Correlation-Based Deep Learning for Multimedia Semantic Concept Detection

Hsin-Yu Ha(✉), Yimin Yang, Samira Pouyanfar,
Haiman Tian, and Shu-Ching Chen

School of Computing and Information Sciences,
Florida International University, Miami, FL 33199, USA
{hha001,yyang010,spouy001,htian005,chens}@cs.fiu.edu

Abstract. Nowadays, concept detection from multimedia data is considered as an emerging topic due to its applicability to various applications in both academia and industry. However, there are some inevitable challenges including the high volume and variety of multimedia data as well as its skewed distribution. To cope with these challenges, in this paper, a novel framework is proposed to integrate two correlation-based methods, Feature-Correlation Maximum Spanning Tree (FC-MST) and Negative-based Sampling (NS), with a well-known deep learning algorithm called Convolutional Neural Network (CNN). First, FC-MST is introduced to select the most relevant low-level features, which are extracted from multiple modalities, and to decide the input layer dimension of the CNN. Second, NS is adopted to improve the batch sampling in the CNN. Using NUS-WIDE image data set as a web-based application, the experimental results demonstrate the effectiveness of the proposed framework for semantic concept detection, comparing to other well-known classifiers.

Keywords: Deep learning · Feature selection · Sampling · Semantic concept detection · Web-based multimedia data

1 Introduction

In recent decades, the number of multimedia data transferred via the Internet increases rapidly in every minute. Multimedia data, which refers to data consisting of various media types like text, audio, video, as well as animation, is rich in semantics. To bridge the semantic gap between the low-level features and high-level concepts, it introduces several interesting research topics like, data representations, model fusion, imbalanced data issue, reduction of feature dimensions, etc.

Because of the explosive growth of multimedia data, the complexity rises exponentially with linearly increasing dimensions of the data, which poses a great challenge to multimedia data analysis, especially semantic concept detection. Due to this fact, it draws multimedia society's attention to identify useful feature subsets, reduce the feature dimensions, and utilize all the features extracted from

J. Wang et al. (Eds.): WISE 2015, Part II, LNCS 9419, pp. 473–487, 2015.
DOI: 10.1007/978-3-319-26187-4_43

different modalities. Many researchers develop feature selection methods based on different perspectives and methodologies. For example, whether the label information is fully explored [1–5], whether a learning algorithm is included in the method [7–11], etc. However, most feature selection methods are applied on data with one single modality. Recently, a Feature-Correlation Maximum Spanning Tree (FC-MST) [12] method has been proposed for exploring feature correlations among multiple modalities to better identify the effective feature subset.

On the other hand, the imbalanced data set is another major challenge while dealing with real world multimedia data. An imbalanced data set is defined by two classes, i.e., positive class and negative class, where the size of positive data is way smaller than the size of negative one. When training a classification model with unevenly distributed data, the model tends to classify data instances into the class with a larger data size. To resolve the issue, two types of sampling methods are widely applied, i.e., oversampling and undersampling. Oversampling methods are proposed to duplicate the positive instances to balance the data distribution. However, the computation time will increase accordingly. Undersampling methods are also widely studied to remove the negative instances to make the data set be evenly distributed. Unlike most undersampling methods, which remove the negative instances without specific criteria, Negative-based Sampling (NS) [13] is proposed to identify the negative representative instances and keeps them in the later training process.

Recently, applying deep learning methods to analyze composite data, like videos and images, has become an emerging research topic. Deep learning is a concept originally derived from artificial neural networks and it has been widely applied to model high-level abstraction from complex data. Among different deep learning methods, the Convolutional Neural Network (CNN) [14] is well established and it demonstrates the strength in many difficult tasks like audio recognition, facial expression recognition, content-based image retrieval, etc. The capability of CNN in dealing with complex data motivates us to incorporate it for multimedia analysis. Specifically, the advantages of CNN are two folds. First, CNN is composed of hierarchical layers, where the features are thoroughly trained in a bottom-up manner. Second, CNN is a biologically-derived Multi-Layer-Perceptron (MLP) [15], thus it optimizes the classification results using the gradient of a loss function with respect to all the weights in the network.

In this paper, an integrated framework is proposed to solve the semantic concept detection problem by applying two correlation-based methods, e.g., FC-MST and NS, on refining the CNN's architecture. FC-MST aims to obtain the effective features by removing other irrelevant or redundant features and it is further applied on deciding the dimension of the CNN's input layer. NS is introduced to solve the data imbalance problem and it is proposed to better refine the CNN's batch assigning process.

The rest of this paper is organized as follows. Section 2 provides related work on training the deep learning models for multimedia data analysis. A detailed description of the proposed framework is presented in Sect. 3. The experiment

dataset and the experimental results are discussed in Sect. 4. Lastly, the paper is concluded in Sect. 5 with the summarization.

2 Related Work

With the enormous growth of data such as audio, text, image, and video, multimedia semantic concept detection has become a challenging topic in current digital age [16–18]. Deep learning, a new and powerful branch of machine learning, plays a significant role in multimedia analysis [19–21], especially for the big data applications, due to its deep and complex structure utilizing a large number of hidden layers and parameters to extract high-level semantic concepts in data.

To date, various deep learning frameworks have been applied in multimedia analysis, including Caffe [22], Theano [23], Cuda-convnet [24], to name a few. Deep convolutional networks proposed by Krizhevsky et al. [25] were inspired by the traditional neural networks such as MLP. By applying a GPU implementation of a convolutional neural network on the subsets of Imagenet dataset in the ILSVRC-2010 and ILSVRC-2012 competitions [26], Krizhevsky et al. achieved the best results and reduced the top-5 test error by 10.9 % compared with the second winner. A Deep Convolutional Activation Feature (Decaf) [27], the direct precursor of Caffe, was used to extract the features from an unlabeled or inadequately labeled dataset by improving the convolutional network proposed by Krizhevsky et al. Decaf learns the features with high generalization and representation to extract the semantic information using simple linear classifiers such as Support Vector Machine (SVM) and logistic Regression (LR).

Although deep convolutional networks have attracted significant interests within multimedia and machine learning applications, generating features from scratch and the duplication of previous results are tedious tasks, which may take weeks or months. For this purpose, Caffe, a Convolutional Architecture for Fast Feature Embedding, was later proposed by Jia et al. [22], which not only includes modifiable deep learning algorithms, but also collects several pre-trained reference models. One such reference model is Region with CNN features (R-CNN) [28], which extracts features from region proposals to detect semantic concepts from very large datasets. R-CNN includes three main modules. The first module extracts category-independent regions (instead of original images) used as the inputs of the second module called feature extractor. For feature extraction and fine-tuning, a large CNN is pre-trained using the Caffe library. Finally, in the third module, the linear SVM is applied to classify the objects. Based on the evaluation results on one specific task called PASCAL VOC, CNN features carry more information compared to the conventional methods' extracted simple HOG-based features [29].

Many researchers recently utilize a pre-trained reference model to improve the results and to reduce the computational time. Snoek et al. [30] retrained a deep network, which was trained on ImageNet datasets. The input of the deep network is raw image pixels and the outputs are scores for each concept. These scores are later fused with those generated from another concept detection framework, which uses a mixture of low level features and a linear SVM for

concept detection. The overall combination framework achieves the best performance results for 9 different concepts in the Semantic Indexing (SIN) task of TRECVID 2014 [31]. Ngiam et al. [32] developed a multimodal deep learning framework for feature learning using a Restricted Boltzmann Machines (RBMs). To combine information from raw video frames with audio waveforms, a bimodal deep autoencoder is proposed, which is greedily trained by separate pre-trained models for each modality. In this model, there is a deep hidden layer, which models the relationship between audio and video modalities and learns the higher order correlation among them.

Based on the successful results acquired by deep learning techniques, an important question arises: whether deep networks are the solution for multimedia feature analysis or not. Wan et al. [33] addressed this question for Content-Based Image Retrieval (CBIR). In particular, CNN is investigated for the CBIR feature representation under the following schemes: (1) Direct feature representation using a pre-trained deep model; (2) Refining the features by similarity learning; and (3) Refining the features by model retraining using reference models such as ImageNet, which shows the promising results on the Caltec256 dataset. However, the extracted features from deep networks may not capture better semantic information compared with conventional low-level features.

More recent research in multimedia deep learning has addressed challenges such as feature extraction/selection and dimension reduction, where the input is raw pixel values. Specifically, CNN is widely used as a successful feature extractor in various multimedia tasks. However, it is still unknown how it can perform as a classifier for semantic detection tasks.

We address the aforementioned challenges by bridging the gap between semantic detection and a deep learning algorithm using general features including low-level visual and audio features as well as textual information, instead of fixed pixel values of the original images. FC-MST, a novel feature extraction method, is proposed to remove irrelevant features and automatically decide the input layer dimension. Furthermore, NS is utilized to handle the imbalanced datasets. Finally, by leveraging FC-MST and NS in the CNN structure, not only the important and relevant features are fed to the network and the data imbalance issue is solved, but also the computational time and memory usage are significantly reduced.

3 Proposed Framework

As shown in Fig. 1, the proposed framework starts from collecting the data derived from different data types, such as images, videos, and texts. Each modality requires the corresponding pre-processing step. For instance, shot boundary detection and key frame detection are applied to obtain the basic video elements, e.g., shots and keyframes, respectively. Then, low-level visual features and audio features can be extracted from them. For the image data, visual features can be directly extracted from each instance and possibly combined with the corresponding textual information including tags, title, description, etc. For the text

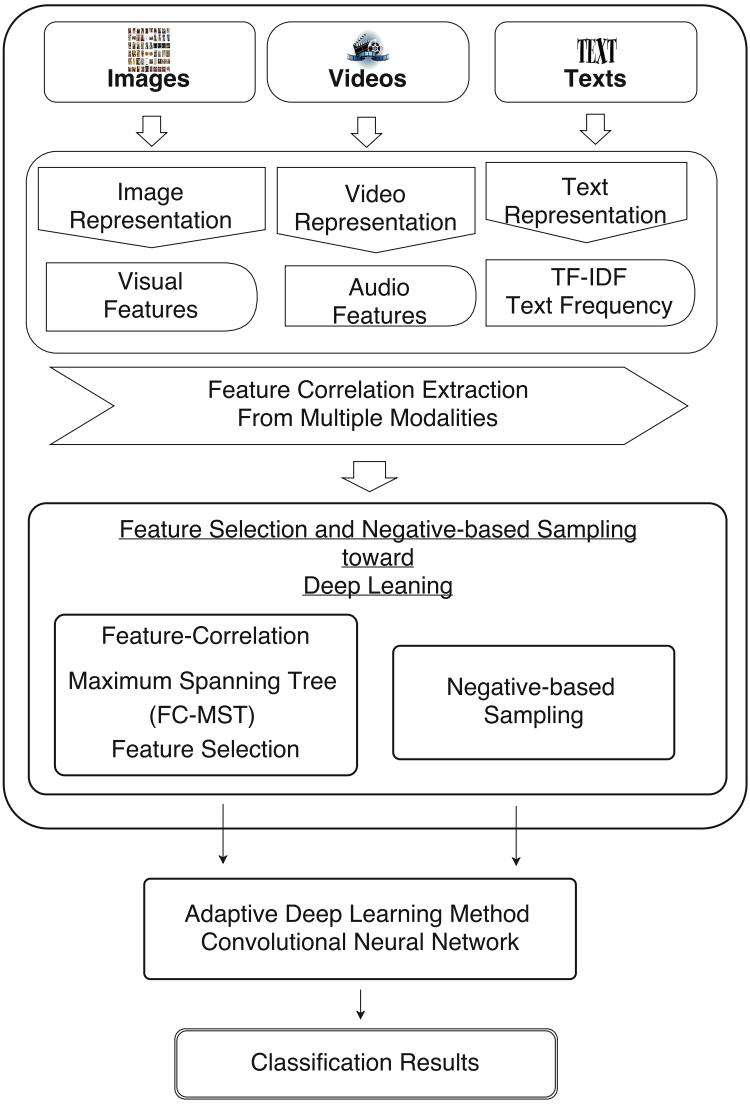

Fig. 1. Overview of the proposed framework

data, it is usually represented by its frequency or TF-IDF [34] values. Once all the features are extracted and are integrated into one, the proposed FC-MST method is adopted to select useful features and decide the dimension of the input layer. On the other hand, NS is carried out to enhance the batch instance selection for every feature map in each iteration process. Hence, the architecture of the original CNN is automatically adjusted based on the FC-MST's feature

selection and NS sampling scheme. At the end, each testing instance is labeled as 1 or 0 as an indication of a positive instance or a negative one, respectively.

3.1 Convolutional Neural Network

CNNs are hierarchical neural networks, which reduce learning complexity by sharing the weights in different layers [14]. CNN is proposed with only minimal data preprocessing requirements, and only a small portion of the original data are considered as the input of small neuron collections in the lowest layer. The obtained salient features will be tiled with an overlap to the upper layer in order to get a better representation of the observations. The realization of CNN may vary in the layers. However, basically they always consist of three types of layers: convolutional layers, pooling layers (or sub-sampling layers), and fully-connected layers. One example of the relationships between different CNN layers is illustrated in Fig. 2.

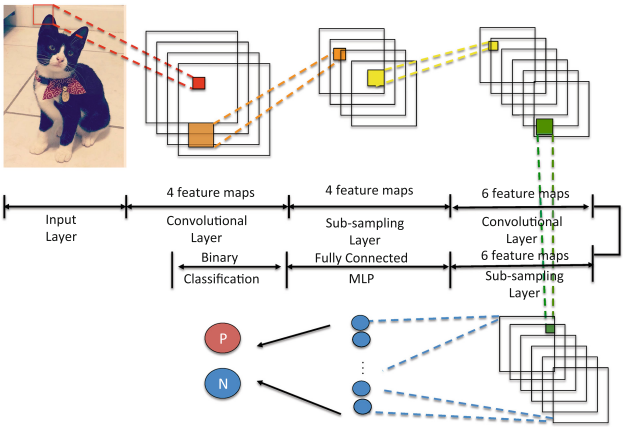

Fig. 2. Convolutional neural network

1. Convolutional layer
   There are many feature maps (representation of neurons) in each convolutional layer. Each map takes the inputs from the previous layer with the same weight W and repeatedly applies the tensor function to the entire valid region. In other words, the convolution of the previous layer's input x is fulfilled with a linear filter, where the weight for the k^{th} feature map is indicated as W^k and the corresponding bias is indicated as b_k. Then, the filtered results are applied to a non-linear activation function f. For example, if we denote the k^{th} feature map for the given layer as h^k, the feature map is obtained as follows.

$$h^k = f((W^k * x) + b_k). \tag{1}$$

The weights can be considered as the learnable kernels, which might be different in each feature map. In order to compute the pre-nonlinearity input to some unit x, the contributions from the previous layer need to be summed up and weighted by the filter components.

2. Pooling layer (Sub-sampling layer)
   Pooling layers usually come after the convolutional layers to reduce the dimensionality of the intermediate representations as shown in Fig. 2. It takes feature maps from the convolutional layer into non-overlapping blocks and subsamples them to produce a single output from each sub-region. Max-pooling is the most well-known pooling method, which takes the maximum value of each block [14,35], and it is used in the proposed framework. It is worth nothing that this type of layer does not learn by itself. The main purpose of such layer is to increase the spatial abstractness and to reduce the computation for the later layers.
3. Fully-connected MLP layer
   After several convolutional layers and pooling layers, the high-level reasoning in the neural network is done via one fully connected MLP layer. It takes all the feature maps at the previous layer as the input to be processed by a traditional MLP, which includes the hidden layer and the logistic regression process. At the end, one score is generated per instance for the classification. For a binary classification CNN model as depicted in Fig. 2, each instance is either classified as positive or negative class based on the generated score.

Convolutional neural network processes ordered data in an architecturally different way, which transparently shares the weights. This model has been shown to work well for a number of tasks, especially for object recognition [36] and it has become popular recently on multimedia data analysis [22].

3.2 FC-MST Method in Deciding Input Layer Dimension

CNN is a biologically-evolving version of MLP and it is originally implemented for tasks like MNIST digit classification or facial recognition. Though different implementations might have its own unique CNN's architecture, such as different numbers of filtering masks, sizes of the pooling layers, etc., most of them take the original image as the input and process the image as $Height \times Width$ pixel values. Here, the low-level features are selected by the proposed FC-MST and are deployed as the context of CNN's input layer.

FC-MST is proposed in [12], which aims to obtain the effective features by removing both redundant and irrelevant features. The methodology utilizes two correlations listed as follows.

- The correlation among features across multiple modalities;
- The correlation between each feature towards the target positive concept.

Given the revealed correlation, the proposed FC-MST is able to distinguish the effective features from others and greatly reduces the feature dimension. It

Algorithm 1. How to decide the dimension of CNN's input Layer by FC-MST

input : The given training data set D with feature set as $TDF = F_1, F_2, ..., F_M$, along with the class label C
output: SF: A set of selected features, which indicates the dimension of CNN's input layer $size_H$ and $size_W$

1 $ISF \longleftarrow FCMST(TDF)$;
2 **if** Num_{ISF} mod $6 = 0$ **then**
3 $\quad size_H = 6$;
4 $\quad size_W = Num_{ISF}/6$;
5 **else**
6 $\quad Num_{ISF} = Num_{ISF} - (Num_{ISF} \bmod 6)$;
$\quad$ /* Num_{ISF} represents the number of features in ISF */
7 $\quad Num_{DF} = Num_{ISF} \bmod 6$;
$\quad$ /* Num_{DF} represents the number of features which are going to be removed from ISF */
8 $\quad size_H = 6$;
9 $\quad size_W = Num_{ISF}/6$;
10 $SF \longleftarrow RemoveNumDF(ISF)$;
11 return $SF, size_H, size_W$

motivates us to apply FC-MST onto the input layer of the convolutional neural network. Hence, only the important features are considered in the process and the computation time can be greatly reduced. The process is depicted in Algorithm 1. All features from multiple modalities are combined into one unified feature set indicated as TDF. ISF represents the initial selected features after applying FC-MST on the original data set TDF (as described in Algorithm 1, line 1). Next, the number of selected features is checked on two conditions: whether it is a prime number and whether it can be divided by number 6. The checking process is described in Algorithm 1, from line 2 to line 9. The conditions are set because the dimension of the input layer needs to be completely divided by the dimension of the feature map in every convolutional layer, e.g., 2×2. Num_{DF} is obtained by getting the remainder of Num_{ISF} divided by 6. Then, Num_{DF} features are removed based on their correlation towards the positive concept and the deletion operation is performed on the least correlated features (as described in Algorithm 1, line 10). At the end, the selected feature set SF along with the decided dimension of the input layer, e.g., $size_H$ and $size_W$, are returned.

3.3 Negative-Based Sampling in Deciding Batch Sampling Process

The data imbalance problem has been one of the major challenges when classifying a multimedia data set. When the data size of the major class is way larger than that of the minor's, it usually results in poor classification performance. The problem becomes worse when applying the deep learning methods, such as CNN, on the skewed data set. The reason is because most of the deep learning

Algorithm 2. Negative-based CNN batch sampling process

input : The given training data set D is composed of positive set P and negative set N.

1 **while** *Iterating in Pooling Layer or Convolutional Layer* **do**
2 $Num_P \longleftarrow |P|$;
3 $Num_N \longleftarrow |N|$;
4 $Num_D \longleftarrow |D|$;
5 $BatchSize = Num_D/100$;
6 $NF \longleftarrow FCMST(D)$;
7 **for** *all training negative instances* $I_i, i = 1, ..., Num_N$ **do**
8 $NegRank(I_i) = MCA_{NF}(I_i)$;
9 **for** *Each batch* $B_j, j = 1, ..., 100$ **do**
10 $B_j \longleftarrow \emptyset$;
11 **if** $Num_P > 1/2BatchSize$ **then**
12 $B_j \longleftarrow$ randomly pick $1/2BatchSize$ from P;
13 **else**
14 $B_j \longleftarrow P$;
15 $BP_j \longleftarrow |B_j|$;
16 $BN_j \longleftarrow (BatchSize - BP_j)$;
17 $B_j \longleftarrow$ select BN_j instances with higher Negative Ranking Score from the first $j^{th}BatchSize$ of instances;
18 Continuing in training CNN model;

methods, including CNN, start the training process by assigning instances into different batches and each batch might contain no positive instance but all negative instances due to this uneven distribution. Assigning random instances into each batch is not able to resolve the data imbalance problem and it could result in poor classification results.

To tackle this challenge, "the NS method", which is published in [13], is adopted to improve the CNN batch sampling process as shown in Algorithm 2. As long as the training process is still within either the pooling or convolutional layer, the same negative-based CNN batch sampling process is applied (as described in Algorithm 2, line 1). At the beginning, the number of positive set, negative set, and the combined data set, are obtained and represented as Num_P, Num_N, and Num_D, respectively. The number of instances in each batch is set to be 1/100 of the total number of instances Num_D. A set of features NF are selected based on the negative-based FC-MST method, which are highly correlated with the target negative concept (as described in Algorithm 2, line 2–6). All the negative instances are looped through to generate the corresponding negative-based ranking score. The negative ranking score is generated by the method called Multiple Correspondence Analysis (MCA) [37,38] using the above-selected features NF. The higher the score is, the more negative-representative the instance is (as described in Algorithm 2, line 7–8). For each batch, it starts with an empty set and then is assigned with either the whole positive set P or the half batch size of the positive instances (as described in

Algorithm 2, line 9–17). The last step in this batch sampling process is to obtain the subtraction of $BatchSize$ and the current numbers of the assigned positive and negative instances are denoted as BP_j and BN_j, respectively. From the $j^{th} BatchSize$ number of instances, the first BN_j instances with higher negative ranking scores are selected into batch B_j. The same process is applied and looped through all the batches.

4 Experiment

4.1 NUS-WIDE Dataset

The proposed framework is validated using the well-known multimedia data set called NUS-WIDE [39]. It is a web image data set downloaded from Flickr website including six types of low-level features. The lite version, which contains 27,807 training images and 27,808 testing images, is conducted in this experiment. The data set contains relatively low Positive to Negative Ratios for all 81 concepts, which is depicted in Fig. 3.

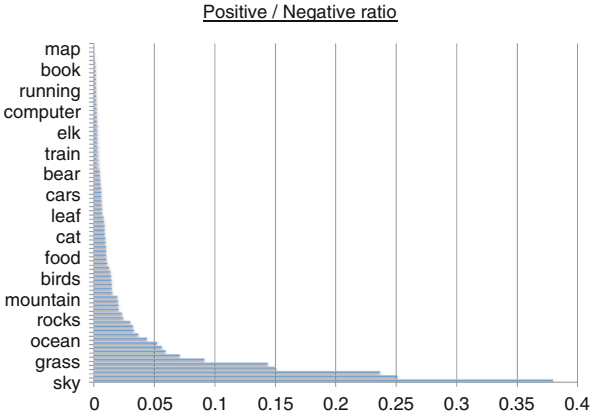

Fig. 3. Positive and Negative Ratios of NUSWIDE Lite 81 concepts

4.2 Experiment Setup and Evaluation

The proposed framework is compared with two well-known classifiers, e.g., K-Nearest Neighbors (KNN) and SVM. It is also compared to MCA-TR-ARC [40], which is applied on the NUSWIDE data set to remove the noisy tags and combine the ranking scores from both tag-based and content-based models. In addition, a sensitivity analysis is conducted to justify which component contributes the most in enhancing the classification results.

,

Table 1. Average Precision (AP) of the proposed method and other classifiers

Method	Average Precision (AP)
KNN	9.87 %
SVM	11.23 %
CNN	10.41 %
MCA-TR-ARC	33 %
Proposed Method	35.61 %

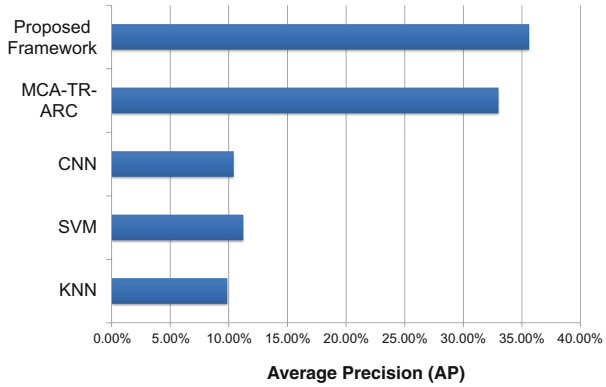

Fig. 4. Average Precision comparing with other methods

4.3 Results

The Average Precision (AP) of NUS-WIDE's 81 concepts for 4 different classifiers and the proposed framework is shown in Table 1. KNN performs the worst with an AP value of 9.87 %, which shows that a huge amount of unselected features and the data imbalance issue actually result in very poor classification performance. The same issue affects both SVM and CNN. SVM produces an AP value of 11.24 %, which is 1.37 % higher when compared to KNN, because it is able to better separate the positive instances from the negative ones. With regard to CNN, it is not able to reach a better performance because how it assigns instances into batches does not resolve the data imbalanced issue. However, CNN has the ability of iterating the training process until it reaches the optimal results, and thus it is able to obtain slightly higher AP values against KNN. MCA-TR-ARC produces a relatively much higher AP value compared to others because of two reasons. First, it applies MCA to remove the noisy tag information. Second, it explores the correlation between the tag-based model and the content-based model, and fuses the ranking scores into one. Finally, the proposed framework, which combines two correlation-based methods, can

Table 2. Sensitivity Analysis (SA) in evaluating contribution for each component

Method	Average Precision	Dropped Performance
The Proposed Work	35.61 %	—
Remove FC-MST	23.85 %	11.76 %
Remove NS	19.39 %	16.22 %
Remove Both	10.41 %	25.20 %

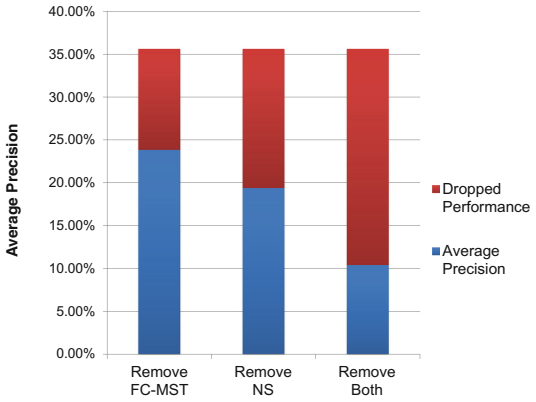

Fig. 5. Sensitivity analysis on the proposed work (Color figure online)

outperform all the other classifiers in the NUS-WIDE dataset. Figure 4 also visually depicts the aforementioned classification results.

A sensitivity analysis is further performed to better analyze the contribution for each component. In Table 2, the first column is the AP values performed by the proposed framework, which includes both FC-MST and NS, and it is able to reach 35.61 %. If FC-MST is removed from the proposed framework, then the AP value dropped by 11.76 %. On the other hand, if NS is removed from the proposed framework, the performance dropped even more. The results indicate that identifying useful features can efficiently increase the average precision, but better assigning the instances into each training batch plays a much important role. Figure 5 highlights the dropped performance in color red when removing different components. The rightmost bar, which is indicated as "Remove Both", represents the performance of the original CNN.

5 Conclusion

In this paper, an integrated framework is proposed to adopt two correlation-based methods, e.g., FC-MST and NS, in adjusting the architecture of one well-known deep learning method called CNN. First, FC-MST is proposed to identify effective features and decide the dimension of CNN's input layer instead of using

fixed pixel values of the original images. The features are selected based on their correlation towards the target positive class. Second, NS is proposed specifically to cope with the imbalanced data sets, which usually results in poor classification performance due to its uneven distribution. The problem is worse when the original CNN randomly assigns data instances into each batch. Thus, NS is adopted to alleviate the problem. The experiment shows this proposed integrated framework is able to outperform other well-known classifiers and each correlation-based method can independently contribute to enhance the results.

Acknowledgment. This research was supported in part by the U.S. Department of Homeland Security under grant Award Number 2010-ST-062-000039, the U.S. Department of Homeland Security's VACCINE Center under Award Number 2009-ST-061-CI0001, NSF HRD-0833093, CNS-1126619, and CNS-1461926.

References

1. Zhu, Q., et al.: Feature selection using correlation and reliability based scoring metric for video semantic detection. In: 2010 IEEE Fourth International Conference on Semantic Computing (ICSC) (2010)
2. Shyu, M.-L., et al.: Network intrusion detection through adaptive sub-eigenspace modeling in multiagent systems. ACM Trans. Auton. Adapt. Syst. (TAAS) **2**(3), 9 (2007)
3. Shyu, M.-L., et al.: Image database retrieval utilizing affinity relationships. In: Proceedings of the 1st ACM International Workshop on Multimedia Databases (2003)
4. Shyu, M.-L., et al.: Mining user access behavior on the WWW. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, pp. 1717–1722 (2001)
5. Shyu, M.-L., et al.: Generalized affinity-based association rule mining for multimedia database queries. Knowl. Inf. Syst. (KAIS) **3**, 319–337 (2001)
6. Ha, H.-Y., et al.: Content-based multimedia retrieval using feature correlation clustering and fusion. Int. J. Multimedia Data Eng. Manage. (IJMDEM) **4**(5), 46–64 (2013)
7. Li, X., et al.: An effective content-based visual image retrieval system. In: Proceedings of the 26th IEEE Computer Society International Computer Software and Applications Conference (COMPSAC) (2002)
8. Huang, X., et al.: User concept pattern discovery using relevance feedback and multiple instance learning for content-based image retrieval. In: Proceedings of the Third International Workshop on Multimedia Data Mining (MDM/KDD), in conjunction with the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2002)
9. Chen, S.-C., et al.: Augmented transition networks as video browsing models for multimedia databases and multimedia information systems. In: Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), pp. 175–182 (1999)
10. Chen, S.-C., et al.: Identifying overlapped objects for video indexing and modeling in multimedia database systems. Int. J. Artif. Intell. Tools **10**(4), 715–734 (2001)

11. Chen, X., et al.: A latent semantic indexing based method for solving multiple instance learning problem in region-based image retrieval. In: Proceedings of the IEEE International Symposium on Multimedia (ISM), pp. 37–44 (2005)
12. Ha, H.-Y., Chen, S.-C., Chen, M.: FC-MST: feature correlation maximum spanning tree for multimedia concept classification. In: IEEE International Conference on Semantic Computing (ICSC) (2015)
13. Ha, H.-Y., Chen, S.-C., Shyu, M.-L.: Negative-based sampling for multimedia retrieval. In: The 16th IEEE International Conference on Information Reuse and Integration (IRI) (2015)
14. LeCun, Y., et al.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
15. Ruck, D.W., et al.: The multilayer perceptron as an approximation to a Bayes optimal discriminant function. IEEE Trans. Neural Netw. **1**(4), 296–298 (1990)
16. Yang, J., Yan, R., Hauptmann, A.G.: Cross-domain video concept detection using adaptive svms. In: Proceedings of the 15th ACM International Conference on Multimedia (2007)
17. Meng, T., Shyu, M.-L.: Leveraging concept association network for multimedia rare concept mining and retrieval. In: IEEE International Conference on Multimedia and Expo (ICME) (2012)
18. Ballan, L., et al.: Event detection and recognition for semantic annotation of video. Multimedia Tools Appl. **51**(1), 279–302 (2011)
19. Mobahi, H., Collobert, R., Weston, J.: Deep learning from temporal coherence in video. In: Proceedings of the 26th ACM Annual International Conference on Machine Learning (2009)
20. Zou, W., et al.: Deep learning of invariant features via simulated fixations in video. In: Advances in Neural Information Processing Systems (2012)
21. Yang, Y., Shah, M.: Complex events detection using data-driven concepts. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part III. LNCS, vol. 7574, pp. 722–735. Springer, Heidelberg (2012)
22. Jia, Y., et al.: Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the ACM International Conference on Multimedia (2014)
23. Bastien, F., et al.: Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590 (2012)
24. Krizhevsky, A.: Cuda-convnet (2012). https://code.google.com/p/cuda-convnet/
25. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (2012)
26. Berg, A., Deng, J., Fei-Fei, L.: Large scale visual recognition challenge 2010 (2010). www.imagenet.org/challenges
27. Donahue, J., et al.: Decaf: a deep convolutional activation feature for generic visual recognition. arXiv preprint arXiv:1310.1531 (2013)
28. Girshick, R., et al.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
29. Felzenszwalb, P.F., et al.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
30. Snoek, C.G.M., et al.: MediaMill at TRECVID 2013: searching concepts, objects, instances and events in video. In: NIST TRECVID Workshop (2013)
31. Over, P., et al.: TRECVID 2010: an overview of the goals, tasks, data, evaluation mechanisms, and metrics (2011)

32. Ngiam, J., et al.: Multimodal deep learning. In: Proceedings of the 28th International Conference on Machine Learning (ICML) (2011)
33. Wan, J., et al.: Deep learning for content-based image retrieval: a comprehensive study. In: Proceedings of the ACM International Conference on Multimedia (2014)
34. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manage. **24**(5), 513–523 (1988)
35. Serre, T., et al.: Robust object recognition with cortex-like mechanisms. IEEE Trans. Pattern Anal. Mach. Intell. **29**(3), 411–426 (2007)
36. McCann, S., Reesman, J.: Object detection using convolutional neural networks
37. Lin, L., et al.: Weighted subspace filtering and ranking algorithms for video concept retrieval. IEEE MultiMedia **18**(3), 32–43 (2011)
38. Yang, Y., Chen, S.-C., Shyu, M.-L.: Temporal multiple correspondence analysis for big data mining in soccer videos. In: The First IEEE International Conference on Multimedia Big Data (BigMM) (2015)
39. Chua, T.-S., et al.: NUS-WIDE: a real-world web image database from National University of Singapore. In: Proceedings of the ACM International Conference on Image and Video Retrieval (2009)
40. Chen, C., et al.: Web media semantic concept retrieval via tag removal and model fusion. ACM Trans. Intell. Syst. Technol. (TIST) **4**(4), 61 (2013)