

Affective Audio Annotation of Public Speeches with Convolutional Clustering Neural Network

Jiahao Xu, Boyan Zhang, Zhiyong Wang, *Member, IEEE*, Yang Wang, Fang Chen, Junbin Gao, and David Dagan Feng, *Fellow, IEEE*

Abstract—Public speaking is a critical skill in daily communication. While more practicing such as rehearsal is helpful to improve such a skill, lack of personalized feedback limits the effectiveness of practicing. Therefore, we formulate the task of personalized feedback as an affective audio annotation problem by learning knowledge from online public speech videos. Considering the great success of deep learning techniques such as convolutional neural networks in a wide range of applications including speech recognition and object recognition, we propose a novel convolutional clustering neural network (CCNN) to solve this multi-label classification problem. Instead of aggregating the features of different channels through pooling, we introduce a novel clustering layer to derive intermediate representation for improved annotation performance. In order to evaluate the performance of our proposed method, we purposely built an affective audio annotation dataset by collecting more than 2,000 video clips from the TED website. Experimental results on this dataset demonstrate that our proposed method outperforms traditional CNN-based approaches with a lower hamming loss for affective annotation.

Index Terms—affective annotation, public speech, convolutional neural network, intermediate representation, clustering.

1 INTRODUCTION

PUBLIC speaking is almost an inevitable part of our daily life, from sharing experiences with friends to giving a presentation for a project in school or at the workplace. It has been an essential skill for everyone in a modern society [1]. However, public speaking is very challenging and could be the worst nightmare for many people [2]. Various books and training courses have been available to guide individuals to practice their public speaking skills, as it is widely believed that practice makes perfect. Nevertheless, the effectiveness of practicing may not be optimal without the adequate level of personalized feedback.

With the advancements in computing techniques, many studies have been conducted to help people with their public speaking skills. Batrinca *et al.* [3] introduced a system with the virtual audience for public speaking training. Similarly, Torsten [4] used the virtual audience to assess the anxiety level of speakers. And Chollet *et al.* [5] studied how virtual audience feedback can improve speaking performance. Overall, all these attempts only focus on building systems or virtual agents so that speakers can gain more realistic experiences during self-practicing.

Some other studies have been recently conducted to provide speakers feedback in various settings. Tanveer *et al.* [6] proposed a framework which can extract non-verbal behavioral cues (e.g., gestures and body movements) from

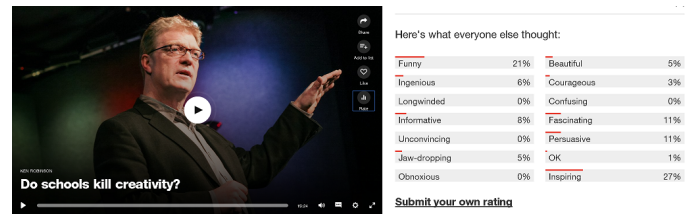


Fig. 1: A sample TED talk and its user ratings.

public speech videos without supervision. In [7], a Google Glass interface was also implemented in public speaking scenarios to provide speakers with real-time in-situ feedback, and different feedback strategies were experimented to balance effectiveness and distraction. Ali *et al.* [8] developed LISSA (Live Interactive Social Skill Assistance), an interactive conversation system, to provide live feedback to users, aiming to assist them in socializing with others. Similarly, Hoque *et al.* [9] built a coaching system, namely MACH (My Automated Conversation Coach), to help users to improve their performance in job interviews. In the system, a user will be asked to answer interview questions, and the interactions are recorded and analyzed to provide the user with visual feedback such as speaking speed. Both LISSA and MACH utilized a human-like virtual agent to simulate interpersonal communication. However, these studies mainly focus on providing basic and statistics feedback, such as speaking volume and speed, which does not directly reflect the audience affective states of a speech.

In general, a public speech is given for a purpose, such as inspiring or persuading the audience, and its effectiveness is measured in terms of the audience's affective perception [1]. Therefore, providing feedback from the audience's perspective is helpful for speakers. Meanwhile, there have been a

- J. Xu, B. Zhang, Z. Wang and D. Feng are the School of Computer Science, The University of Sydney, NSW 2006, Australia.
E-mail: {jixu7952,bzha8220}@uni.sydney.edu.au, {zhiyong.wang,dagan.feng}@sydney.edu.au.
- Y. Wang and F. Chen are with Data61, CSIRO, NSW 2015, Australia.
E-mail: {yang.wang,fang.chen}@data61.csiro.au
- J. Gao is with Discipline of Business Analytics, The University of Sydney Business School, NSW 2006, Australia.
E-mail: junbin.gao@sydney.edu.au

large number of high-quality public speech videos available online together with affective ratings from viewers, which makes it possible to train computers to predict the affective outcomes of public speeches. Therefore, we formulate the task of personalized feedback as an affective audio annotation problem by learning from online resources such as TED talks from the TED website¹ as shown in Fig. 1.

In recent years, convolutional neural network (CNN) based deep learning techniques have demonstrated significantly superior performance over traditional machine learning approaches in many applications, such as object recognition [10], image annotation [11], as well as speech recognition [12]. While these methods are able to learn discriminative features layer by layer, the learned convolutional masks mainly focus on low-level patterns such as lines and dots. As a result, a gap exists between low-level patterns and high-level semantics. The presence of the gap may not be optimal for problems where the output labels are at high semantic level, such as the affective states conveyed through audio signals of a public speech. For example, *funny* speeches often consist of a number of attributes or high-level features such as exaggerated imitation and deliberated pauses, while convoluting and pooling on low-level features may not be able to represent such information.

Meanwhile, it has been shown that mid-level features are helpful for many machine learning tasks [13], [14]. For example, Lefter *et al.* [14] used modulation as mid-level representation to help stress recognition. Both speech and gesture inputs are decomposed into semantics and modulation (e.g., speech intonation and gesture speed) as that is how humans perceive stress. However, it is unknown what types of mid-level features could benefit the affective annotation. Therefore, we propose a neural network model to derive mid-level representation from deep CNN features, and to annotate public speeches with multiple affective labels simultaneously. That is, instead of taking a two-step approach — extracting CNN features and performing traditional clustering, we devise a novel convolutional clustering neural network (CCNN), which includes a clustering layer to derive speech attributes as mid-level representation. CCNN dynamically update the cluster centroids in the training process, effectively represent the training instances and enhance the affective annotation performance. Different clustering strategies have also been investigated within the proposed network structure. We introduce discriminative clustering to better annotate the samples with more ambiguous labels. As CCNN is a general framework, other unsupervised clustering strategies (such as Self-organizing Map or Radial Basis Function) could also be easily integrated used for classification or regression tasks.

Several studies have been undertaken for different tasks by combining convolutional neural networks and clustering techniques [15]–[17]. However, our model is different from these methods in two aspects. First, instead of following fixed rules to update cluster centroids, we utilize error back-propagation through the network to learn cluster centroids. Second, our model further derives cluster based representation for characterizing inter-mediate features, which is different from those studies.

1. <http://www.ted.com/>

In summary, the key contributions of our work are as follows:

- We formulate the task of personalized feedback for public speaking as an affective audio annotation problem so that speakers can have their speeches rated with affective labels. In particular, we propose the first deep learning method for affective annotation of public speeches.
- We propose a novel network model, namely convolutional clustering neural network (CCNN), to learn mid-level features by introducing a new network layer, clustering layer, measuring the distances between convoluted features and clustering weights as mid-level representation.
- Under the proposed framework, different clustering strategies have been investigated. We propose a discriminative clustering method to mitigate the impact of ambiguity in labels during the annotation process.
- To evaluate our proposed method, we purposely built a public speeches dataset of more than 2,000 public speech videos collected from the TED website. To the best of our knowledge, this dataset is the first of its kind on public speeches with user ratings.

The rest of this paper is organized as follows. In Section 2, we review the relevant studies. In Section 3, we describe the proposed method in detail. In Section 4, we present the experimental results on the dataset we purposely built, followed by conclusions and future work in Section 5.

2 RELATED WORK

Our work is related to affective analysis of audio signals, such as emotion recognition [18] and emotion detection [19]. However, most of these methods formulate affective analysis as a single label classification which aims to exclusively categorize a given input audio signal into one of the affective states such as *Angry* and *Happy*. Meanwhile, some other studies adopt multi-label approaches, such as emotional profiles in [20] where emotional content is described by multiple probabilistic class labels and the adoption of parallel categories of valence, arousal, and dominance in [21].

Technically, our work is closely related to audio annotation [22] which aims to assign multiple labels or tags to an audio clip. By formulating audio annotation as a multi-label classification problem, many multi-label classification methods have been proposed (e.g., [23], [24]). Recently, deep learning techniques have been successfully utilized for audio annotation with promising performance. Therefore, in this section, we focus on reviewing deep learning based audio annotation methods. In terms of different types of audio signals, we organize related literature into two categories, music tagging and audio event tagging.

2.1 Music Tagging

Music tagging is to associate a piece of music with multiple musical attributes (e.g., *happy*, *rock*, *guitar*) which cover several aspects of the music (e.g., emotion, genre, and instrument). Some studies treat music tagging as a combination of multiple distinct classification tasks.

Hamel and Eck [25] first examined the feasibility of learning music audio features with deep belief networks. The learned features had been proved superior to hand-crafted features such as MFCCs (Mel-Frequency Cepstral Coefficients) and can be tailored for music tagging. In [26], Hamel *et al.* further demonstrated that combining several pooling functions can improve annotation performance. They also introduced a multi-scale learning method to combine feature learning, time pooling, and classification all together through a deep learning framework.

Based on such success, Dieleman and Schrauwen [27] proposed to use convolutional neural networks (CNN) for music tagging and achieved results comparable to spectrogram-based approaches. Their end-to-end learning framework can learn useful features from raw audio signals and discover phase invariant features with pooling layers. Meanwhile, Zhang *et al.* [28] achieved superior performance in music genre classification using CNN. They utilized both max pooling and average pooling to keep more statistical information and adopted shortcut connections between layers inspired by ResNet [29].

In addition to CNN, Choi *et al.* [30] applied fully convolutional neural networks (FCNs) to music tagging by taking Mel-spectrogram as input. However, the proposed method can only handle fixed length input. Later in [31], Choi *et al.* introduced a CNN framework integrated with a recurrent neural network (RNN) for music classification. The hybrid structure achieved better results than standalone CNN as the recurrent layers can handle the temporal information better than plain fully connected layers. The later work from the same group [32] demonstrated the feasibility of transfer learning in music tasks using CNNs with spectrogram as input. More recently, Oramas [33] proposed a deep learning based method to fuse features of multi-modal data (e.g., audio, text, and image) for music tagging on a large scale multi-modal music dataset *MuMu* (i.e., 31K albums with 250 genre classes).

However, these methods cannot be directly applied to our task. For example, the public speeches are generally much longer than music clips. Hence the deep learning models proposed for music tagging may not be able to characterize rich information about a speech thoroughly. Note that even music tagging is generally formulated as a multi-label classification task, the tags are generally from different distinct aspects (e.g., genre and instrument), while the affective attributes of a TED talk could be more relevant to each other.

2.2 Audio Event Tagging

Audio event tagging aims to identify various audio events (e.g., *Child Speech* and *Percussion*) which constitute an acoustic scene. It has attracted increasing attention from researchers due to the initiative of DCASE (Detection and Classification of Acoustic Scenes and Events) challenge.

The study by Cakir *et al.* [34] was the first attempt of applying deep learning models on audio event tagging task, which outperformed the state-of-the-art methods at that time. In their later work [35], they adopted a convolutional recurrent neural network (CRNN) to replace the conventional acoustic features such as MFCCs and achieved higher

accuracy. In [36], Kong *et al.* proposed a joint detection-classification neural network model to detect and classify an audio event simultaneously in an audio clip. Phan *et al.* [37] proposed to use CNN with different filter sizes and 1-max pooling to tag the events in audio signals.

A series of studies has been carried out by Xu *et al.* [24], [38], [39]. In [24], Xu *et al.* proposed a DNN (deep neural network) framework to address audio annotation as a multi-label classification task in a regression approach. As the data are weakly-labeled (no frame-level labels available), all frames were fed into DNN to perform a multi-label regression for expected tags. The fully connected DNN can well utilize the long-term temporary information and map sequences of acoustic features into multi-tag vectors. Besides, they also designed a deep pyramid structure to extract more robust high-level features for target tags to achieve better performance. Compared with the conventional methods, the proposed DNN method could preserve and utilize the long-term temporal information and achieve better performance. In [38], Xu *et al.* proposed a symmetric deep denoising auto-encoder to derive unsupervised features for multi-label classification. Moreover, in their recent work [39], Xu *et al.* applied an attention module and a localization module on a deep convolutional recurrent model and achieved further improvement on annotation accuracy.

While audio event tagging aims to localize the temporal position of an audio event, our affective annotation aims to identify affective state based on an audio input instead of specific audio events contained in the audio. In addition, there is little research on annotating public speeches. Therefore, we propose a novel deep architecture to derive intermediate representation for affective annotation of public speeches.

3 PROPOSED ANNOTATION METHOD

As shown in Fig. 2, our proposed annotation method takes audio signals with arbitrary lengths as input, and the output are the corresponding affective labels (e.g., *funny*, *persuasive*, *inspiring*). It consists of four key components: audio proposal selection, proposal feature learning, proposal feature clustering, and proposal feature pooling. Firstly, a set of proposals are selected from the raw audio signals and transformed into spectrogram representation. Secondly, proposal spectrograms are fed into a convolutional framework to extract convolutional features. Thirdly, a novel clustering layer is applied to the extracted mid-level representation of all the proposals. In the clustering layer, each weight vector can be seen as the location of a cluster center, while the distances between convolutional features and cluster centers will be calculated and used as the mid-level representation. Finally, average pooling and concatenation will be performed on the convoluted features and clustered features, and fully connected layers with final sigmoid activation are utilized to generate final multi-label predictions.

As mentioned above, we formulate this annotation task as a multi-label classification problem. Different from other deep multi-class classification models, our ground truth labels are not mutually exclusive, as any input audio signal could have multiple labels at the same time.

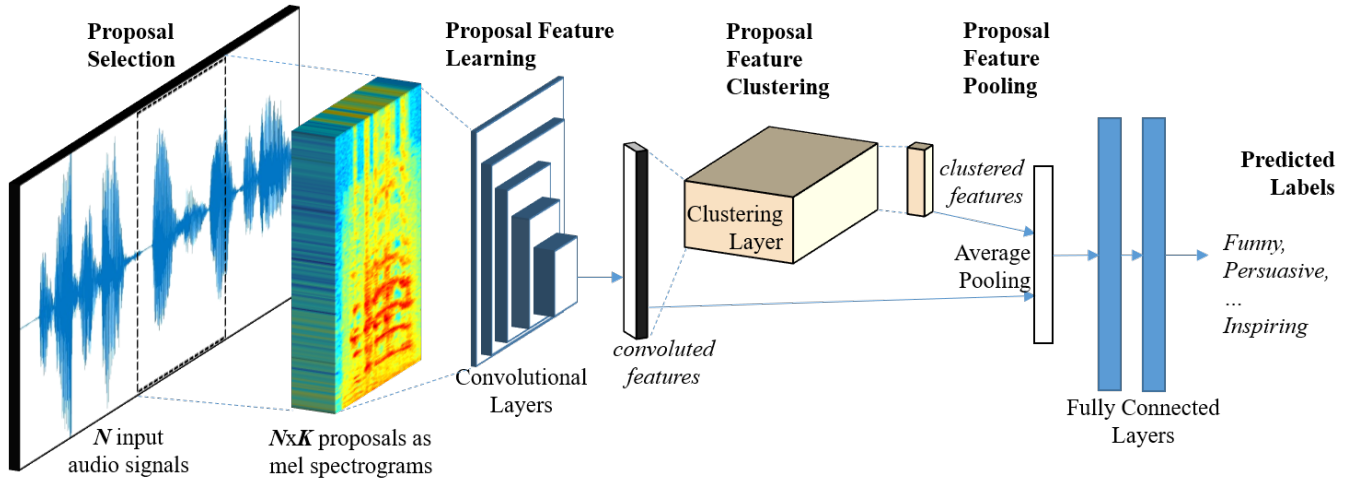


Fig. 2: Illustration of the proposed Convolutional Clustering Neural Network (CCNN) for affective audio annotation, which consists of four components (from left to right): (1) Proposal Selection, (2) Proposal Feature Learning, (3) Proposal Feature Clustering, and (4) Proposal Feature Pooling for multi-label annotation with fully connected layers.

3.1 Audio Proposal Selection

Proposals represent the highlighted segments of audio signals. While existing proposal extraction methods [40], [41] focus on image proposals for computer vision tasks, we design our proposal selection method with the following two criteria:

Representativeness: Proposals generally explain the major content of the raw data [42]. In our study, proposals should be emotional-salient segments within the whole speech. As public speeches are essentially long audio signal sequences, not all the parts of a sequence are of the same importance. Therefore, in order to better represent the whole signal, the selected proposals should be representative to describe the major emotional content.

Data Efficiency: Though deep models are always data-demanding, the size of every single training sample is expected to be small. In our model, all selected proposals from an input audio signal will be used as the representation and fed into the convolutional layers to extract deep convolutional features. As a result, a larger number of proposals will lead to exponentially higher computational cost. Therefore, we have to consider the data efficiency when selecting proposals, to ensure a low computational cost while not sacrificing the discriminative power.

tion method as illustrated in Fig. 3, by drawing inspirations from the natural language processing (NLP) field. Briefly, we use the bag-of-audio-words model to represent each input audio signal as a sequence of audio words, and select proposals whose audio words have higher TF-IDF (Term Frequency-Inverse Document Frequency) values. We finally select the top K segments as final proposals based on average TF-IDF ranking.

Firstly, we build a dictionary of audio words using K-means clustering on sliding-window frames from all the input audio signals, where MFCC features are used to represent the frames. Secondly, we split each audio signal into proposal candidates, which are adjacent fixed-length overlapping pieces. Thirdly, we represent the proposal candidates as sequences of audio words with the dictionary built earlier. In this case, we treat each proposal candidate as a document, and treat each audio word as a term, then we calculate the TF-IDF value of each audio word in the proposal candidates. Finally, we select the top K candidates with the largest average TF-IDF values, as the final proposals to represent an input audio signal. While K is a hyperparameter and we describe the setting of K in Section 4.

TF-IDF value is a good indicator of the importance of a term in a document, and has been widely used in many language processing tasks. TF-IDF value of a given term in a document can be obtained by calculating the term frequency divided by the total occurrence of the term in the whole corpus. In our study, each “Term” is an audio word extracted using 25ms sliding-window, while “Document” means the candidates extracted in the first step. Mathematically, the TF-IDF value of audio word t in segment d and entire audio signals set D could be calculated with the following equation:

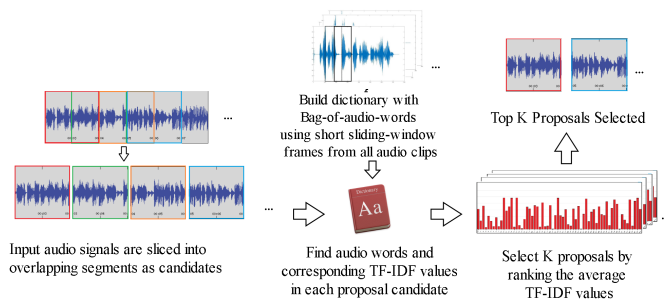


Fig. 3: Illustration of the proposal selection method.

To meet the criteria above, we propose a proposal selec-

$$TF-IDF(t, d, D) = tf(t, d) \cdot idf(t, D), \quad (1)$$

subject to:

$$\begin{aligned} \text{tf}(t, d) &= f_{t,d}, \\ \text{idf}(t, D) &= \log \frac{N}{1 + |\{d \in D : t \in d\}|}, \end{aligned} \quad (2)$$

where $f_{t,d}$ indicates the frequency count of a specific audio word t in a proposal d . N is the total number of audio signals in the training set, and $|\{d \in D : t \in d\}|$ indicates the total number of training audio which contains the audio word t .

Moreover, the nature of public speaking scenarios makes it necessary to do some preprocessing, such as removing the environmental sounds (e.g., background noise, audience chatting and applause). Therefore, we mask the input audio signals using voice activity detection (VAD) to exclude non-speech parts within audio signals before selecting the proposals with the proposed method described above.

3.2 Proposal Feature Learning

With the same number of proposals representing each input audio signal, the second stage of our whole framework is feature learning on the proposals, aiming to extract deep convolutional features for the annotation task.

Directly taking audio signals as input to deep model has been proven to have limited efficiency [43], and using spectrograms as the input of CNN has been proven successful in various audio tasks [32], [44]–[46]. Besides, the handcrafted audio features are limited in the capacity of mapping audio signals into a spatial domain, therefore cannot be used in CNN. To address this issue, we firstly convert input signals into log-Mel spectrograms. More specifically, we segment the input audio signals according to the proposals selected, and feed the spectrograms of the proposal segments as the input to the convolutional layers for feature learning.

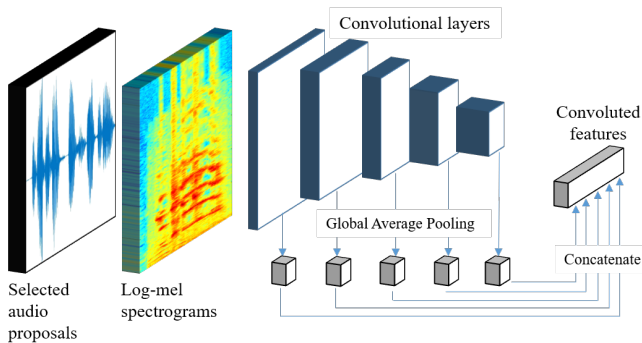


Fig. 4: Illustration of the convolutional layers used in the proposed CCNN framework.

With the spectrograms generated, we now have each input audio signal represented by a set of log-Mel spectrograms, where the number of spectrograms is equal to the number of proposals. Our proposed network uses a similar convolutional structure as in [32], where there are five convolutional layers and the first two convolutional layers have max-pooling layers attached. ReLU (Rectified Linear Unit) activation is applied to all layers to introduce the non-linearity. After each convolutional layer, a 2D global average pooling will be carried out. Finally these pooled

features are concatenated as the final deep convolutional features as shown in Fig. 4.

3.3 Proposal Feature Clustering

With the deep convolutional features, existing multi-label learning models such as [47] will directly feed those features into fully connected layers for final predictions. However, such network structure makes the transition between the convolutional layers and fully connected layers vulnerable to spatial information loss during the flattening operation (also known as vectorization). To address this problem, we design the clustering layer, to bridge the gap between low-level audio features and high-level user ratings. This clustering layer will be used to transform and prepare the deep convolutional features before feeding into fully connected layers.

3.3.1 Design of Clustering Layer

As shown in Fig. 5, high dimensional vectors are projected into 2D space for visualization. For each audio signal, we calculate the Euclidean distances between K deep convolutional features (denoted as the black dots) and I cluster centers in the clustering layer (denoted as the white stars). The mid-level representation d_{ij} denotes the Euclidean distance between the i^{th} proposal feature and the j^{th} cluster centroid.

We now describe how the clustering layer is different from conventional network layers. We first consider a fully connected layer in a network. For the deep convolutional features \mathbf{X}^{conv} , its output through a fully connected layer will be calculated as follows:

$$f(\mathbf{X}^{\text{conv}}) = \sigma(\mathbf{W}\mathbf{X}^{\text{conv}} + \mathbf{b}), \quad (3)$$

where \mathbf{W} denotes the weights of the fully connected layer, \mathbf{b} is the bias which is also known as the offset in some literature, and σ is the non-linear activation function (e.g., ReLU, sigmoid and tanh).

Now considering our proposed clustering layer, the clustered features \mathbf{X}^{cl} can be derived by calculating the distances between convolutional feature $\mathbf{X}^{\text{conv}} \in \mathbb{R}^{D \times K}$ and the layer weights $\mathbf{W} \in \mathbb{R}^{D \times I}$ as

$$\mathbf{X}^{\text{cl}} = g(\mathbf{X}^{\text{conv}}) = \|\mathbf{W} - \mathbf{X}^{\text{conv}}\|. \quad (4)$$

In our study, we adopt Euclidean distance as the metric. As a result, for the distance between proposal $\mathbf{x}_k^{\text{conv}}$ and cluster centroid \mathbf{w}_i , we can rewrite Equation (4) as

$$\mathbf{x}_{k,i}^{\text{cl}} = g(\mathbf{x}_k^{\text{conv}})_i = \sqrt{\sum_{j=1}^D (\mathbf{w}_{i,j} - \mathbf{x}_{k,j}^{\text{conv}})^2}, \quad (5)$$

where $\mathbf{x}_k^{\text{conv}}$ denotes the k^{th} proposal of a training sample and \mathbf{w}_i denotes the i^{th} cluster centroids in the clustering layer. We specify the clustering layer weights to have the same dimension D to the convolutional features. The dimensions of clustering layer output will be determined by the total number of proposals K , as well as the size of clustering layer I , $\mathbf{X}^{\text{cl}} \in \mathbb{R}^{K \times I}$. Since the calculation here is not linear, no further activation is required to introduce non-linearity.

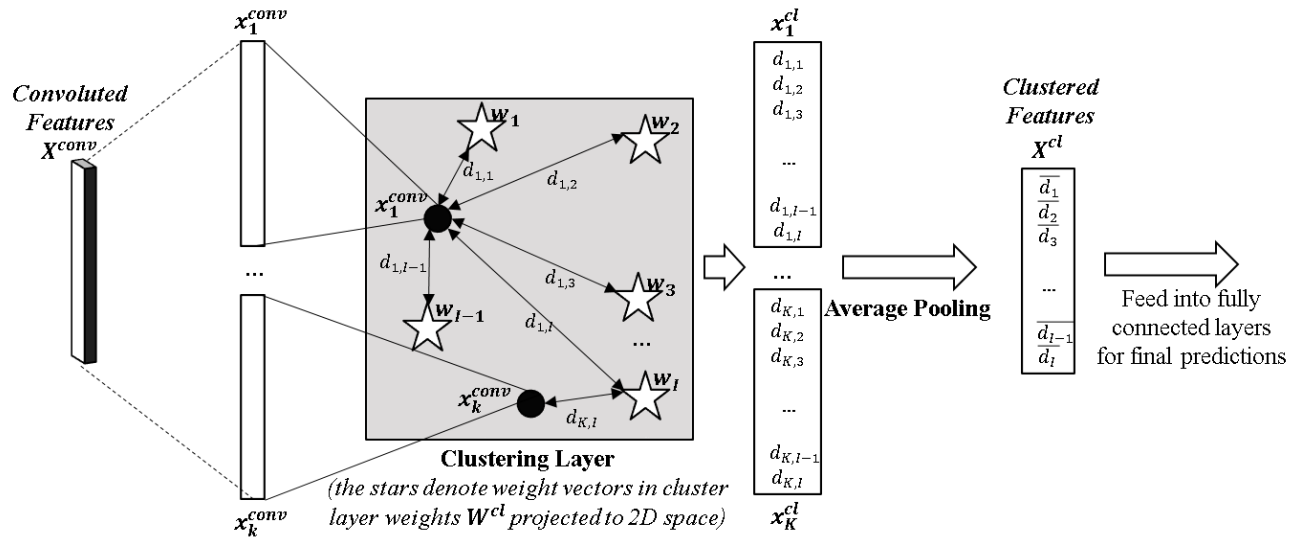


Fig. 5: Illustration of the proposal feature clustering and proposal feature pooling in the CCNN framework.

In the proposed framework, the clustering layer weights will be first initialized by clustering all training instances with K-means algorithm, and refined in the training process similar to the weights in conventional network layers with back-propagation.

3.3.2 Back-propagation Based Clustering

In neural network training, back-propagation chain rule is used to calculate the gradients of each layer in regard to the loss function, and to update the network weights correspondingly aiming to minimize the loss. Similarly, our proposed clustering layer can also be optimized with back-propagation. The derivative of the distance calculation in Equation (5) to x_k^{conv} can be calculated as

$$\frac{\partial g(x_k^{conv})}{\partial x_k^{conv}} = \frac{\mathbf{W} - x_k^{conv}}{\sqrt{\sum_{j=1}^D (\mathbf{W}_j - x_{k,j}^{conv})^2}}, \quad (6)$$

where $\mathbf{W} \in \mathbb{R}^{D \times I}$ is the weights of the clustering layer, D is the dimension of the convoluted feature x_k^{conv} , and I is the size of the clustering layer. Then the derivative of the clustering layer to convoluted features \mathbf{X}^{conv} can be defined as

$$\frac{\partial g(\mathbf{X}^{conv})}{\partial \mathbf{X}^{conv}} = \sum_{i=1}^I \frac{\sum_{k=1}^K (\mathbf{W} - x_k^{conv})}{\|\mathbf{W}_i - x_k^{conv}\|_2}. \quad (7)$$

According to the chain rule, the total model loss can be propagated to clustering layer from the fully connected layers, and then can be further back-propagated to convolutional layers as

$$\frac{\partial J(\theta)}{\partial \mathbf{X}^{conv}} = \frac{\partial J(\theta)}{\partial g(\mathbf{X}^{conv})} \frac{\partial g(\mathbf{X}^{conv})}{\partial \mathbf{X}^{conv}}, \quad (8)$$

where $J(\theta)$ is the loss function used for the proposed network. Similar to multi-class classification tasks, we adapt sigmoid cross entropy loss as the basis of our loss function.

Suppose there are N speech audio signals in the dataset labeled with L classes, and $\mathbf{y}_i = [y_{i1}, y_{i2}, \dots, y_{iL}]$ is the ground truth label of the i^{th} audio. $y_{il} = 1 (j = 1, 2, \dots, L)$ if the audio is annotated with class l , and otherwise $y_{il} = 0$.

The ground-truth probability of the i^{th} audio signal for class l is defined as $p_{il} = y_{il}$ (1 or 0), and the predicted probability of the i^{th} audio signal for class l is defined as the sigmoid activation of last layer output x^{out} , denoted as

$$\hat{p}_{il} = \hat{y}_{il} = \frac{1}{1 + e^{-x_{il}^{out}}} \quad (9)$$

Therefore, the corresponding loss function is defined as

$$J(\theta) = - \sum_{i=1}^N \sum_{j=1}^L \frac{p_{ij} \log \hat{p}_{ij} + (1 - p_{ij}) \log (1 - \hat{p}_{ij})}{N \times L}. \quad (10)$$

3.3.3 Discriminative Clustering Based Clustering

As our proposed clustering layer is very different from existing types of network layers, the conventional loss function focusing on the annotation error only cannot effectively optimize the clustering layer weights in our case. To solve this problem, we propose to introduce an extra penalty term to better supervise the clustering.

The idea of discriminative clustering is very straightforward: we want the clusters to be better at discriminating different classes by positioning them into locations with less ambiguity. Though the proposals which share the same labels are generally close to each other, there are some proposals existing in all audio signals. These inter-class evenly distributed proposals are not very helpful in our task, so during clustering we should alleviate these influences by penalizing the cluster centroid if it is moving towards locations surrounded by these ambiguous proposals.

To guide the update of the clustering layer weights to alleviate ambiguity, we introduce the discriminative clustering (DC) penalty term to restrict the proposals closer to cluster centroids having similar labels, by accumulating the total number of different labels in the neighborhood proposals divided by their distances to each cluster centroid. The discriminative clustering penalty term is defined as

$$P = \frac{\lambda}{N \times K \times L} \sum_{n=1}^N \sum_{k=1}^K \max \left(\sum_{i=1}^I \frac{\sum_{l=1}^L y_{n,l}}{\|\mathbf{w}_i - x_{i,k}^{conv}\|_2} \right), \quad (11)$$

where λ is the coefficient of the penalty term, N is the total number of input audio signals, K is the number of proposals for each audio signal, and I is the number of clusters in the clustering layer. Besides, w_i is the i^{th} cluster in the clustering layer while $x_{n,k}^{conv}$ is the convoluted feature of the k^{th} proposal in the n^{th} signal. For a given cluster in the clustering layer, if its surrounding proposals do not have many different labels then the penalty term will be smaller, and vice versa. So the new loss function is now defined as

$$J'(\theta) = J(\theta) + P, \quad (12)$$

where $J(\theta)$ and P have been defined in Equation (10) and Equation (11). With the discriminative clustering penalty term, the clusters in the clustering layer will tend to move towards locations closer to proposals with the same labels. As a result, the learned weights are more discriminative as they less influenced by the ambiguity.

3.4 Proposal Feature Pooling

After the clustering layer, each input audio signal is represented by a set of clustered features. The dimensions of these clustered features are equal to the clustering layer size and the number of proposals for each audio signal. As shown in the right-hand part of Fig. 5, we carry out average pooling across the clustered features of the proposals.

Suppose that x_i ($i = 1, 2, \dots, K$) is the clustered feature of the i^{th} proposal, and x_i^j ($j = 1, 2, \dots, I$) is the j^{th} dimension of x_i . The proposal average pooling can be formulated as

$$x^j = \frac{1}{K} \sum_{i=1}^K x_i^j, \quad (13)$$

where K is the total number of proposals selected for each audio signal. Same pooling will be done on the convoluted features, both pooled features will be concatenated as final representation, which will be fed into fully connected layers for annotation. As we formulate the annotation as a multi-label learning problem, we apply sigmoid activation over the last fully connected layer and treat each dimension in the final output as a binary indicator of the presence of a

specific label. For each class label, 1 means positive and 0 means negative.

We choose average pooling with the intuition of how users give ratings on a speech. As most human audiences rate public speeches based on the overall impression rather than one or two specific sentences, we decide to fuse these features using average pooling. By applying the pooling before the fully connected layers, the overall computation can also be reduced, leading to faster iteration.

4 EXPERIMENTAL RESULTS AND DISCUSSIONS







4.1 Dataset

To evaluate the effectiveness of our proposed method, we built a public speeches dataset including both videos and their affective ratings as collected from the TED website. TED talks are hosted by a not-for-profit organization encouraging people to share ideas worth spreading. The talks spread over various topics, ranging from life stories to cutting-edge technologies, and their speakers are from all over the world. Many of the videos have attracted millions of online viewings as shown on the TED website. Note that the videos may have been viewed on other online platforms such as Youtube. The diversity and scale of TED talks make them perfectly suitable resources for our research.

The ratings were provided by viewers to as affective responses to TED talks. Each user can rate a video with one or more of the 14 affective labels: *Confusing, Ingenious, Unconvincing, Beautiful, Informative, Inspiring, Persuasive, Funny, OK, Fascinating, Obnoxious, Courageous, Longwinded, Jawdropping*. Six sample talks are shown in TABLE 1.

In July 2017, we crawled all the talks uploaded before July 2016, as the newly uploaded videos do not have enough user ratings which could lead to biased labels. We excluded the talks with voiced segments less than half of the total length (e.g., silent show, drama, and some other performances). Some corrupted videos have also been removed from the raw dataset. The final dataset consists of 2,056 video clips with duration ranging from 3 minutes to half an hour across more than ten years (from 2006 to 2016). The

TABLE 1: Some samples of the talks in our collected TED Talk data set

Thumbnail	Talk Title	URL	Labels
	The mothers who found forgiveness, friendship	https://www.ted.com/talks/9_11_healing_the_mothers_who_found_forgiveness_friendship	<i>beautiful, courageous, inspiring</i>
	America's native prisoners of war	https://www.ted.com/talks/aaron_huey	<i>courageous, jawdropping, persuasive</i>
	Visualizing ourselves ... with crowd-sourced data	https://www.ted.com/talks/aaron_koblin	<i>beautiful, fascinating, ingenious, jawdropping</i>
	Making sense of a visible quantum object	https://www.ted.com/talks/aaron_o_connell_making_sense_of_a_visible_quantum_object	<i>confusing, fascinating, ingenious, jawdropping, unconvincing</i>
	What we learned from teetering on the fiscal cliff	https://www.ted.com/talks/adam_davidson_what_we_learned_from_teetering_on_the_fiscal_cliff	<i>informative, longwinded, obnoxious, persuasive, unconvincing</i>
	How I turned a deadly plant into a thriving business	https://www.ted.com/talks/achenyo_idachaba_how_i_turned_a_deadly_plant_into_a_thriving_business	<i>beautiful, courageous, ingenious, inspiring, ok</i>

total length of the video data is more than 500 hours. We randomly select 30% of the videos as the testing set and use the rest 70% as the training set.

As users tend to rate more on some labels such as *beautiful* and *OK*, the ratings are unbalanced across the 14 labels. For each class, we manually label 30% of the top rated video as positive samples, to alleviate the impact of inter-class unbalanced number of ratings. In the dataset, there are about 600 positive samples for each class, and the average number of labels per video is 4.08.

4.2 Experimental Settings

Our proposed model has four hyper-parameters which could impact the final annotation performance, including: the size of audio word dictionary, the time duration of each proposal, the number of proposals selected as input of CCNN and the number of clusters in the clustering layer. We carry out a series of experiments with brute-force search (exhaustive search) for the optimal combination of the aforementioned hyper-parameters. We use Keras with Tensorflow backend and set 10% validation split to tune the parameters. Experiments were conducted on a server with Intel i7 CPU, 2 NVIDIA GTX 1080Ti GPUs and 16GB memory. In this section, we describe our model with the optimal hyper-parameters combination which is also used to report the performance in Section 4.4. We further explore the impact of different parameters in Section 4.5.

We first slice input speech audio signals into 2-second segments with 50% overlap, and use these segments as the proposal candidates. For proposal selection, we build a dictionary of 3,000 audio words by applying the k-means clustering to MFCC features of each 25ms audio-words from the whole audio corpus. MFCC is used due to its popularity in a wide range of applications and its low computation requirements, though other audio features can also be used. Then we select the top 100 proposal candidates in terms of average TF-IDF value, and each proposal is represented using a log-Mel spectrogram extracted using 25ms analysis window before being fed into our network.

Due to the lack of pre-trained models for affective audio annotation, we take a hybrid approach for network training: we first pre-train the convolutional layers on binary classification tasks which classify the existence of a specific label, and then fine-tune the convolutional layers weights in proposed network afterwards for final annotation. For pre-training, we use the same convolutional layers but alter the fully connected layers to predict the existence of each class label. For final annotation, we transfer the trained weights of convolutional layers, and initialize the clustering layer by applying K-means clustering on convoluted features and use the cluster centroids as the starting weights of the clustering layer.

4.3 Evaluation Metric

Commonly used multi-label evaluation metrics include MaP (Mean Average Precision), Hamming score, Hamming loss, and AUC (Area Under Curve) of ROC (Receiver Operating Characteristic) curve. As some videos are not ranked in the top 30% in any of the 14 classes and do not have class labels, leading some of the measures could not divide by 0

errors, we use Hamming loss as our evaluation metric in our experiment instead of others.

Hamming loss can effectively evaluate how many times an input is classified incorrectly on each label for multi-label classification tasks, and is defined as an exclusive-or (XOR) operation between predicted label vector $\hat{\mathbf{y}}_t$ and ground truth label vector \mathbf{y}_t as

$$L_{Hamming}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{i=1}^N \sum_{l=1}^L \frac{(\hat{y}_i^l \oplus y_i^l)}{N \times L}, \quad (14)$$

where N is the number of input samples, L is the number of total labels which is 14 in our study, and the operator \oplus outputs the symmetric difference between \hat{y}_i and y_i as the output of XOR operation.

4.4 Experimental Results

We compare our proposed CCNN models with several relevant methods as shown in TABLE 2. The Hypotheses Cross Pooling (HCP) model (also known as multi-label CNN) described in [47] achieved state-of-the-art performance for image annotation under a setting similar to our task. A plain CNN with sigmoid loss (namely CNN-Sigmoid) and a plain CNN with Weighted Approximate-Rank Pairwise loss (namely CNN-WRAP) aim to produce confidence scores for individual labels as multi-label classifiers. CNN-SVM and HCP-Softmax perform multi-class classification on each label. For our models, CCNN-BP refer to plain back-propagation based CCNN, and CCNN-DC refer to discriminative clustering based CCNN. Both models we proposed outperform the baseline method, which indicates that the mid-level representations derived through the proposed clustering layer are more effective for high-level affective annotation than the convolutional features.

TABLE 2: Comparison of performance in terms of hamming loss, convergence speed and computational cost

Methods	With pre-trained Conv Layers		Without pre-training		Time Cost (seconds) /Epoch
	Hamming Loss	Epochs to converge	Hamming Loss	Epochs to converge	
CNN-SVM (Binary Relevance)	0.4059	-	-	-	-
HCP-Softmax	0.3960	56	-	-	-
CNN-Sigmoid	0.3729	43	-	-	42
CNN-WRAP [11]	0.3665	37	-	-	44
HCP [47]	0.3457	31	0.3506	46	70
CCNN-BP	0.3250	49	0.3197	100*	35
CCNN-DC	0.3088	27	0.3104	42	37

* Maximum number of Epochs reached in this case, the performance had not been improved in the last 10 epochs though not converged.

As shown in the second and third column of TABLE 2, there is no clear difference in annotation effectiveness between using pre-training and not using pre-training (i.e., training from scratch), although it takes longer for the latter to converge. It is also observed in the last column of TABLE 2 that our proposed methods are more computationally efficient. Such a reduction of computational cost is also owing to the CCNN architecture. Instead of connecting all deep convoluted features directly with fully connected layers, our proposed architecture retains only one pooled and concatenated feature vector for fully connected layers, which reduces the number of connections and the computation cost.

TABLE 3: T-test for the significance of improvement from our proposed methods over baseline method

Methods Compared Class Label	CCNN-BP over HCP		CCNN-DC over HCP	
	<i>t</i> -statistic	<i>p</i> -value	<i>t</i> -statistic	<i>p</i> -value
<i>beautiful</i>	-1.3116	1.90E-01	-4.1086	4.24E-05
<i>confusing</i>	-1.1641	2.45E-01	-3.6110	3.17E-04
<i>courageous</i>	-1.6705	9.51E-02	-2.3087	2.11E-02
<i>fascinating</i>	-3.3427	8.54E-04	-1.3928	1.64E-01
<i>funny</i>	-3.3361	8.75E-04	-1.0456	2.96E-01
<i>informative</i>	-1.7540	7.97E-02	-1.8624	6.28E-02
<i>ingenious</i>	-1.6379	1.02E-01	-1.4161	1.57E-01
<i>inspiring</i>	-3.4126	6.64E-04	-1.7016	8.91E-02
<i>jawdropping</i>	-4.6308	4.03E-06	-4.3907	1.23E-05
<i>longwinded</i>	-1.5728	1.16E-01	-2.4103	1.61E-02
<i>obnoxious</i>	-0.9452	3.45E-01	-2.6312	8.62E-03
<i>ok</i>	-1.6599	9.72E-02	-1.7456	8.11E-02
<i>persuasive</i>	-1.3741	1.70E-01	-1.4622	1.44E-01
<i>unconvincing</i>	-0.9652	3.35E-01	-5.1371	3.24E-07
Overall	-2.8733	4.13E-03	-2.9435	3.31E-03

In addition to the comparison on the overall annotation performance, we use the statistical *t*-test (also known as Student’s *t*-test) to demonstrate that the performance improvement over HCP is significant. The *t*-test is a widely used statistical hypothesis test to verify whether test statistic follows *t*-distribution as stated in the null hypothesis. In our case, we calculate the cross entropy loss of each class label for each predicted testing instance and compare them in pairs. So the null hypothesis here is that both samples follow a same distribution, which means the improvement is no more than random deviation. The *t*-statistic and *p*-values are shown in Table 3. We can see that for most label classes the improvement of our proposed methods is significant at 90% confidence level (with *p*-value < 0.1), and at 99% confidence level the overall performance improvement is significant (*p*-value < 0.01).

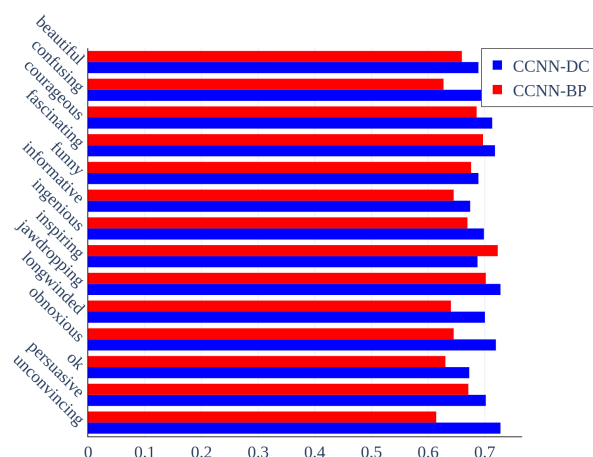


Fig. 6: Comparison between CCNN-BP and CCNN-DC in terms of label-wise precision

In order to investigate the advantage of CCNN-DC over CCNN-BP, we evaluate annotation performance for each class in terms of precision (i.e., the percentage of correctly classified samples). As shown in Fig. 6, CCNN-DC is able to improve annotation performance for all classes except *inspiring*, which confirms that the discriminative label in-

formation is helpful for producing discriminative cluster centers as well as discriminative mid-level representation.

TABLE 4: Label statistics of the dataset: the average number of labels appearing in a video annotated with a specific label

Class Label	Average Number of Labels
<i>beautiful</i>	4.2
<i>confusing</i>	5.7
<i>courageous</i>	4.1
<i>fascinating</i>	4.4
<i>funny</i>	4.5
<i>informative</i>	4.4
<i>ingenious</i>	4.3
<i>inspiring</i>	3.8
<i>jawdropping</i>	4.2
<i>longwinded</i>	5.6
<i>obnoxious</i>	5.6
<i>ok</i>	5.4
<i>persuasive</i>	4.4
<i>unconvincing</i>	5.7

For label *inspiring*, the lower precision could be due to the distribution of our data. As shown in TABLE 4, the average number of labels appearing in a video annotated with a specific label (e.g., those appearing at each row in the table) ranges from 3.8 to 5.7. A large value of a label means that the videos containing the label are generally annotated with more other labels. As *Inspiring* has the smallest value, which may not be able to provide sufficient discriminative information for the clustering process. On the contrary, for labels with larger number of co-existed labels, such as *confusing* and *unconvincing* (both 5.7), the improvement of CCNN-DC over CCNN-BP is most significant.

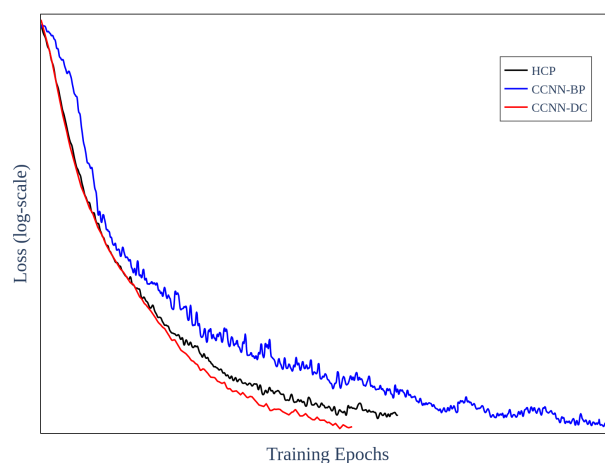


Fig. 7: Convergence curves for the training with different methods.

We further look into how these methods converge during training. All methods are trained with a commonly used strategy in deep learning, named early stopping strategy which monitors a criterion during training and stops the training process when there is no further improvement so as to avoid over-training and over-fitting. As shown in Fig. 7, all these three methods converge relatively fast, although CCNN-BP and CCNN-DC achieve lower final loss than that of the HCP method. It is also noticed that there are stronger oscillation patterns along the convergence curve

of CCNN-BP than that of CCNN-DC. This demonstrates that the discriminative clustering strategy can also reduce oscillation by imposing label information for more effective training.

4.5 Impact of Experimental Settings

As stated in Section 4.2, we investigate the impact of four different hyper-parameters of our proposed method on annotation performance. The four hyper-parameters we studied are the time duration of each proposal, the size of the dictionary established, the number of proposals selected and the size of clustering layer.

TABLE 5: Experimental results with different duration of proposals in terms of Hamming Loss

Duration of Proposals (seconds)	Hamming Loss
1	0.3651
2	0.3088
4	0.3097
8*	0.3420

* Some audio signals are not long enough to have sufficient proposals.

We first study how proposal duration could influence annotation performance in our proposed model. As shown in TABLE 5, a moderate proposal length would lead to a better performance. Setting the proposal duration too short would reduce the representative power, while setting the proposal duration too long will lead to insufficient proposals generated in our study, which also leads to a higher annotation hamming loss.

TABLE 6: Experimental results with different dictionary sizes in terms of Hamming Loss and computational time on building the dictionary

Dictionary Size	Hamming Loss	Time Cost (seconds)
200	0.4413	66
500	0.4071	314
1000	0.3568	1,175
2000	0.3207	5,823
3000	0.3088	11,979
5000	0.3122	45,781

To investigate the impact of dictionary size in proposal selection on annotation performance, we conduct various experiments with different dictionary sizes. When applying the K-means clustering algorithm on MFCC features obtained from audio-word frames to build the dictionary, the number of words in the dictionary will affect the selection of proposals and overall performance. Whereas a larger dictionary takes exponentially longer to build as shown in TABLE 6, better annotation performance can be achieved when the dictionary size increases. However, we need to balance performance improvement and computational costs. Therefore, we set the dictionary size to 3,000 in our experiments.

The impact of the number of proposals on annotation performance is also investigated. As shown in TABLE 7, in general increasing the number of proposals will improve annotation effectiveness (i.e., reducing Hamming Loss values)

with certain variations, while resulting in increased training time, as more proposals will provide more representative training samples with higher training time cost. Therefore, in our study, we select 100 proposals for each input audio signal.

TABLE 7: Experimental results with different numbers of proposals

Number of Proposals	Hamming Loss	Training Time (second)
10	0.3918	35
20	0.3674	35
30	0.3351	36
50	0.3126	36
100	0.3088	37
150*	0.3097	39
200*	0.3161	41

* Not all audio signals are long enough to have sufficient proposals.

Similarly, the impact of the number of clusters in the clustering layer, which is essentially the size of clustering layer, is investigated. As shown in TABLE 8, increasing the size of clustering layer generally improves annotation performance as well as increases training time. Therefore, in our experiments, we set the number of clusters to 4096 in the clustering layer, as further increasing clustering layer size cannot lead to significant performance improvement but requires much more computational resources.

TABLE 8: Experimental results with different clustering layer sizes

Clustering Layer Size	Hamming Loss	Training Time (seconds)
64	0.3907	30
256	0.3411	30
1,024	0.3253	31
4,096	0.3088	37
8,192*	0.3102	46
16,384*	0.3075	79

* Two extra GPUs are used as more memory is required.

5 CONCLUSION

In this paper, we present a novel deep neural network, namely Convolutional Clustering Neural Network (CCNN), for affective annotation of public speeches. Instead of directly using deep features, we introduce a clustering layer in front of fully connected layers to obtain mid-level representation as there is a gap between low-level deep features and high-level annotation labels. Such an architecture is suitable for our multi-label classification problem where affective labels do not correspond to specific audio events. We further explore different clustering strategies and investigate their impact on annotation performance. In order to evaluate our proposed method, we built the first dataset of its kind from the TED website with more than 2,000 video clips. Experimental results on this dataset demonstrate that our proposed method outperforms traditional CNN based approaches with lower Hamming loss. To the best of our knowledge, our work is one of the first studies on affective annotation of public speeches. Our future work will aim

to further improve the annotation performance by differentiating salient proposals, and exploring new clustering techniques such as fuzzy clustering [48] for interpretable clustered features.

ACKNOWLEDGEMENT

The research presented in this paper is partially supported by Data61-CISRO Postgraduate Scholarship and ARC grants.

REFERENCES

- [1] S. R. Brydon and M. D. Scott, *Between one and many: The art and science of public speaking*. McGraw-Hill, 2003.
- [2] R. F. Verderber, K. S. Verderber, and D. D. Sellnow, *The challenge of effective speaking*. Cengage Learning, 2011.
- [3] L. Batrinca, G. Stratou, A. Shapiro, L. P. Morency, and S. Scherer, "Cicero - Towards a multimodal virtual audience platform for public speaking training," in *International Workshop on Intelligent Virtual Agents*, 2013, pp. 116–128.
- [4] W. Torsten and S. Scherer, "Automatic assessment and analysis of public speaking anxiety: A virtual audience case study," in *International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2015, pp. 187–193.
- [5] M. Chollet, T. Wörtwein, L.-p. Morency, A. Shapiro, and S. Scherer, "Exploring feedback strategies to improve public speaking," in *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2015, pp. 1143–1154.
- [6] M. I. Tanveer, J. Liu, and M. E. Hoque, "Unsupervised extraction of human-interpretable nonverbal behavioral cues in a public speaking scenario," in *ACM International Conference on Multimedia (ACM-MM)*, 2015, pp. 863–866.
- [7] M. I. Tanveer, E. Lin, and M. E. Hoque, "Rhema: A real-time in-situ intelligent interface to help people with public speaking," in *ACM International Conference on Intelligent User Interfaces (IUI)*, 2015, pp. 286–295.
- [8] M. R. Ali, D. Crasta, L. Jin, A. Baretto, J. Pachter, R. D. Rogge, and M. E. Hoque, "LISSA - Live Interactive Social Skill Assistance," in *International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2015, pp. 173–179.
- [9] M. Hoque, M. Curgeon, and J. Martin, "MACH: My automated conversation coach," in *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 697–706.
- [10] A. Krizhevsky, I. Sutskever, and H. Geoffrey E., "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 1097–1105.
- [11] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe, "Deep convolutional ranking for multilabel image annotation," *arXiv preprint arXiv:1312.4894*, 2013.
- [12] P. Golik, Z. Tüske, R. Schlüter, and H. Ney, "Convolutional neural networks for acoustic modeling of raw time signal in LVCSR," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2015, pp. 26–30.
- [13] L. Feng and B. Bhanu, "Semantic concept co-occurrence patterns for image annotation and retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 785–799, 2016.
- [14] I. Lefter, G. J. Burghouts, and L. J. Rothkrantz, "Recognizing stress using semantics and modulation of speech and gestures," *IEEE Transactions on Affective Computing*, vol. 7, no. 2, pp. 162–175, 2016.
- [15] A. Coates and A. Y. Ng, "Learning feature representations with k-means," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 561–580.
- [16] A. Dundar, J. Jin, and E. Culurciello, "Convolutional clustering for unsupervised learning," *arXiv preprint arXiv:1511.06241*, 2015.
- [17] C.-C. Hsu and C.-W. Lin, "Cnn-based joint clustering and representation learning with feature drift compensation for large-scale image data," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 421–429, 2017.
- [18] Z. Aldeneh and E. M. Provost, "Using regional saliency for speech emotion recognition," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, no. 3, pp. 2741–2745, 2017.
- [19] A. Mencattini, E. Martinelli, F. Ringeval, B. Schuller, and C. Di Natlae, "Continuous estimation of emotions in speech by dynamic cooperative speaker models," *IEEE Transactions on Affective Computing*, vol. 3045, no. c, pp. 1–1, 2016.
- [20] E. Mower, M. J. Mataric, and S. Narayanan, "A framework for automatic human emotion classification using emotion profiles," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 5, pp. 1057–1070, 2011.
- [21] M. Mäntylä, B. Adams, G. Destefanis, D. Graziotin, and M. Ortu, "Mining valence, arousal, and dominance: possibilities for detecting burnout and productivity?" in *Proceedings of the 13th International Conference on Mining Software Repositories*. ACM, 2016, pp. 247–258.
- [22] H. Y. Lo, J. C. Wang, H. M. Wang, and S. D. Lin, "Cost-sensitive multi-label learning for audio tag annotation and retrieval," *IEEE Transactions on Multimedia*, vol. 13, no. 3, pp. 518–529, 2011.
- [23] C. Sanden and J. Z. Zhang, "Enhancing multi-label music genre classification through ensemble techniques," in *ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 2011, pp. 705–714.
- [24] Y. Xu, Q. Huang, W. Wang, P. J. Jackson, and M. D. Plumbley, "Fully DNN-based multi-label regression for audio tagging," *arXiv preprint arXiv:1606.07695*, 2016.
- [25] P. Hamel and D. Eck, "Learning features from music audio with deep belief networks," in *International Society for Music Information Retrieval Conference (ISMIR)*, 2010, pp. 339–344.
- [26] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck, "Temporal pooling and multiscale learning for automatic annotation and ranking of music audio," in *International Society for Music Information Retrieval Conference (ISMIR)*, 2011, pp. 729–734.
- [27] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 6964–6968.
- [28] W. Zhang, W. Lei, X. Xu, and X. Xing, "Improved music genre classification with convolutional neural networks," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 3304–3308.
- [29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
- [30] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," in *International Society for Music Information Retrieval Conference*, 2016, pp. 805–811.
- [31] K. Choi, G. Fazekas, M. Sandler, and K. Cho, "Convolutional recurrent neural networks for music classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 1–5.
- [32] —, "Transfer learning for music classification and regression tasks," *arXiv preprint arXiv:1703.09179*, 2017.
- [33] S. Oramas, O. Nieto, F. Barbieri, and X. Serra, "Multi-label music genre classification from audio, text, and images using deep features," *arXiv preprint arXiv:1707.04916*, 2017.
- [34] E. Cakir, T. Heittola, H. Huttunen, and T. Virtanen, "Polyphonic sound event detection using multi label deep neural networks," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2015, pp. 1–7.
- [35] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 25, no. 6, pp. 1291–1303, 2017.
- [36] Q. Kong, Y. Xu, W. Wang, and M. Plumbley, "A joint detection-classification model for audio tagging of weakly labelled data," *arXiv preprint arXiv:1610.01797*, 2016.
- [37] H. Phan, L. Hertel, M. Maass, and A. Mertins, "Robust audio event recognition with 1-max pooling convolutional neural networks," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 3653–3657.
- [38] Y. Xu, Q. Huang, W. Wang, P. Foster, S. Sigtia, P. J. Jackson, and M. D. Plumbley, "Unsupervised feature learning based on deep models for environmental audio tagging," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1230–1241, 2017.
- [39] Y. Xu, Q. Kong, Q. Huang, W. Wang, and M. D. Plumbley, "Attention and localization based on a deep convolutional recurrent model for weakly supervised audio tagging," in *Conference of the*

International Speech Communication Association (INTERSPEECH), 2017, pp. 3083–3087.

- [40] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3286–3293.
- [41] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision (ECCV)*, 2014, pp. 391–405.
- [42] B. Zhang, Z. Wang, D. Tao, X. S. Hua, and D. D. Feng, "Automatic preview frame selection for online videos," in *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, Nov 2015, pp. 1–6.
- [43] T. N. Sainath, R. J. Weiss, A. Senior, K. W. Wilson, and O. Vinyals, "Learning the speech front-end with raw waveform CLDNNs," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2015.
- [44] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "Cnn architectures for large-scale audio classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 131–135.
- [45] R. Hyder, S. Ghaffarzadegan, Z. Feng, J. H. Hansen, and T. Hasan, "Acoustic scene classification using a cnn-supervector system trained with auditory and spectrogram image features," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 3073–3077.
- [46] P. Yenigalla, A. Kumar, S. Tripathi, C. Singh, S. Kar, and J. Vepa, "Speech emotion recognition using spectrogram & phoneme embedding," in *Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 3688–3692.
- [47] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan, "HCP: A flexible CNN framework for multi-label image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1901–1907, 2016.
- [48] S. Zeng, Z. Wang, R. Huang, L. Chen, and D. Feng, "A study on multi-kernel intuitionistic fuzzy c-means clustering with multiple attributes," *Neurocomputing*, vol. 335, pp. 59–71, 2019.

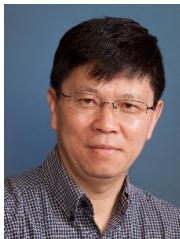


Yang Wang is a principal research scientist of Analytics Group in DATA61, CSIRO. He received his PhD in Computer Science from the National University of Singapore in 2004. His research interests include machine learning and information fusion techniques and their applications to intelligent infrastructure, cognitive and emotive computing.



Fang Chen is a senior principal research scientist of Analytics Group in DATA61, CSIRO. She holds a PhD in Signal and Information Processing, an MSc and BSc in Telecommunications and Electronic Systems respectively, and an MBA. Her research interests are behaviour analytics, machine learning, and pattern recognition in human and system performance prediction and evaluation. She has done extensive work on human-machine interaction and cognitive load modelling. She pioneered theoretical framework

of measuring cognitive load through multimodal human behaviour, and provided much of empirical evidence on using human behaviour signals, and physiological responses to measure and monitor cognitive load.



Junbin Gao graduated from Huazhong University of Science and Technology (HUST), China in 1982 with BSc. degree in Computational Mathematics and obtained PhD from Dalian University of Technology, China in 1991. He is a Professor of Big Data Analytics in the University of Sydney Business School at the University of Sydney and was a Professor in Computer Science in the School of Computing and Mathematics at Charles Sturt University, Australia. He was a senior lecturer, a lecturer in Computer Science from 2001 to 2005 at University of New England, Australia. From 1982 to 2001 he was an associate lecturer, lecturer, associate professor and professor in Department of Mathematics at HUST. His main research interests include machine learning, data analytics, Bayesian learning and inference, and image analysis.



David Dagan Feng received his M. Eng. degree in electrical engineering and computer science (EECS) from Shanghai Jiao Tong University, Shanghai, China, in 1982, and the M.Sc. degree in biocybernetics and the Ph.D. degree in computer science from the University of California, Los Angeles (UCLA), Los Angeles, CA, USA, in 1985 and 1988, respectively, where he received the Crump Prize for Excellence in Medical Engineering. He is Director of the Biomedical & Multimedia Information Technology Research Group, and Research Director of the Institute of Biomedical Engineering and Technology at the University of Sydney, Sydney, Australia. He has published over 700 scholarly research papers, pioneered several new research directions, and made a number of landmark contributions in his field. More importantly, many of his research results have been translated into solutions to real-life problems and have made tremendous improvements to the quality of life for those concerned. He has served as Chair of the International Federation of Automatic Control (IFAC) Technical Committee on Biological and Medical Systems, has organized/chaired over 100 major international conferences/symposia/workshops, and has been invited to give over 100 keynote presentations in 23 countries and regions. He is a Fellow of IEEE and Australian Academy of Technological Sciences and Engineering.

and management, Internet-based multimedia data mining, human-centred multimedia computing, and pattern recognition.



Jiahao Xu received his B. IT. in software engineering from the University of Canberra, Australia in 2012, and his M. IT. Degree in Computer Science from the University of Sydney, Australia in 2015. He is currently pursuing his Ph.D. degree in Computer Science at the University of Sydney. His research interests include affective computing, cross modality learning and machine learning.



Boyan Zhang received his Bachelor Degree in Electrical and Information Engineering from Tianjin Polytechnic University, China in 2012, his M. CS. Degree in Multimedia Information Processing from the University of Wollongong, Australia in 2013, and his M. IT. Degree from the University of Sydney, Australia in 2015. He is currently working as a research assistant at the University of Sydney. His research interests include computer vision and machine learning.



Zhiyong Wang received his B. Eng. and M. Eng. Degrees in electronic engineering from South China University of Technology, Guangzhou, China, and his Ph.D. degree from Hong Kong Polytechnic University, Hong Kong. He is currently an Associate Professor and Associate Director of the Multimedia Laboratory with the School of Information Technologies, The University of Sydney, Sydney, Australia. His research interests focus on multimedia computing, including multimedia information processing, retrieval