3-2016

Opinion question answering by sentiment clip localization

Lei PANG

Chong-wah NGO
Singapore Management University, cwngo@smu.edu.sg

Opinion Question Answering by Sentiment Clip Localization

LEI PANG and CHONG-WAH NGO, City University of Hong Kong

This article considers multimedia question answering beyond factoid and how-to questions. We are interested in searching videos for answering opinion-oriented questions that are controversial and hotly debated. Examples of questions include "Should Edward Snowden be pardoned?" and "Obamacare—unconstitutional or not?". These questions often invoke emotional response, either positively or negatively, hence are likely to be better answered by videos than texts, due to the vivid display of emotional signals visible through facial expression and speaking tone. Nevertheless, a potential answer of duration 60s may be embedded in a video of 10min, resulting in degraded user experience compared to reading the answer in text only. Furthermore, a text-based opinion question may be short and vague, while the video answers could be verbal, less structured grammatically, and noisy because of errors in speech transcription. Direct matching of words or syntactic analysis of sentence structure, such as adopted by factoid and how-to question-answering, is unlikely to find video answers. The first problem, the answer localization, is addressed by audiovisual analysis of the emotional signals in videos for locating video segments likely expressing opinions. The second problem, questions and answers matching, is tackled by a deep architecture that nonlinearly matches text words in questions and speeches in videos. Experiments are conducted on eight controversial topics based on questions crawled from Yahoo! Answers and Internet videos from YouTube.

Categories and Subject Descriptors: H.4.m [**Information Systems Applications**]: Miscellaneous

General Terms: Algorithms, Performance, Experimentation

Additional Key Words and Phrases: Multimedia question answering, opinion clip localization, multimodality sentiment analysis

ACM Reference Format:
Lei Pang and Chong-Wah Ngo. 2015. Opinion question answering by sentiment clip localization. ACM Trans. Multimedia Comput. Commun. Appl. 12, 2, Article 31 (November 2015), 19 pages.
DOI: http://dx.doi.org/10.1145/2818711

1. INTRODUCTION

The query "what's your opinion" will retrieve 10 million questions from Yahoo! Answers. Each of these questions could be associated with four to ten answers. Not surprisingly, search engines such as YouTube's will return more than 2 million hits of videos with this query. Social media has, no doubt, a platform to voice opinion, and video is becoming a medium for hosting such social activities. Generally speaking, expressing opinion through video has an advantage in that vivid gesture, speaking tone, and facial expression are more easily comprehended than opinions expressed through the text-only modality. Despite the advantage and the growth in opinion-related videos, text-only

The work described in this article was supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 118812 and CityU 11210514).
Authors' addresses: L. Pang, Creative Media Center M5001, Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Tong, Hong Kong; email: leipang3-c@my.cityu.edu.hk; C.-W. Ngo, Creative Media Center M5003, Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Tong, Hong Kong; email: cscwngo@cityu.edu.hk.

DOI: http://dx.doi.org/10.1145/2818711

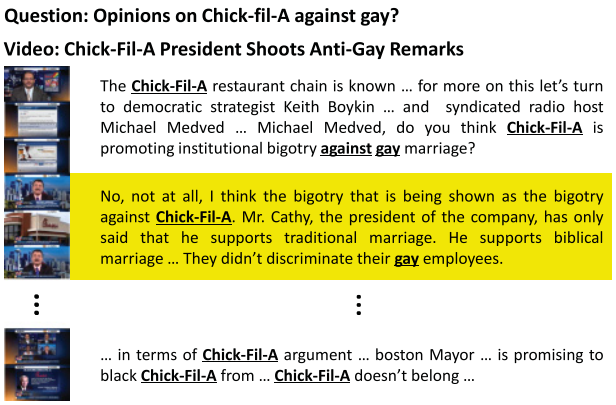

Fig. 1. An example of opinion question and video answer. In this article, we are interested in locating a segment in the video (the highlighted box) that has an answer to the question. The challenges include locating a clip in which an opinion holder expresses views of the question from a lengthy video, and the fact that there are very few overlapping words between text-question and speech transcripts (underlined) for reliable matching.

answers remain the major medium because of the great difficulty of matching and searching video answers, especially when the questions are short, such as "Why occupy Wall Street?".

This article addresses the problem of matching opinion-oriented text questions to video answers. Figure 1 illustrates the problem with an example of the question "Opinions on Chick-fil-A against gay?" with only a few words. The goal is to search a video and locate the segment with potential answers to the question. As observed from the speech transcripts, there are very few words in the target segment matching the question. On the other hand, the keywords "Chick-fil-A" and "gay" are distributed throughout the videos, making the chance of locating the right segment of answer very slim. We address the challenge in three steps: preprocessing the videos by analyzing the sentiment content (Section 3), localizing the opinion-oriented segments based on audiovisual cues (Section 4), and performing nonlinear matching of speech tracks with the text question posted by the user (Section 5). The novelty of this three-step process originates from narrowing the search scope of video answers by integrating nontextual evidence for opinion clip localization, and the proposal of a deep learning architecture for matching text questions and the speech tracks of video clips.

Figure 2 depicts the proposed framework with three major building blocks corresponding to the three-step process. The first building block decomposes input videos into segments by speaker diarization. A series of content processing—including face tracking, speech transcription, and caption extraction—is performed on each segment. The extracted multimodal signals, with the aid of an ontology for sentiment inference, are further analyzed to identify the sentiment talking tracks while filtering the nonsentimental-oriented segments, which are mostly for information rather than opinion expression. The second building block aims to locate the talking tracks with opinion holders, which are referred to as opinion clips in this article. We differentiate the opinion holders from subjects such as anchor persons and journalists, whose roles are to deliver information or moderate discussion rather than voicing personal opinion. To do so, a variety of ad-hoc features obtained through audiovisual processing of sentiment tracks are derived for characterizing opinion holders. A heuristic reasoning algorithm based on an expectation-maximization algorithm is then proposed for the selection of opinion clips. Finally, given a question, the third building block searches and ranks the

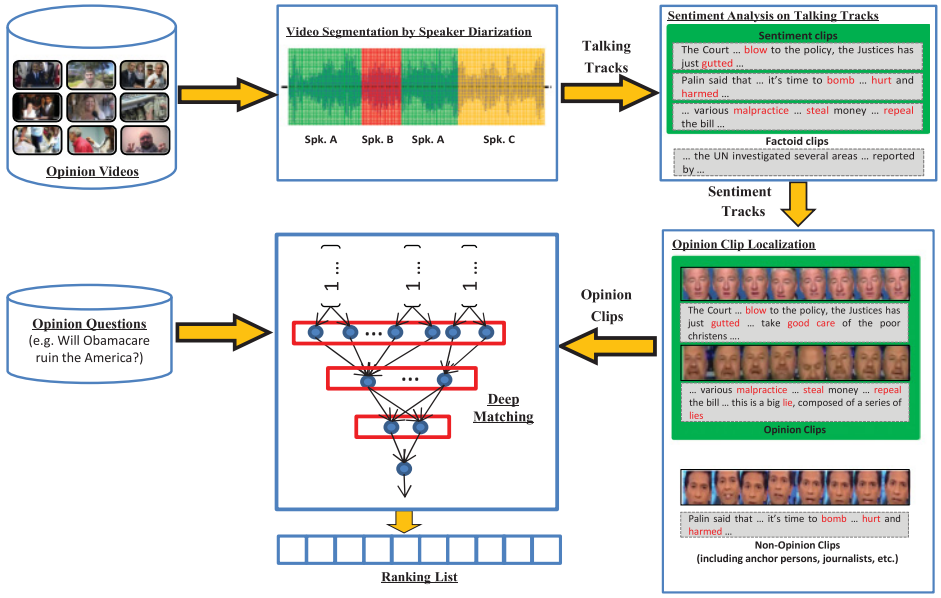

Fig. 2. Framework of the proposed system. The boxes in green indicate potential video segments to be selected for question answering.

opinion clips in the database that match the question. The key component for matching is a deep neural network designed and learned specifically for modeling the latent semantics of topics of interest. The network enables the nonlinear matching of short texts and video speeches through latent semantics, which is potentially more powerful than the traditional ways of keyword matching, such as TF-IDF.

We emphasize that this article addresses only the opinion questions with sentiment tendencies. Answering questions such as "Why are the uniforms in the Olympics for every country in English?" is out of our scope. But questions such as "Do U.S. doctors like Obamacare? If yes or no, why?", which are likely to invoke emotional responses and answers, rather than absolute answers, fall into the scope of this article. Furthermore, this article targets finding sentiment clips with opinion holders, rather than clips showing emotional signals but no opinion holders, as candidate answers. Only in such clips in which emotion is visible through the expressions or gestures of opinion holders can the advantage of using videos as answers to controversial topics be demonstrated. Based on this assumption, only those videos with on-camera opinion holders are considered in this article. The contributions of this work can be summarized as follows:

—*Opinion question and answer* (*QA*): To the best of our knowledge, the answering of opinion questions with videos has not been studied. Previous works on multimedia QA fall in the categories of answering "factoid" and "how-to" questions, which could be tackled by directly matching the texts observed in videos and questions [Li et al. 2010; Chua et al. 2009; Yang et al. 2003]. Opinion QA poses challenges to these works because there could be very few or even no overlap in words between a question and an answer. For example, the best answer chosen in Yahoo! Answers for the question "Why did Romney insult the NAACP (National Association for the Advancement of Colored People)?" is "It was deliberate: he tried to pander to the GOP base of white supremacists & Christian ultra-nationalists," for which there is no word intersection.

—*Clip localization:* Answering by not only providing a video but also the clips that likely answer a question remains a new problem yet to be explored in the literature. This article sheds light on how answer localization can be done in the domain of sentiment-based QA.
—*Cross-media matching:* The traditional way of QA pair matching is by linguistics analysis of sentence structure [Brill et al. 2001; Hermjakob et al. 2002; Radev et al. 2001]. Such analysis is not applicable for video domain as the speech transcripts can be noisy. This article proposes the employment of a deep neutral network, which is learned by QA pairs in text domain, but is leveraged for matching video answers. This, again, is a new technique not previously attempted.

The application scenario of the proposed work is to retrieve and rank videos that could answer opinion questions. Particularly, the locations where potential answers reside in a video can be made known to facilitate video browsing. Furthermore, video segments with opinion holders who deliver stronger emotion signals are preferred when presenting the potential answers. The remainder of this article is organized as follows. Section 2 describes the related works. Section 3 presents the preprocessing step, which includes the extraction, classification, and filtering of talking tracks. Only sentiment tracks are retained after the preprocessing step. Section 4 further describes an unsupervised learning algorithm for locating opinion clips out of the sentiment talking tracks. Section 5 presents the architecture of a deep neural network in matching the text-based questions and opinion clips. The techniques for topic modeling and parameter learning are described. Section 6 contains experimental results, and Section 7 presents our conclusions.

2. RELATED WORK

In the literature, multimedia question answering (MQA) is mostly tackled by mining answers from a large volume of multimedia content (e.g., images and videos) on the Web. These works could be broadly classified into two categories based on the type of questions. The first category is to answer "factoid" questions. One of the earliest developed system is VideoQA [Yang et al. 2003], which leverages visual content, Automatic Speech Recognition (ASR) transcripts, and online information for locating news video segments as answers for factoid questions. A passage retrieval algorithm for QA was developed in the video documentary domain [Wu and Yang 2008]. By video caption recognition and pattern-based passage ranking, the algorithm returns the passages associated with short video clips as answers. Following these works, several video QA systems were also proposed to investigate the "factoid" question answering, but in different domains, such as educational videos and bilingual videos [Cao and Nunamaker 2004; Lee et al. 2009; Wu et al. 2004].

The second category is to answer "how-to" questions. In Li et al. [2010], a community-based QA system supporting how-to questions was developed for retrieving Web videos as answers in the domain of consumer electronics. A unified framework for tackling both "factoid" and "how-to" questions was proposed in Chua et al. [2009], by extending the text-based QA techniques to multimedia QA. The system was designed to find multimedia answers from Web-scale media resources such as Flickr and YouTube. More recently, Nie et al. [2011] presented a method to predict the media type (text, image, or video) that will best answer the "factoid" and "how-to" questions. Based on the predicted media type, images and videos are retrieved for enriching the text answers.

There were also a few research works devoted to multimodal question answering. Specifically, the questions are composed of multimedia objects, such as images and videos, in addition to text. In Kacmarcik [2005], a QA system was proposed to allow the

players in a virtual world to pose questions without textual input. By giving annotated semantic information to the objects in the virtual world, questions and answers are generated based on contextual information among the objects. However, the system provides very limited questions and answers to the players. Given a photo as question, photo-based QA [Yeh et al. 2008] exploited visual recognition techniques to answer the factoid questions about physical objects in photos. A more general multimodal QA system was developed in Zhang et al. [2012] to answer questions of various types, including "factoid," "how-to," and "opinion" questions. However, the system provides only textual rather than multimedia answers. There are also some commercial Web sites that have emerged to provide videos as answers for factoid and how-to questions. The most representative one is eHow[1], which provides how-to videos by recruiting amateur photographers to shoot problem-solving videos. However, producing these videos is expensive compared to the automatic search of video answers, as we describe in this article.

Our work is also different from the traditional QA on how to deal with the lexical and stylistic gaps between the question and answer domains. In text QA, these gaps are usually bridged by question reformulation, from rule-based rewrites [Brill et al. 2001], more sophisticated paraphrases [Hermjakob et al. 2002], to question-to-answer translations [Radev et al. 2001]. In multimedia QA, the gaps are usually bridged by query expansion [Cao and Nunamaker 2004; Chua et al. 2009; Li et al. 2010; Yang et al. 2003]. Keywords, which are expanded with contextually related words from WordNet and Web resources, are used for answer matching. In Wang et al. [2009], the gaps are bypassed by posting the problem of QA as the similar questions search in community QA Web sites. The developed technique decomposes the parse tree of a question into tree fragments recursively, and measures the similarity between two questions based on the degree of overlap in tree fragments. This technique works well only for "factoid" and "how-to" questions and not opinion questions, for which the answers can vary more wildly in both lexicon and stylistics. In this article, inspired by Lu and Li [2013], we employ recent advances in deep learning to bridge the gaps by capturing the localness and hierarchical intrinsics of sentences for question answering.

3. SENTIMENT DETECTION

This section outlines the method for detecting sentiment-oriented speeches in the video domain. We start by presenting the extraction of talking tracks (Section 3.1), followed by classification of the talking tracks based on their sentiment content (Section 3.2). The major challenge is that speech analysis alone, obtained through ASR, is imperfect for sentiment analysis. We approach this problem by considering multiple sources of modalities for analysis.

3.1. Extracting Talking Tracks

We employ speaker diarization for video partitioning [Rouvier et al. 2013]. Different from the definition of shot, each partition corresponds to a segment with a person speaking. The technique performs hierarchical agglomerative clustering of speakers by Cross-likelihood Ratio (CLR) on the audio track. No prior information, such as the number of speakers or samples of voices, is required.

Given the video partitions in which each has a speaker identity, we are interested only in the segments with talking heads. Voiceover segments, which generally deliver only the background information of a topic, are excluded. To do this, the speaker

[1]http://www.ehow.com/.

face in a segment has to be detected and tracked[2]. We employ the Viola-Jones detector [Viola and Jones 2004] for detecting faces, followed by the Kanade-Lucas-Tomasi (KLT) method [Shi and Tomasi 1994] for tracking faces across frames. Face tracks extracted in such a way were shown to be robust to occlusion and drift problem as demonstrated in Everingham et al. [2006]. To determine whether a person is talking, the mouth region has to be tracked also. We utilize the dark area inside a mouth region as a cue for detection. A face track is declared as belonging to a talking person if there is a consecutive and significant change in the proportion of dark area to the size of mouth area. To ensure the robustness of tracking, a series of steps is carried out, including normalizing the face to a canonical pose with a resolution of 80×80 pixels based on the position of the eyes, and performing a histogram equalization to diminish the sensitivity to skin colors.

3.2. Sentiment Classification

A simple way of identifying whether the speech content of a talking track is sentimental is by spotting words such as *support* and *repeal* from speech. However, this method performs poorly in practice. Instead, we extract three features from speeches, captions, and metadata, respectively, from a track, then develop a Naive Bayes classifier for sentiment detection. Each feature is represented as a binary vector of 155,287 dimensions, where each dimension refers to a word in SentiWordNet [Baccianella et al. 2010]. An element of the vector is set to the value of 1 if the corresponding word is present. Denoting $\Upsilon \in \{A, C, M\}$ as a binary vector of N dimensions for speeches (A), captions (C), and metadata (M), respectively, we can approximate the probability distribution of a feature given a sentiment $s \in \{positivity, negativity, neutrality\}$ as

$$P(\Upsilon \mid s) = \prod_{j=1}^{N} P(w_j \mid s)^{n_j(\Upsilon)}, \tag{1}$$

assuming that words are conditionally independent. The function $n_j(\Upsilon)$ outputs 1 if word w_j presents and 0 otherwise. $P(w_j \mid s)$ is estimated by SentiWordNet, which outputs a value in the range of 0 to 1, indicating the degree of sentiment.

By Bayes' rule, the sentiment s of a talking track T is defined as

$$P(s \mid T) = \frac{P(s)P(T \mid s)}{Z}, \tag{2}$$

where Z is a normalizing constant that can be omitted. Because metadata provides prior regarding the sentiment of a video, we estimate $P(s)$ with $P(M \mid s)$ in Equation (1). Furthermore, $P(T \mid s)$ is jointly estimated by $P(A \mid s)$ and $P(C \mid s)$. To this end, we develop a Naive Bayes classifier as

$$P(s \mid T) \sim P(M \mid s)P(A \mid s)P(C \mid s) \tag{3a}$$

$$\sim \prod_{j=1}^{N} P(w_j \mid s)^{n_j(M)+n_j(A)+n_j(C)}. \tag{3b}$$

We want to emphasize that, by using SentiWordNet, no training data is required for classifier learning. Based on Equation 3.2(b), we use log-likelihood to estimate the

[2]Note that using audio for talking-head detection is not necessary because speaker diarization basically ensures that each segment contains only one speaker.

sentiment score as follows:

$$L(s \mid T) = \sum_{j=1}^{N} log P(w_j \mid s)^{n_j(M)+n_j(A)+n_j(C)} \quad (4)$$

To this end, each talking track is associated with three sentiment scores. The tracks with higher scores in neutrality than positivity and negativity are filtered out from further processing. Note that, in addition to SentiWordNet, there are other methods for sentiment analysis [Machajdik and Hanbury 2010; Borth et al. 2013; Chen et al. 2014], which operate directly on image features. Nevertheless, these works cannot be directly applied to our problem. This is mainly because we consider only video clips with human subjects as the focus, hence the sentiment signals are mostly from spoken content and surrounding texts rather than visual effect.

4. OPINION CLIP LOCALIZATION

In this section, we are interested in locating the video segments that have opinion holders expressing views for a topic of interest. Under our problem definition, opinion clip localization is a task equivalent to the selection of sentiment-oriented talking tracks from opinion holders. Section 4.1 presents the ad-hoc features derived for characterizing the opinion holder. Based on these features, an expectation-maximization (EM) algorithm is proposed for opinion clip localization (Section 4.2).

4.1. Features

We adopt a heuristic approach for the identification of opinion holders. First, the duration of delivering opinion, indicated by the length of a talking track, shall be relatively longer than the ones from nonopinion holders such as the host or anchor person. Second, an opinion holder should possess a higher sentiment score, as computed by Equation (4). Third, the name of the opinion holder is often shown in the video caption. In contrast, a nonopinion holder tends to speak in a relatively shorter duration, appears at the beginning and end of a talk show, and mentions frequently the name of the opinion holder. Sometimes, there are voiceover segments by a nonopinion holder introducing the background history of a topic. To vividly translate these heuristics into numeric scores, a total of 11 features based on audiovisual cue processing are extracted for representing a talking track. These features are briefly described as follows.

—*Visual appearance frequency* (f_1). Face diarization [Khoury et al. 2013], which groups face tracks based on visual similarity, is performed to cluster the talking tracks in a video. The similarity is measured based on the set of facial feature points extracted from faces. For a given talking track T_i, the feature f_1 counts the percentage of face tracks in a video that falls into the same cluster as T_i.
—*Audio appearance frequency* (f_2). Based on the result of speaker diarization in Section 3.1, each talking track T_i is associated with a speaker cluster. Similar to f_1, but based on audio processing, the feature f_2 measures the percentage of talking tracks falling into the same speaker cluster as T_i.
—*Voice-over* (f_3). Video segments with voice but without face appearances are regarded as voiceover segments. The feature f_3 of a talking track T_i is assigned to the value of 1 if there exists a voiceover segment that falls into the same speaker cluster as T_i, and 0 otherwise.
—*Duration* (f_4). The duration of a talking track.
—*Temporal location* (f_5). A discrete value, ranging from 1 to 3, is assigned depending on whether a talking track appears at the beginning (1), middle (2), or end (3) of a

video sequence. The beginning is defined as the first 5% of video length and the end is defined as the last 5% of video length.

—*Number of person names* (f_6, f_7, f_8). Named-entity recognition is performed on the ASR transcript. The feature f_6 measures the number of distinct person names that are mentioned in the speech of a talking person. The feature f_7 (f_8), on the other hand, counts the number of distinct names detected n seconds before (after) the speech of a talking person. A nonopinion holder is expected to possess a higher value of f_6 for introducing opinion holder(s). In reverse, an opinion holder is expected to possess higher values of f_7 and f_8 for introductions by a host before the start and after the end of one's speech.

—*Number of names in subtitles* (f_9). The name of opinion holder is assumed appearing in subtitles together with the opinion holder. The feature f_9 counts the number of distinct names found in the captions and subtitles along with a talking track.

—*Number of faces* (f_{10}). A typical studio setup for interview is that all the faces of hosts and opinion holders are visible at the beginning, followed by the middle or close-up shots of each opinion holder when expressing an opinion. The feature f_{10} counts the number of face tracks appearing together with a talking track T_i; ideally, the value should be lower for a talking track belonging to an opinion holder.

—*Sentiment score* (f_{11}). The feature f_{11} is a score representing the degree of positivity or negativity in sentiment, as computed by Equation (4).

Ideally, an opinion holder takes longer to express opinions (higher f_4 value) than nonopinion holders, while the frequency of an opinion holder appearing in a video is usually lower (lower f_1 and f_2 values). Opinion holders should express an opinions for a controversial topic (higher value of f_{11}), probably supplemented with visual cues such as their names being introduced by host (higher f_7 value) and printed on screen (higher f_9 value). In addition, clips with only one opinion holder in the scene (lower f_{10} value) are more focused and thus preferred.

4.2. EM Algorithm

Based on the 11 designed features, we adopt an EM algorithm for clustering the talking tracks into the categories of opinion and nonopinion holders. Assuming that these features are conditionally independent, the probability of a talking track $T_i \in T$ given a category c_j is:

$$p(T_i \mid c_j) = \prod_{k=1}^{K} p(t_{i,k} \mid c_j), \tag{5}$$

where $t_{i,k}$ denotes the the kth feature of T_i, and $K = 11$ is the length of the feature vector. We model the features with continuous value (f_1, f_2, f_4, f_{11}) using normal distribution, and the features with discrete value using multinomial distribution. The model parameters θ are estimated by maximizing the log-likelihood of joint distribution in E step:

$$\mathcal{L}(T;\theta) = \log\left(\prod_{i=1}^{N} p(T_i \mid \theta)\right) = \sum_{i=1}^{N} log\left(\sum_{k=1}^{K} p(c_j)p(t_{i,k} \mid c_j, \theta)\right), \tag{6}$$

where N is the number of talking tracks in T. In M step, the posterior probability of c_j is updated by Bayes' rule:

$$p(c_j \mid T_i)^{new} = \frac{p(c_j)^{old} p(T_i \mid c_j)^{old}}{p(c_1)^{old} p(T_i \mid c_1)^{old} + p(c_2)^{old} p(T_i \mid c_2)^{old}}. \tag{7}$$

The parameters of both normal and multinomial distributions for each category c_j are updated separately. For features modeled with normal distribution, we update the mean $\mu_{j,k}^{new}$ and variance $\sigma_{j,k}^{new}$ with the following equations:

$$\mu_{j,k}^{new} = \frac{\sum_{i=1}^{N} p(c_j \mid T_i)^{new} t_{i,k}}{\sum_{i=1}^{N} p(c_j \mid T_i)^{new}} \tag{8}$$

$$\sigma_{j,k}^{new} = \frac{\sum_{i=1}^{N} p(c_j \mid T_i)^{new} (t_{i,k} - \mu_{j,k}^{new})^2}{\sum_{i=1}^{N} p(c_j \mid T_i)^{new}}. \tag{9}$$

For multinomial distribution, the marginal probabilities over features are directly updated as follows:

$$p(t_{i,k} \mid c_j)^{new} = \prod_{x=1}^{X} [p(t_{i,k} = x \mid c_j)^{new}]^{\mathbf{1}(t_{i,k}=x)}, \tag{10}$$

where X is the number of possible values in feature $t_{i,k}$ and $\mathbf{1}(\cdot)$ is an indicator function. The probability $p(t_{i,k} = x \mid c_j)^{new}$ is further smoothed as follows:

$$p(t_{i,k} = x \mid c_j)^{new} = \frac{1 + \sum_{r=1}^{N} p(c_j \mid T_r)^{new} \mathbf{1}(t_{r,k} = x)}{X + \sum_{r=1}^{N} p(c_j \mid T_r)^{new}}. \tag{11}$$

To this end, the category model is updated as follows:

$$p(c_j)^{new} \approx \frac{1}{N} \sum_{i=1}^{N} p(c_j \mid T_i)^{new} \tag{12}$$

$$p(T_i \mid c_j)^{new} = \prod_{k=1}^{K} p(T_{i,k} \mid c_j)^{new}. \tag{13}$$

In the implementation, we employ K-means to estimate the initial parameters. E-step and M-step are iterated until convergence. Finally, the cluster with the lower value of f_{10} is selected, and the corresponding talking tracks are regarded as belonging to the opinion holders. The heuristics is practical because nonopinion holders often appear together with one or several speakers, and the frequency of appearance is usually higher. This strategy also works well for personal videos, which usually have one person talking throughout a video.

5. QUESTION-ANSWERING BY DEEP LEARNING

Next, we describe the matching of text questions with candidate video answers, more specifically, the extracted opinion clips as presented in the previous section. Generally speaking, question-answering is by no means an easy task because of the lexical and stylistic gaps in how questions are asked and answers are elaborated. Lexical gap refers to the vocabulary difference between questions and answers, while stylistic gap refers to the syntactic difference in sentence structure. The gaps lead to ambiguity in word features. Traditional relevance measures based on frequency of overlapping words, such as cosine similarity and KL divergence, are not effective for question-answer semantic modeling. Inspired by the ideas in DeepMatch [Lu and Li 2013], which models the relevance between questions and answers with a deep neural network, we construct a new deep architecture (called DeepHPam) with the hierarchical Pachinko

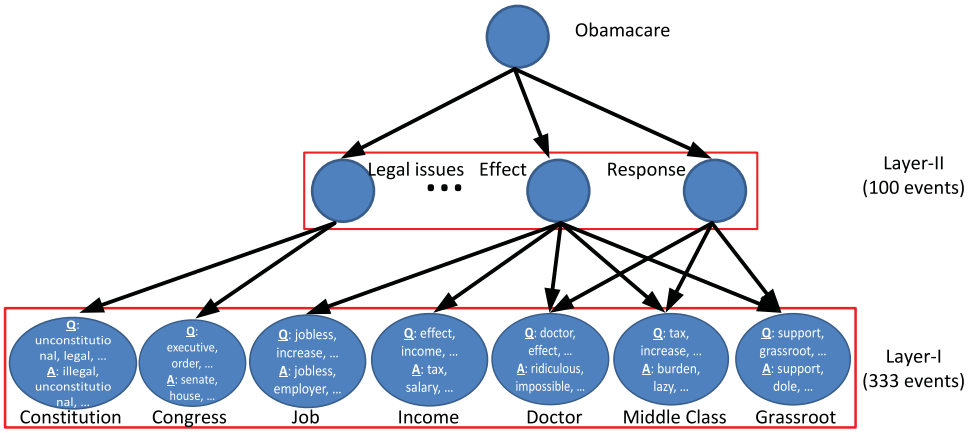

Fig. 3. An example of DAG for the topic "Obamacare." The label on each subtopic is manually defined for illustration purposes. The symbols "Q:" and "A:" indicate the words from question and answer domains, respectively.

Allocation Model (hPam) [Mimno et al. 2007] to make a composite decision on matching opinion clips to questions in a hierarchical way (Section 5.1). The major improvement of DeepHPam over DeepMatch is that we incorporate the probability distribution over the latent semantics of topics learned by hPam into the construction (Section 5.2) and initialization (Section 5.3) of the hierarchical structure of the neural network.

5.1. Topic Modeling by hPam

We view question-answering as a translation problem, for which questions and answers are treated as two separate domains. The task is to compute how likely an answer can be "translated" from a question of a different domain. To achieve this, two vocabularies, one from each domain, are generated. Denote $|V_q|$ and $|V_a|$ as the sizes of question and answer vocabularies. A QA pair is represented by a vector of length $|V_q| + |V_a|$, by concatenating both vocabularies. An element in the vector encodes the frequency of a word. Using the vectors as input, the algorithm hPam [Mimno et al. 2007] captures the salient patterns composed of words from different domains. For example, for the topic "Obamacare," the pattern (or event) "job" is described by two sets of words, "kill, job, unemployed ..." and "trim, company, fire ...," in the question and answer domains, respectively.

With hPam, a layered directed acyclic graph (DAG) is constructed to model the events under a topic. Here, we mean "topic" to be a subject of discussion, such as "Obamacare" and "Edward Snowden." An event is regarded as a "latent subtopic" mined by hPam for characterizing topic generation. DAG organizes the major events of a topic into a two-level hierarchical graph. Figure 3 shows an example of DAG constructed for the topic "Obamacare," in which each node represents an event encoded by a set of words. For example, the event "grass root" is composed of words such as "bankrupt" and "dole." Furthermore, each word is associated with a probability indicating its likelihood to an event. The two-level hierarchy models the event granularity, describing how an event at a higher layer is generated from a mixture of events at a lower level. For example, the event "effect" at Layer II can be jointly modeled with the events "job" and "income" in Layer I. By DAG, a question or answer can be represented by "latent subtopics" or events. For example, the question "What might the effects of Obamacare be on jobs for lower middle citizens?" is jointly described by "job," "income," and "middle class" by using the DAG shown in Figure 3. An advantage of using this representation is that a

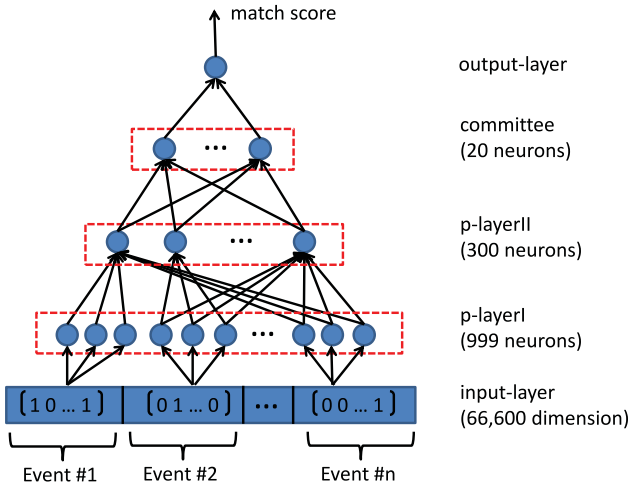

Fig. 4. Illustration of the neural network based on hierarchical topic structure.

potential answer to a question does not necessarily have an overlap in words, as long as it is sharing similar event distributions.

Note that each event in DAG is composed of words from both domains. To suppress noise, for each event in Layer I of DAG, we pick only the top-10 words with the highest probabilities from the question domain, and the top-20 words from the answer domain for encoding a question–answer (QA) pair. The reason for picking more words from the answer domain is due to the fact that the length of an answer is used to be longer than a question. To this end, with respect to an event, a QA pair is represented as a vector of 10×20 dimensions, each dimension corresponding to a word pair composed of words in the question and answer domains. We use binary vector representation in this case. Specifically, an element in the vector is set to a value of 1 if the corresponding word pair is observed in the QA pair. The vector is then fed into the neural network for learning and classification.

5.2. Deep Architecture

The architecture of the neural network is depicted in Figure 4. The first two hidden layers, p-layerI and p-layerII, correspond to the first and second layers of a DAG, in which each event is jointly modeled by three neurons. In the implementation of hPam, we learn 333 and 100 events for the first and second layers, respectively. As a result, p-layerI and p-layerII in the deep architecture are composed of 999 and 300 neurons, correspondingly, and respectively. The input to the neural network is composed of 333 binary vectors, each of 10×20 dimensions generated by one of the events in DAG. The connection between the input layer and p-layerI is based on the event to which a binary vector belongs. In other words, for a binary vector of an event, the 10×20 elements are fully connected to the three neurons in charge of this event, but have no connection with any other neurons.

The neurons between p-layerI and p-layerII are partially connected, simulating the hierarchy of DAG. A committee layer of 20 neurons is added and fully connected with p-layerII to model the event relationships at a higher semantic level. Finally, the output layer generates a matching score, indicating the goodness of translation from question to answer. For all the neurons, we adopt the sigmoid function as the activation function.

5.3. Hyper Parameter Learning

In the architecture, there are in total 506,840 parameters to be learned; only around 40% are kept to be nonzero values. The parameters between p-layerI and p-layerII are initialized based on the probability distribution learned by hPam in modeling events between the two levels of hierarchy. Similarly, the parameters between input layer and p-layerI are initialized based on the outcome of hPam. Specifically, based on Bayes' rule, we multiply the probabilities of two words in a pair as the initial value for the parameter connecting a word pair and a neuron. The parameter values are updated and learned by employing a discriminative training strategy with a large margin objective. The training instance is in the form of triple $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-)$, where $\mathbf{x}$ is a question, $\mathbf{y}^+$ is the corresponding answer, and $\mathbf{y}^-$ is a false answer. We define the following ranking-based loss as objective:

$$\mathcal{L}(\mathcal{W}, \mathcal{D}_{trn}) = \sum_{(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-) \in \mathcal{D}_{trn}} e_{\mathcal{W}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-) + R(\mathcal{W}), \tag{14}$$

where $R(\mathcal{W})$ is the L2 regularization term, and $e_{\mathcal{W}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$ is the error for a triple $(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-)$, given by the following large margin form:

$$e_{\mathcal{W}}(\mathbf{x}_i, \mathbf{y}_i^+, \mathbf{y}_i^-) = \max(0, m + s(\mathbf{x}_i, \mathbf{y}_i^-) - s(\mathbf{x}_i, \mathbf{y}_i^+)), \tag{15}$$

where $s(\mathbf{x}, \mathbf{y})$ represents the score of the output layer and $0 < m < 1$ controls the margin in training. Here, we empirically set $m = 0.1$. We use stochastic subgradient descent with mini-batches [Lecun et al. 1998] for training, where each batch consists of 20 randomly generated triples $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-)$.

6. EXPERIMENTS

The experiments are split into three parts for evaluating the accuracy of opinion clip localization (Section 6.2), opinion question answering (Section 6.3) and a user study (Section 6.4) supporting the claim in this article.

6.1. Datasets

6.1.1. Video Dataset. A Web video dataset named OWE (Opinion Web vidEo) was constructed for experimentation. The dataset is composed of eight opinion-oriented topics, consisting of 800 videos with a duration of around 340 hours. The eight topics are "Affordable Care Act" (Obamacare), "Syria chemical weapons" (Syria), "Edward Snowden" (Snowden), "US government shutdown 2013" (Shutdown), "Mitt Romney's Tax Return" (Tax Return), "Chick-fil-A same-sex marriage controversy" (Chick-fil-A), "Occupy Wall Street" (Occupy W.S.), and "Romney's speech to NAACP" (NAACP Speech). These topics are highly controversial and have triggered many discussions in Yahoo! Answers, thus are selected for experimentation. In the dataset, about 26% of the videos are personal videos; the remainder are official news videos, such as talk shows. Table I shows the details of the OWE dataset.

To construct the dataset, the top-100 ranked videos of each topic along with their metadata and ASR[3] were downloaded from YouTube. The toolset Tesseract[4] was employed to extract captions and subtitles from the videos. The talking tracks, which were extracted from the videos based on the method in Section 3.1, were manually labeled by human subjects. Three human subjects, who are familiar with the eight

[3]As personal videos were often captured in an indoor environment, the results of speech recognition are acceptable for question-answering. Note that some ASR of personal videos are actually transcripts uploaded by the video owners.

[4]https://code.google.com/p/tesseract-ocr/.

Table I. Statistics for the Dataset OWE

Topics	# talking tracks	# sentiment tracks	# opinion clips
Obamacare	1082	979	904
Syria	1536	1381	993
Snowden	1339	1116	871
Shutdown	1332	1253	1001
Tax Return	957	738	683
Chick-fil-A	1294	1037	697
Occupy W.S.	1810	1206	1099
NAACP Speech	981	913	873
All	10331	8623	7121

topics, were recruited for ground-truth generation. The evaluators were asked to first label whether a talking track is sentiment-oriented based on speech content. The subjects were further instructed to judge whether a sentiment-oriented track contains an opinion holder expressing the view for a topic of interest. During this process, we guaranteed that each talking track would be evaluated by at least two evaluators. Any inconsistency in labeling would be picked up and judged by the third evaluator. Table I shows the statistics of the OWE dataset, in which there are around 7,000 opinion clips out of about 10,000 talking tracks being annotated in the dataset.

6.1.2. QA Dataset. Another corpus composed of question-answer pairs from Yahoo! Answers was constructed for the eight topics, using the same keywords posted to YouTube. There is a total of 53,611 QA pairs and each question has 7.6 answers, on average. Based on our observation, most of these questions are opinion questions or opinion-related. The QA pairs are preprocessed by stopword removal and stemming. The average vocabulary sizes for questions and answers are 18,630 and 36,896, respectively. There are 42,000 QA pairs generated for each topic, on average.

To learn the network parameters, we sampled 31,000 $(\mathbf{x}, \mathbf{y}^+, \mathbf{y}^-)$ triples from the collected QA pairs for each topic. The answer $\mathbf{y}^+$ was selected from either the best answer picked by an asker or any answer given by a user. The false answer $\mathbf{y}^-$ was randomly selected from the answer pool of a topic. In learning the neural network, a random subset of 1,000 triples was picked as the validation set for parameter tuning, including setting the coefficients for L2 regularization.

The testing set was formed by randomly picking ten questions per topic from the QA corpus. The performance is measured by nDCG@10 with three levels (2, 1, 0) of relevance; 2 means a retrieved opinion clip fully answering a question, 1 means partial answer, and 0 means not relevant to a question. For example, for the question "Opinions on Chick-fil-A against gays?", the opinion clip "No, not at all, I think the bigotry that is being shown as the bigotry against Chick-fil-A. Mr. Cathy, the president of the company, has only said that he supports traditional marriage. He supports biblical marriage . . . They didn't discriminate their gay employees." was labeled as 2. Another clip, "It's fascism for these guys to say you cannot come to my town if I disagree with your political view. That's fascism.", was labeled as 1, since the speech is about the comments of Boston's mayor: "Anti-gay Chick-fil-A not welcome in this city." The relevance score for the clip "Work hard and don't let anything stop you. Happy in Chick-fil-A" was labeled as 0. A total of 16 human subjects were invited for answer labeling. Each topic was labeled by two subjects, and the average score between them for each question and video pair is used as the ground-truth. On average, each subject labeled 500 question–video pairs pooled from five approaches (described in Section 6.3). In the experiment, we combined the human labels to simulate the ideal DCG (IDCG); the average value of the IDCG is 28.1.

Table II. Accuracy of Localizing Sentiment-Oriented Talking Tracks

Topics	ASR	ASR+C	ASR+M	ASR+M+C
Obamacare	0.681	0.693	0.697	**0.735**
Syria	0.705	0.729	0.763	**0.797**
Snowden	0.678	0.711	0.754	**0.810**
Shutdown	0.677	0.696	0.711	**0.746**
Tax Return	0.690	0.721	0.749	**0.773**
Chick-fil-A	0.591	0.613	0.621	**0.655**
Occupy W.S.	0.578	0.593	0.619	**0.622**
NAACP Speech	0.697	0.751	0.739	**0.790**
All	0.657	0.682	0.702	**0.734**

Note: M = video metadata, C = caption.

Table III. Accuracy of the Opinion Clip Localization

Topics	Random	K-Means	EM Learning
Obamacare	0.287	0.513	**0.631**
Syria	0.391	0.505	**0.619**
Snowden	0.357	0.492	**0.597**
Shutdown	0.311	0.501	**0.573**
Tax Return	0.306	0.487	**0.502**
Chick-fil-A	0.332	0.476	**0.514**
Occupy W.S.	0.304	0.510	**0.527**
NAACP Speech	0.291	0.503	**0.522**
Average	0.325	0.499	**0.562**

Note: The best performance is highlighted.

6.2. Opinion Localization

This section evaluates the accuracy of locating opinion clips. Table II first shows the results of detecting sentiment talking tracks using different combinations of ASR, caption (C) and metadata (M). As presented in Section 3.2, the prior probability for Equation (2) is directly derived from metadata. For the combinations without metadata, we set the prior as 0.5 in the experiment. The result shows that, by ASR only, the accuracy of detection is around 0.65. This result is boosted by 4% and 7%, respectively, when fusing with a caption and metadata. The recognition rate of OCR is around 85%, which is the reason that a caption introduces less improvement than metadata. The best performance is attained when all the modalities are considered. From our result analysis, both captions and metadata are good supplements of ASR. For example, the relevance score for a clip entitled "We, young Americans, pay for the services and we don't use" is weak in sentiment. But when combining with captions such as "opposer of Obamacare" extracted from the video track or the description "Why a majority of Americans remain opposed to Obamacare" from metadata, the speech potentially expresses an opinion and can be used for sentiment-based question-answering.

Table III further shows the result of opinion clip localization. We compare our approach (EM) to two baselines implemented based on k-means and random guess (Random), respectively. Both EM and k-means outperform "Random" by a large margin. By modeling distributions of multiple feature types, EM is more probabilistically sound compared to k-means, which is biased towards generating two equally sized clusters. As indicated in Table III, EM achieves the best performance, 0.562, with an improvement of 13% over k-means. In addition, we also experimented with the difference between EM and a supervised learning algorithm. Using 20% sentiment tracks randomly selected from the eight topics as training samples, an SVM classifier with χ^2 RBF kernel is learned for classification. We compared the performance of EM and SVM on the remaining 80% of sentiment tracks. The accuracy of SVM is 0.573, which is sightly better than EM, having an accuracy of 0.565.

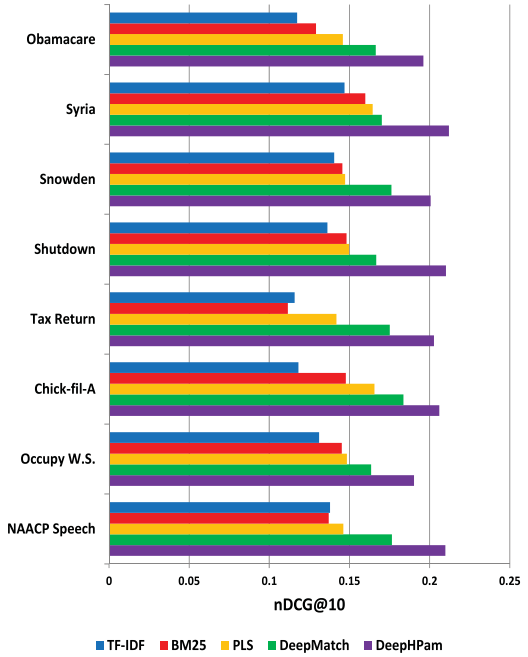

Fig. 5. Performance comparison of five approaches on opinion question-answering, measured in terms of average nDCG@10.

6.3. Opinion Question Answering

We compare our proposed model DeepHPam with four other methods: TF-IDF, BM25 [Robertson et al. 1996], partial least square (PLS) [Wu et al. 2013; Rosipal and Krmer 2006], and DeepMatch [Lu and Li 2013]. The first three methods can be considered as linear models; DeepHPam and DeepMatch are nonlinear models. For BM25, we set $k_1 = 1.2$ and $b = 0.75$ according to the safe range suggested in Manning et al. [2008]. PLS projects the TF-IDF vectors from question and answer domains into a latent subspace by linear mapping, and then measures the matching between them by dot product. Using the QA corpus mentioned in Section 6.1.2, PLS learns a subspace of 300 dimensions for projection. The architecture of DeepMatch is similar to DeepHPam: 999-300-20 neurons in the first, second, and committee layers, respectively, all with sigmoid functions. In DeepMatch, Latent Dirichlet allocation (LDA) is applied for generating the nodes (or events) at p-layerI and p-layerII separately. The connections between the nodes of the two layers are determined on an ad-hoc basis based on word overlapping between two nodes. The parameters between p-layerI and p-layerII are randomly initialized. Note that, although DeepHPam and DeepMatch share the same representation for an input QA pair, the binary vectors (as described in Section 5.2) corresponding to an input are different as the underlying event models are generated by hPam and LDA, respectively.

The experiment was conducted by matching text questions to the opinion clips mined in Section 6.2. The ASR of a clip is extracted, then the similarity to a given question is measured. Figure 5 shows the result of performance comparison in terms of nDCG@10. Basically, nonlinear matching of QA pairs shows an improvement of 56% (by DeepHPam) and 32% (by DeepMatch) over the baseline TF-IDF. Because of the lexical gap between questions and answers, only a few words co-occur between them. As a result, short clips are preferred in retrieval based on TF-IDF and BM25. Obviously, short

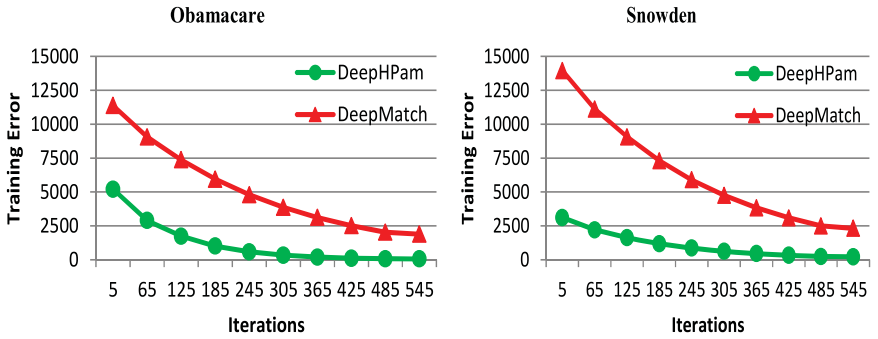

Fig. 6. Trends of ranking-based loss of DeepHPam and DeepMatch on the topics "Obamacare" and "Snowden." The loss is summed over the entire training set. The learning rate is 0.01 and the coefficient of L2 regularization term is 0.0001. The training losses of other topics follow similar trends.

clips may not be informative enough for answering opinion questions. For example, the speech "Who cares about Romney's tax returns? We know he is rich, that's enough." is retrieved as the top-1 opinion clip by BM25 for the question "Why is Mitt Romney hiding his tax returns?". By DeepHPam, we are able to retrieve the clip "not only he hasn't paid about thirteen percent in taxes in the years that we know, he pumping up and up in just one year all the way up to the fourteen percent now. But is there money in the Cayman Islands? Is there money in Bermuda area, in the Swiss bank, in China and every? . . ." In this example, the words "tax" and "return" in the question domain are modeled together with 'Cayman," 'Bermuda," and "bank" in the answer domain by a neuron, thus a better answer can be successfully retrieved. Another observation is that PLS is incapable of dealing with words of a complex relationship using linear projection. For example, the word "abuse" is used to describe the usage of "chemical weapons" and the "power of president to launch military strike." This relationship can be modeled by DeepMatch and DeepHPam, but not PLS.

To show the advantage of using hPam [Mimno et al. 2007] for building and initializing the parameters in DeepHPam, Figure 6 compares the learning efficiency of DeepHPam and DeepMatch. As shown in the figure, the initial loss of DeepHPam is much less and the convergence speed is quicker than DeepMatch. In addition to learning efficiency, DeepHPam consistently outperforms DeepMatch across all topics. We browsed through and compared the opinion clips retrieved by both approaches. Our observation is that DeepHPam is more capable of retrieving video answers, which are labeled as "2" (i.e., fully answer a question), while DeepMatch is more susceptible to frequent words, which lack specificity in question answering. We attribute this to the use of hPam in constructing a more precise neuron connectivity in the deep architecture, versus DeepMatch, in which the connections are merely established upon overlapping words between player-I and player-II. Figure 7 shows examples of the opinion clips retrieved by DeepHPam. The examples range from very specific questions, such as "Do U.S. doctors like Obamacare?", to general questions, such as "Do you think Obama will attack Syria?".

6.4. User Study

We are interested to know whether the retrieved video answers are more preferable than text answers. To verify this, we conducted a subjective test by providing users both video and text answers, asking them to pick the preferred answer medium. A total of 12 evaluators (5 females and 7 males) were invited for the user study. Each evaluator was assigned 8 questions and 16 answers. Only questions that have the best text answers

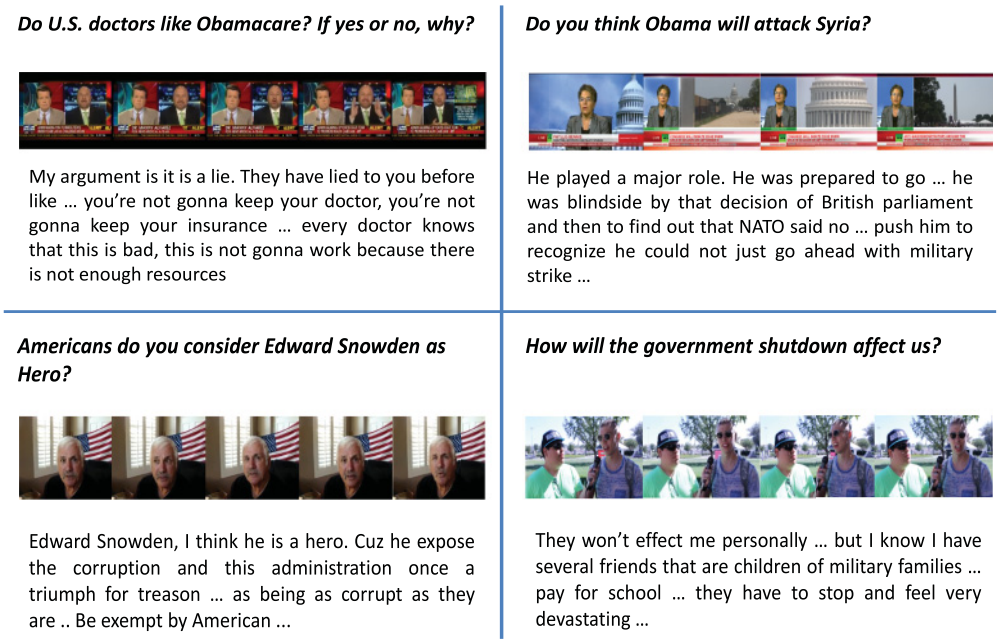

Fig. 7. Example of opinion clips retrieved by DeepHPam. The key frames and partial transcripts of the opinion clips are shown for illustration purposes.

picked by the askers were selected for testing. During the test, an evaluator was shown with a question, along with the best text answer and the top-1 video answer retrieved by our DeepHPam. Besides picking a preferred answer, an evaluator was also asked to provide a score, in the range of 1 to 3, judging how good the preferred answer is versus the other one. The score "2" ("3") means a (definitely) better answer, and "1" means that both answers are comparable and not significantly different. The scores between "2" and "3" differ in the degree of preference, while "1" could indicate a selection by chance.

The result of user studies shows that 56.7% of selections indicate that video answer is a better medium, with an average score of 1.82 for the questions for which video answers were picked. For text answers, the average score is 1.32. Among all the 96 selections, there are 22.68% (6.19%) of times for which video (text) answers were picked, with a score of 3. The result basically indicates that video answers are preferred, but not exceeding the threshold of the better or definitely better categories in most cases.

We further conducted an ANalysis Of VAriance (ANOVA) test to evaluate the level of significance. The first evaluation is about the significance of result, that is, the hypothesis that there is no difference between text and video answers. By F-test statistics, the value is 21.64 at the significance level of $p = 6.13 \times 10^{-6}$. In other words, the hypothesis is rejected, indicating that video answers are significantly better than text answers in the user study. The second evaluation tests significance of user factor, that is, the hypothesis that between-user variance is not significant. The value of the F-test statistic is 0.35, with a significance level of $p = 0.95 > 0.1$, showing that the difference among users is statistically insignificant.

The statistics from ANOVA basically support our claim that video answers are likely to be preferred for opinion-oriented questions. Here, we show one interesting example receiving a high score in the study. Referring to the question on the top-left of Figure 7,

the best text answer is "Of course not, that's why so many will retire early. They know more about it than the lib basement occupiers here." The video answer, which shows an opinion holder speaking in a dramatic tone while arguing the question from the perspective of a doctor, receives a score of 3 compared to the text answer. We observed that most video answers receiving high scores are able to provide additional information not in text answers, in addition to vivid display of emotion in expressing opinion.

7. CONCLUSIONS

We have presented our attempt in searching sentiment-oriented clips for answering opinion questions of controversial topics. Two major difficulties in building such a system, answer localization and QA matching, are tackled with the multimodal analysis of emotion content with heuristics and the nonlinear matching of speech and text with deep architecture, respectively. In general, finding a correct segment from a video collection as an answer (analogous to finding a sentence from a document collection) is a highly difficult problem, particularly when the questions of interest are about opinion expression. Our proposed solution, with the strategy of trimming the search space by considering only sentiment tracks with opinion holders as candidates for matching, is shown to work effectively and fits well for questions "born with emotions," for which there are many such examples, as posted on the social media platforms such as Yahoo! Answers. The employment of deep learning in matching text questions with video answers is also a new attempt in the literature. We are able to demonstrate encouraging results leveraging the power of nonlinear and hierarchical matching.

The current work could be extended in two major directions. Speaking tone and facial expression, which provide extra clues in sentiment analysis, could be exploited for a more reliable estimation of sentiment content. Second, this article considers only topic-wise learning of deep architectures for question answering. Future work includes learning the network in a larger scope, for instance, in the domain of "US politics," such that relearning of networks for different topics is unnecessary.

REFERENCES

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*.

Damian Borth, Rongrong Ji, Tao Chen, Thomas Breuel, and Shih-Fu Chang. 2013. Large-scale visual sentiment ontology and detectors using adjective noun pairs. In *Proceedings of the 21st ACM International Conference on Multimedia*.

Eric Brill, Jimmy Lin, Michele Banko, Susan Dumais, and Andrew Ng. 2001. Data-intensive question answering. In *Proceedings of the 10th Text REtrieval Conference (TREC)*.

Jinwei Cao and Jay F. Nunamaker. 2004. Question answering on lecture videos: A multifaceted approach. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries*.

Tao Chen, Felix X. Yu, Jiawei Chen, Yin Cui, Yan-Ying Chen, and Shih-Fu Chang. 2014. Object-based visual sentiment concept analysis and application. In *Proceedings of the ACM International Conference on Multimedia*.

Tat-Seng Chua, Richang Hong, Guangda Li, and Jinhui Tang. 2009. From text question-answering to multimedia QA on web-scale media resources. In *Proceedings of the 1st ACM Workshop on Large-scale Multimedia Retrieval and Mining*.

M. Everingham, J. Sivic, and A. Zisserman. 2006. "Hello! My name is . . . Buffy" – Automatic naming of characters in TV video. In *Proceedings of the British Machine Vision Conference*.

Ulf Hermjakob, Abdessamad Echihabi, and Daniel Marcu. 2002. Natural language based reformulation resource and web exploitation for question answering. In *Proceedings of TREC 2002*.

Gary Kacmarcik. 2005. Multi-modal question-answering: Questions without keyboards. In *Asia Federation of Natural Language Processing*.

Elie Khoury, Paul Gay, and Jean-Marc Odobez. 2013. Fusing matching and biometric similarity measures for face diarization in video. In *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval*.

Y. Lecun, L. Bottou, G. B. Orr, and K. R. Müller. 1998. Efficient backprop. In *Neural Networks: Tricks of the Trade*.

Yue-Shi Lee, Yu-Chieh Wu, and Jie-Chi Yang. 2009. BVideoQA: Online English/Chinese bilingual video question answering. *Journal of the American Society for Information Science and Technology* 509–525.

Guangda Li, Haojie Li, Zhaoyan Ming, Richang Hong, Sheng Tang, and Tat-Seng Chua. 2010. Question answering over community-contributed web videos. *IEEE MultiMedia* 46–57.

Zhengdong Lu and Hang Li. 2013. A deep architecture for matching short texts. In *Advances in Neural Information Processing Systems*.

Jana Machajdik and Allan Hanbury. 2010. Affective image classification using features inspired by psychology and art theory. In *Proceedings of the International Conference on Multimedia*.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY.

David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with Pachinko allocation. In *Proceedings of the 24th International Conference on Machine Learning*.

Liqiang Nie, Meng Wang, Zhengjun Zha, Guangda Li, and Tat-Seng Chua. 2011. Multimedia answering: Enriching text QA with media information. In *Proceedings of the 34th International ACM Conference on Research and Development in Information Retrieval*.

Dragomir R. Radev, Hong Qi, Zhiping Zheng, Sasha Blair-Goldensohn, Zhu Zhang, Weiguo Fan, and John Prager. 2001. Mining the web for answers to natural language questions. In *Proceedings of the 10th International Conference on Information and Knowledge Management*.

S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1996. Okapi at TREC-3. 109–126.

Roman Rosipal and Nicole Krmer. 2006. Overview and recent advances in partial least squares. In *Subspace, Latent Structure and Feature Selection Techniques*. 34–51.

Mickael Rouvier, Gregor Dupuy, Paul Gay, Elie Khoury, Teva Merlin, and Sylvain Meignier. 2013. An open-source state-of-the-art toolbox for broadcast news diarization. In *INTERSPEECH*.

Jianbo Shi and Carlo Tomasi. 1994. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*. 593–600.

Paul Viola and Michael J. Jones. 2004. Robust real-time face detection. *International Journal of Computer Vision* 57, 2, 137–154.

Kai Wang, Zhaoyan Ming, and Tat-Seng Chua. 2009. A syntactic tree matching approach to finding similar questions in community-based QA services. In *Proceedings of the 32nd International ACM Conference on Research and Development in Information Retrieval*.

Wei Wu, Hang Li, and Jun Xu. 2013. Learning query and document similarities from click-through bipartite graph with metadata. In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*.

Yu-Chyeh Wu, Chia Hui Chang, and Yue-Shi Lee. 2004. CLVQ: Cross-language video question/answering system. In *Proceedings of the IEEE 6th International Symposium on Multimedia Software Engineering.*

Yu-Chieh Wu and Jie-Chi Yang. 2008. A robust passage retrieval algorithm for video question answering. *IEEE Transactions on Circuits and Systems for Video Technology* 10, 1411–1421.

Hui Yang, Tat-Seng Chua, Shuguang Wang, and Chun-Keat Koh. 2003. Structured use of external knowledge for event-based open domain question answering. In *Proceedings of the 26th Annual International ACM Conference on Research and Development in Information Retrieval*.

Tom Yeh, John J. Lee, and Trevor Darrell. 2008. Photo-based question answering. In *Proceedings of the 16th ACM International Conference on Multimedia*.

Wei Zhang, Lei Pang, and Chong-Wah Ngo. 2012. Snap-and-ask: Answering multimodal question by naming visual instance. In *Proceedings of the 20th ACM International Conference on Multimedia*.

Received August 2014; revised March 2015; accepted June 2015