest mean average precision or mAP) using our pipeline.

Using the tags selected above, for each event, we compare our method's performance with our extension of the CoS technique presented in [4]. We use our heuristics (Section 4.4) for mapping detector concepts to continuous word space vectors. As per the authors' recommendation, we test performance using $K = 3, 5$, and present results for the best performing parameter value, $K = 3$. Here, K indicates the number of top related concept detectors used for scoring.

For the DiS baseline of [26], since the value of K (denotes the number of nearest words in the dictionary for each detector concept) used in their experiments is not mentioned in the paper, we repeat the experiment with values of $K = 3, 5, 7$. We present results for $K = 5$, which results in the highest AP. In our case, there is only one detector bank (which corresponds to the image modality), so the multimodal fusion techniques presented in [26] are not applicable.

We also test several tag combinations for a query, and present the best perfoming combinations for the CWS embedding, per event. Tags are combined using AND boolean semantics: query vectors for individual tags are averaged, and then L_2-normalized, to obtain the final query vector. This is equivalent to late fusion by averaging video scores using individual tags. For comparison, the same tags are used as query inputs to the CoS and DiS methods.

5. Results

For each event in the MED2014 dataset, the Average Precision (AP) score using selected single tag queries are presented in Table 1, and using multiple tag queries in Table 2. While the mean Average Precision (mAP) for a single tag query using our CWS method is comparable with CoS and DiS methods, our multiple tag results demonstrate significant gains.

ID	Event Name	Tags	DiS	CoS	CWS
21	Bike trick	bicycle	0.0417	0.0253	**0.0475**
22	Cleaning an appliance	cooler	0.0479	0.0492	**0.0521**
23	Dog show	rink	0.0082	0.0066	**0.0103**
24	Giving direction	cop	0.0500	**0.0551**	0.0477
25	Marriage proposal	marriage	0.0015	0.0029	**0.0039**
26	Renovating a home	furniture	0.0086	0.0110	**0.0236**
27	Rock climbing	climber	**0.1038**	0.0649	0.0823
28	Town Hall meeting	speaker	0.0842	0.1025	**0.1145**
29	Winning a race without a vehicle	track	0.1217	**0.1374**	0.1233
30	Working on a metal crafts project	repair	0.0008	0.0079	**0.0564**
31	Beekeeping	apiary	0.5525	0.5697	**0.5801**
32	Wedding shower	wedding	0.0120	0.0120	**0.0248**
33	Non-motorized vehicle repair	bicycle	0.0247	**0.1559**	0.0191
34	Fixing musical instrument	instrument	0.0131	0.0179	**0.1393**
35	Horse-riding competition	showjumping	0.2711	0.2832	**0.2940**
36	Felling a tree	forest	**0.1593**	0.1468	0.1303
37	Parking a vehicle	vehicle	0.0813	**0.0882**	0.0768
38	Playing fetch	dog	0.0073	**0.0079**	0.0054
39	Tailgating	jersey	0.0022	0.0028	**0.0031**
40	Tuning a musical instrument	piano	0.0687	0.0363	**0.0795**
mAP			0.0830	0.0892	**0.0957**

Table 1. Retrieval performance (AP metric) using single query tag on the NIST MEDTEST 2014 dataset using our "continuous word space" (*CWS*), our "concept space" (*CoS*) extension of [4], and "dictionary space" (*DiS*) [26] embedding approaches.

ID	Event Name	Tags	DiS	CoS	CWS
21	Bike trick	bicycle chain	0.0307	0.0208	**0.0747**
22	Cleaning an appliance	cooler refrigerator	0.0512	0.0493	**0.0551**
23	Dog show	exhibition hall dog competition	0.0060	0.0142	**0.0489**
24	Giving direction	cop map	0.0068	**0.0615**	0.0351
25	Marriage proposal	woman ring	0.0031	**0.0031**	0.0029
26	Renovating a home	furniture ladder	0.0092	0.0110	**0.0884**
27	Rock climbing	mountaineer climber	0.1038	0.0674	**0.1192**
28	Town Hall meeting	speaker town_hall	0.0842	0.0991	**0.1381**
29	Winning a race without a vehicle	swimming track	0.1241	0.1887	**0.2185**
30	Working on a metal crafts project	metal solder	0.0024	0.0028	**0.0264**
31	Beekeeping	honeybee apiary	0.0929	0.5530	**0.5635**
32	Wedding shower	gifts wedding	0.0120	0.0110	**0.0191**
33	Non-motorized vehicle repair	bicycle tools	0.0090	**0.1538**	0.0414
34	Fixing musical instrument	instrument tuning	0.0249	0.0179	**0.2692**
35	Horse-riding competition	showjumping horse	0.2765	0.2863	**0.2985**
36	Felling a tree	tree chainsaw	0.0409	0.0440	**0.0521**
37	Parking a vehicle	parking vehicle	0.0621	**0.0882**	0.0504
38	Playing fetch	dog beach	0.0077	0.0079	**0.1006**
39	Tailgating	jersey car	0.0034	0.0028	**0.0036**
40	Tuning a musical instrument	piano repair	0.1004	0.0320	**0.1458**
mAP			0.0526	0.0857	**0.1176**

Table 2. Retrieval performance (AP metric) for multiple query tags on the NIST MEDTEST 2014 dataset using our "continuous word space" (*CWS*), our "concept space" (*CoS*) extension of [4], and "dictionary space" (*DiS*) [26] embedding approaches. The entries where performance is lower than their corresponding single tag scores, are underlined.

For the case of multiple tag queries (Table 2), most events show improved performance over the corresponding single tag result, using the CWS embedding. In particular, large bumps in performace are observed for the events 26, 29, 34, 39, and 40. This can be explained by implicit query expansion aggregating related high quality concept detectors, whose responses combine favorably during late fusion.

In contrast, both CoS and DiS methods exhibit a performance drop for several events, using multiple query tags. This can be explained by sensitivity to the fixed parameter K. In the case of CoS, the value of K that was aggregating sufficient detector responses for a single tag, may not be enough for multiple tags. The DiS approach will select K nearest dictionary words for every query tag, and may end up selecting concepts that have aggregated scores of noisy detectors. While it is not evident how to select K in case of multiple tags, our CWS embedding avoids this issue by aggregating all relevant detector responses (equivalent to adaptively setting K equal to the number of concepts), as shown in Section 3.1.4.

For visualization, we selected a few tags from the *Single tag* column in Table 1 and show screenshots from the top 5 videos retrieved by our system in Figure 3. It may be noted that the tags “forest” and “vehicle” do not have matching detectors in the concept bank. Hence, the rankings are based on implicit query expansion resulting from the continuous word space video representation.

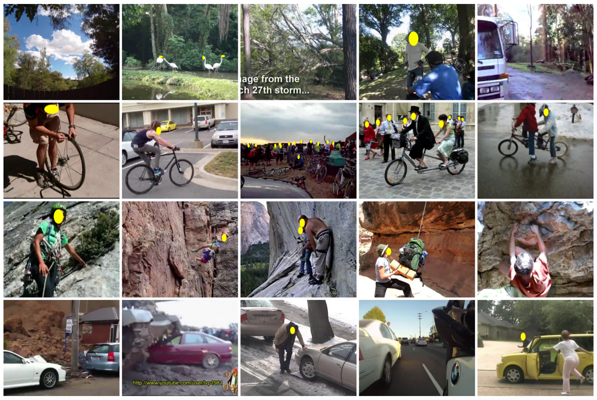

Figure 3. Top 5 video screenshots for single query tags in row 1) “forest”, 2) “bicycle”, 3) “climber”, 4) “vehicle”. Ranks decrease from left to right.

Using our CWS representation, assuming 8-byte double precision floating point numbers, we can fit 200,000 videos (the entire TRECVid [19] database) in 460 MB of memory, versus concept bank representation, which requires over 22 GB.

While the CWS embedding is inherently slightly more computationally intensive, relative to CoS and DiS schemes, it tends to capture diverse detection responses. This is critical in case one, or more, query tags do not have any corresponding concepts in the detector bank. In such a case, strong responses belonging to semantically related concept detectors may not be captured among the top K responses. This becomes apparent in AP scores for events 30 and 34. The query tags, “repair” and “instrument” do not have corresponding concepts in the detector bank, but our CWS embedding captures relevant responses, vastly improving results.

For the case of single tag queries, AP scores are comparable for a given event across columns, suggesting that performance may be limited by the availability and reliability of semantically relevant detectors in the concept bank. Since detector aggregation depends on correlations captured by the continuous word space, the system performance may improve by training word2vec on a special-purpose corpus, that captures scene level correlations.

6. Efficiency

For real-time web-scale applications, careful implementation can significantly reduce query time, as well as disk usage, for the methods evaluated in this paper, without affecting retrieval performance.

A naive implementation would be to store features for each video separately on disk. For CoS and DiS methods, this would entail iterating through many small files to compute similarity score. Instead, a reverse index implementation, would considerably speed-up the computation. Further speed-ups can be obtained by sparsifying detection vector responses using the method of [10]. However, that requires the creation of a HEX graph for the concepts in the detector bank, whose size grows quadratically with the number of detectors.

Even though the proposed CWS embedding neccessitates computation of full query-video dot product, speed-ups can be obtained by using an indexing structure, such as a BSP tree, or KD tree. Since video representation is compact, all videos at the leaf of such a structure can be loaded into memory in one shot, reducing disk I/O time. Since retrieval systems are typically concerned with only the top few videos, we can safely discard a large fraction of the database, and only rank videos that are deemed to be close enough by the indexing data structure. The CWS scheme is also expected to benefit from sparsification using [10].

7. Conclusion and Future Work

We presented a novel “continuous word space” (CWS) video embedding framework for retrieval of unconstrained web videos using tag-based semantic queries. In addition, we extended the method in [4], to create a “concept space” (CoS) video embedding pipeline, and implemented the “dictionary space” (DiS) video embedding retrieval method of [26]. We evaluated the retrieval performance of these three methods on the challenging NIST MEDTest 2014 [18] dataset.

The evaluation results show that our CWS framework outperforms the CoS, DiS based methods. Even though our naive implementation of the CWS method is expensive in comparison to reverse indexing implementations of the CoS and DiS methods, it yields a significantly compact representation in comparison to a mid-level concept detector bank vector representation. This allows implementation of a real-time interactive system on a thin client by loading a database of hundreds of thousands of videos into main memory.

Using the method of [10] for noisy detector response removal, instead of the top 30% score heuristic, is expected to improve the retrieval performance of our CWS embedding, while maintaining a small memory footprint, and low latency.

We used the word2vec [16] continuous word space representation pre-trained on the Google News corpus, which may not capture scene- or event-leval correlations very well, *e.g.*, in news articles, the phrase "road" may not be very highly correlated with a "stop sign". This correlation becomes important for scene identification. Retrieval performance of each method should improve by training on a special-purpose corpus. Lastly, the base retrieval results can be further improved using re-ranking methods, such as [8].

Acknowledgements Supported, in part, by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

[1] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *ICML*. ACM, 2009.

[2] J. Dalton, J. Allan, and P. Mirajkar. Zero-shot video retrieval using content and concepts. In *CIKM*. ACM, 2013.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[4] C. Gan, M. Lin, Y. Yang, Y. Zhuang, and A. G. Hauptmann. Exploring semantic inter-class relationships (sir) for zero-shot action recognition. In *ACAI*, 2015.

[5] D. Graff, J. Kong, K. Chen, and K. Maeda. English gigaword third edition ldc2007t07. In *LDC*, 2009.

[6] S. Guadarrama, E. Rodner, K. Saenko, N. Zhang, R. Farrell, J. Donahue, and T. Darrell. Open-vocabulary object retrieval. In *Robotics: Science and Systems*, 2014.

[7] A. Habibian, T. Mensink, and C. G. M. Snoek. Composite concept discovery for zero-shot video event detection. In *ICMR*, 2014.

[8] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann. Easy samples first: Self-paced reranking for zero-example multimedia search. In *MM*. ACM, 2014.

[9] L. Jiang, T. Mitamura, S.-I. Yu, and A. G. Hauptmann. Zero-example event search using multimodal pseudo relevance feedback. In *ICMR*. ACM, 2014.

[10] L. Jiang, S.-I. Yu, D. Meng, Y. Yang, T. Mitamura, and A. G. Hauptmann. Fast and accurate content-based semantic search in 100m internet videos. In *MM*. ACM, 2015.

[11] A. Karpathy and F. Li. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[12] P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, 2010.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[14] M. Mazloom, A. Habibian, and C. G. M. Snoek. Querying for video events by semantic signatures from few examples. In *MM*. ACM, 2013.

[15] M. Mazloom, X. Li, and C. G. M. Snoek. Few-example video event retrieval using tag propagation. In *ICMR*, 2014.

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[17] G. A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11), 1995.

[18] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quéenot. Trecvid 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID*, 2014.

[19] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR*. ACM, 2006.

[20] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *PAMI*, 2000.

[21] R. Socher, M. Ganjoo, C. D. Manning, and A. Y. Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, 2013.

[22] R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2014.

[23] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CRCV-TR-12-01*.

[24] C. Sun and R. Nevatia. Large-scale web video event classification by use of fisher vectors. In *WACV*, 2013.

[25] J. van Hout, E. Yeh, D. Koelma, C. G. M. Snoek, C. Sun, R. Nevatia, J. Wong, and G. K. Myers. Late fusion and calibration for multimedia event detection using few examples. In *ICASSP*, 2014.

[26] S. Wu, S. Bondugula, F. Luisier, X. Zhuang, and P. Natarajan. Zero-shot event detection using multi-modal fusion of weakly supervised concepts. In *CVPR*, 2014.

[27] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2014.