RUC-Tencent at ImageCLEF 2015: Concept Detection, Localization and Sentence Generation

Xirong Li*[1], Qin Jin*[1], Shuai Liao[1], Junwei Liang[1], Xixi He[1], Yujia Huo[1], Weiyu Lan[1], Bin Xiao[2], Yanxiong Lu[2], Jieping Xu[1]

[1]Multimedia Computing Lab, School of Information, Renmin University of China
[2]Pattern Recognition Center, WeChat Technical Architecture Department, Tencent
{xirong,qjin}@ruc.edu.cn

Abstract. In this paper we summarize our experiments in the ImageCLEF 2015 Scalable Concept Image Annotation challenge. The RUC-Tencent team participated in all subtasks: concept detection and localization, and image sentence generation. For concept detection, we experiments with automated approaches to gather high-quality training examples from the Web, in particular, visual disambiguation by Hierarchical Semantic Embedding. Per concept, an ensemble of linear SVMs is trained by Negative Bootstrap, with CNN features as image representation. Concept localization is achieved by classifying object proposals generated by Selective Search. For the sentence generation task, we adopt Google's LSTM-RNN model, train it on the MSCOCO dataset, and fine-tune it on the ImageCLEF 2015 development dataset. We further develop a sentence re-ranking strategy based on the concept detection information from the first task. Overall, our system is ranked the 3rd for concept detection and localization, and is the best for image sentence generation in both clean and noisy tracks.

Keywords: Concept detection, Concept localization, Image Captioning, Deep Learning, Hierarchical Semantic Embedding, Negative Bootstrap, CNN, LSTM-RNN, Sentence Reranking

1 Introduction

This year we participated in all the subtasks, i.e., *concept detection and localization* and *image sentence generation*, in the ImageCLEF 2015 Scalable Concept Image Annotation challenge. In addition to the 500k web image set (webupv2015 hereafter) and the 2k develop set (Dev2k) provided by the organizers [1], we leverage a number of external resources, listed below:

- Task 1: WordNet, ImageNet [2], Flickr images, a pretrained Caffe CNN [3], and a pretrained HierSE model [4].

* X. Li and Q. Jin contributed equally to this work.

– Task 2: MSCOCO [5], a pretrained VGGNet CNN [6], and a pretrained word2vec model [7].

Next, we introduce in Section 2 our concept detection and localization system, followed by our image sentence generation system in Section 3.

2 Task 1: Concept Detection and Localization

We develop a concept detection and localization system that learns concept classifiers from *image-level* annotations and localizes concepts within an image by *region-level* classification, as illustrated in Fig. 1. Compared with our earlier systems [8,9], this year we make two technical improvements for concept modeling. First, we replace bag of visual words features by an off-the-shelf CNN feature, i.e., the last fully connected layer (fc7) of Caffe [3]. Second, we employ Hierarchical Semantic Embedding [4] for concept disambiguation, which is found to be effective for acquiring positive examples for some ambiguous concepts such as 'mouse', 'basin', and 'blackberry'.

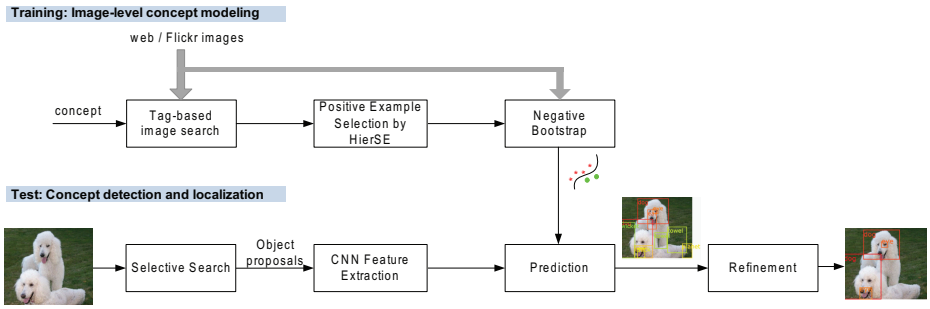


Fig. 1. An illustration of the RUC-Tencent concept detection and localization system.

2.1 Concept Modeling

Positive Training Examples Since hand labeled data is allowed this year, we collect positive training examples from multiple sources of data, including

1. ImageNet. For 208 of the 251 ImageCLEF concepts, we can find labeled examples from ImageNet [2].
2. webupv2015. It consists of 500K web images provided by the organizers [1]
3. flickr2m. We start with the 1.2 million Flickr image set from [10], and extend it by adding more Flickr images labeled with the ImageCLEF concepts, resulting in a set of two million images.

Since the annotations of webupv2015 and flickr2m are noisy, per concept we collect its positive training examples by a two-step procedure. In the first step, we conduct tag-based image search using tags of the concept as the query to generate a set of candidate images. For webupv2015, an image is chosen as a candidate as long as its meta data, e.g., url and query logs, overlap the tags. In the second step, these candidate images are sorted in descending order in terms of their relevance to the concept, and the top-n ranked images are preserved as the positive training set. To compute the relevance score between the given concept and a specific image, we embed them into a common semantic space by the Hierarchical Semantic Embedding (HierSE) algorithm [4]. Consequently, the cross-media relevance score is computed as the cosine similarity between the embedding vectors.

Since HierSE takes the WordNet hierarchy into account, it can resolve semantic ambiguity by embedding a label into distinct vectors, depending on its position in WordNet. Consider the label 'basin' for instance. In the context of ImageCLEF, it is 'basin.n.01', referring to a bow-shaped vessel. On the other hand, it can also be 'basin.n.03', meaning a natural depression in the surface of the land. In HierSE, a label with a given sense is represented by a convex combination of the embedding vectors of this label and its ancestors tracing back to the root. Fig. 2 shows the top-ranked images of 'basin' from flickr2m, returned by the tag relevance (tagrel) algorithm [11] and HierSE, respectively.

Given the HierSE ranked images per concept, we empirically preserve the top $n = 1000$ images as positive training examples. Since the number of genuine positives varies over concepts, this year we also consider an adaptive selection strategy. We train an SVM model using the top 20 images as positives and 20 images at the bottom as negatives. The previously ranked images are classified by the model and labeled as positive if the classification scores exceed 0. We denote this strategy as HierSE + SVM.

Negative Bootstrap To effectively exploit an overwhelming number of (pseudo) negative examples, we learn an ensemble of linear SVMs by the Negative Bootstrap algorithm [12]. The base classifiers are trained using LIBLINEAR [13]. The classifier ensemble is compressed into a single model using the technique developed in [14] so that the prediction time complexity is independent of the ensemble size, and linear w.r.t. the visual feature dimension.

To evaluate the multiple methods for selecting positive and negative examples, we split the ImageNet set into two disjoint subsets, imagenet-clef15train and imagenet-clef15test, which contain positive examples for the 208 ImageCLEF concepts. Using imagenet-clef15test as the test set, the performance of the individual methods is summarized in Table 1. For webupv2015, the combination of HierSE + SVM and Negative Bootstrap performs the best. For flickr2m, the combination of HierSE and Negative Bootstrap performs the best. Still, there is a substantial gap between the model trained on auto-selected examples (MAP 0.375) and the model trained on hand labeled examples (MAP 0.567). So for the 208 concepts, we combine all the models trained on imagenet-clef15train,

(a) basin by tagrel

(b) basin by HierSE

Fig. 2. Images of 'basin' retrieved by (a) tagrel and (b) HierSE, respectively. The HierSE results are more relevant to the ImageCLEF concept, i.e., a bow-shaped vessel.

Table 1. Evaluating methods for positive and negative training example selection. Test set: imagenet-clef15test. Methods are sorted in descending order by MAP.

Data source	Positive examples	Negative examples	MAP
flickr2m	HierSE	Negative Bootstrap	0.375
flickr2m	tagrel	Negative Bootstrap	0.362
flickr2m	HierSE	random sampling	0.356
flickr2m	HierSE + SVM	Negative Bootstrap	0.350
webupv2015	HierSE + SVM	Negative Bootstrap	0.338
flickr2m	HierSE + SVM	random sampling	0.337
webupv2015	HierSE	Negative Bootstrap	0.333
webupv2015	Hierse + SVM	random sampling	0.314
webupv2015	Hierse	random sampling	0.278
imagenet-clef15train	Manual	Negative Bootstrap	0.567
imagenet-clef15train	Manual	random sampling	0.547

webupv2015, and flickr2m, while for the other 43 concepts, we combine all the models trained on webupv2015 and flickr2m. Although learned weights are found to be helpful according to our previous experiences [8, 15], we use model averaging in this evaluation due to time constraints.

2.2 Concept Localization

Object Proposal Generation For each of the 500k images, we use Selective Search [16] to get a number of bounding-box object proposals. In particular, the image is first over-segmented by the graph-based image segmentation algorithm

[17]. The segmented regions are iteratively merged by hierarchical grouping as depicted in [16]. As the CNN feature is extracted per bounding box, the number of chosen object proposals is set to 20 given our computational power.

Detection Given an image and its 20 object proposals, we classify each of them using the 251 concept models, and label it with the concept of maximum response. Notice that for each concept, we have two choices to implement its model. One choice is HierSE, as it was developed for zero-shot learning, and is thus directly applicable to compute cross-media relevance scores. The other choice is linear SVMs trained from the selective positives combined with Negative Bootstrap. HierSE is the same as used for positive example selection, while the SVMs used here are distinct from the SVMs in Section 2.1.

To reduce false alarms, we refine the detection as follows. For object proposals labeled as the same concept, if their number is lower than a given Minimum Detection threshold md (we tried 1, 2, and 3), they are discarded, otherwise we sort them in descending order in terms of detection scores. We go through the ranked proposals, preserving a proposal if its bounding box has less than 30% overlap with the previously preserved proposals.

2.3 Submitted Runs

We submitted eight runs in the concept detection and localization task.

ruc_task1_hierse_md1 is our baseline run, directly using HierSE to compute the relevance score between an object proposal and a concept, with the Minimum Detection threshold md set to 1, namely no removal.

ruc_task1_hierse_md2 is the same as ruc_task1_hierse_md1, but with $md = 2$.

ruc_task1_hierse_md3 is the same as ruc_task1_hierse_md1, but with $md = 3$.

ruc_task1_svm_md1 uses SVMs for concept detection, with $md = 1$.

ruc_task1_svm_md2 is the same as ruc_task1_svm_md1, but with $md = 2$.

ruc_task1_svm_md3 is the same as ruc_task1_svm_md1, but with $md = 3$.

ruc_task1_svm_md2_nostar is the same as ruc_task1_svm_md2, but empirically thresholding the detection results of two concepts 'planet' and 'star', as they tend to be over fired.

ruc_task1_svm_md3_nostar is the same as ruc_task1_svm_md3, but empirically thresholding the detection results of two concepts 'planet' and 'star'.

Result Analysis The performance of the eight runs is shown in Fig. 3. We attribute the low performance of the HierSE runs to the following two reasons. First, HierSE is designed for zero-shot learning. It does not use any examples of the ImageCLEF concepts. Second, the current implementation of HierSE relies on the ImageNet 1k label set [18] to embed an image, while the 1k set is loosely related with the ImageCLEF concepts. We can see that the Minimum Detection threshold is helpful, improving MAP_0.5Overlap from 0.452 to 0.496.

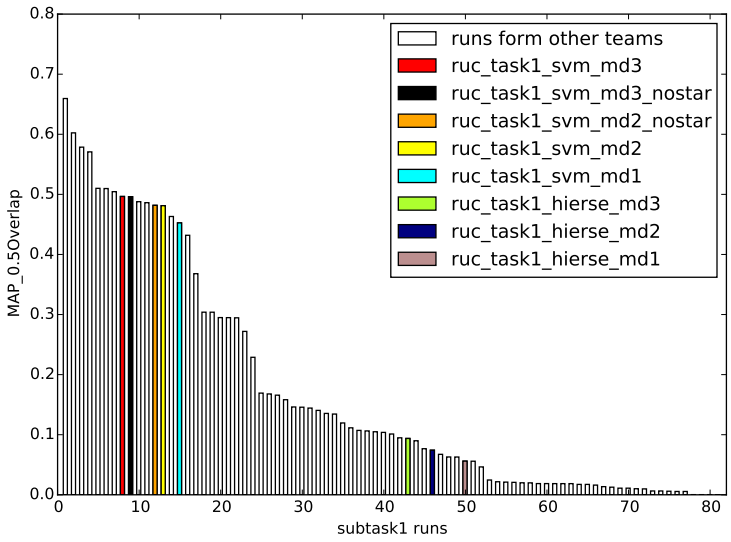


Fig. 3. Comparison of RUC-Tencent runs with other runs for the concept detection and localization task. Performance metric: MAP_0.5Overlap.

3 Task 2: Image Sentence Generation

We have used the MSCOCO dataset [5] in addition to the Dev2k dataset provided by the organizers. For the Dev2k dataset, we split it into a training set with 1,600 images, a validation set with 200 images, and a test set with 200 images.

3.1 Deep Models based Image Sentence Generation

Our image description system is built based on deep models proposed by Vinyals *et al.* from Google [19]. The deep model contains the following key components:

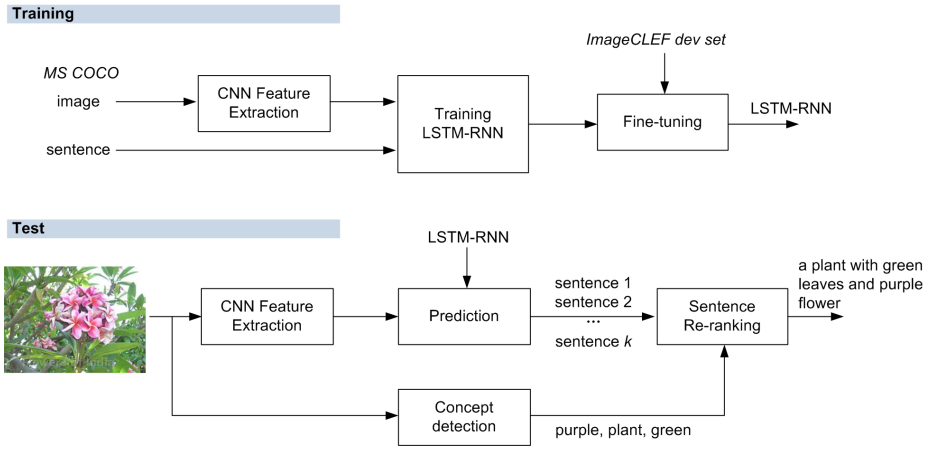


Fig. 4. An illustration of the RUC-Tencent image sentence generation system.

(1) a Convolutional Neural Network (CNN) for image encoding, and (2) a Long-Short Term Memory based Recurrent Neural Network (LSTM-RNN) for sentence encoding, and (3) a LSTM-RNN for sentence decoding. In our system as shown in Fig. 4, we use the pre-trained VGGNet [6] for CNN feature extraction. The LSTM-RNN implementation is from the NeuralTalk project[1]. The encoding LSTM-RNN and the decoding LSTM-RNN are shared.

In the training stage, we first train the LSTM-RNN on the MSCOCO dataset. We then fine-tune the model on the ImageCLEF Dev2k dataset using a low learning rate. Beam search is used in text decoding as in [19]. We finally re-rank the hypothesis sentences utilizing the concept detection results. Given an image, our system first generates k best sentences with confidence scores, meanwhile the concept detection system from task 1 provides m best detected concepts with confidence scores. If a detected concept appears in the hypothesis sentence, we call it a matched concept. We then compute the ranking score for each hypothesis sentence by matching the detected concepts with the hypothesis sentences as follows:

$$RankScore(hypoSent) = \theta \cdot conceptScore + (1 - \theta) \cdot sentenceScore, \quad (1)$$

where $conceptScore$ refers to the average of the confidence scores of all the matched concepts in the hypothesis sentence ($hypoSent$), $sentenceScore$ refers to the confidence score assigned to the hypothesis sentence by the RNN model. The parameter θ has been tuned on the Dev2k validation set. Fig. 5 showcases some examples of how re-ranking helps the system produce better image descriptions.

In the clean track, "golden" concepts, i.e., ground truth concepts by hand labelling, are provided. So we set the confidence score for each concept as 1. In

[1] `https://github.com/karpathy/neuraltalk`

(1) a plane taking off from a runway
(2) a plane taking off from the runway
(3) a plane flying in the sky
(4) an airplane flying in the sky
(5) a plane flying in the air
(6) an airplane is flying in the sky
(7) a plane taking off into the sky

(3) a plane flying in the sky
[flying | plane | sky]

(a)

(1) a plant with green leaves
(2) a cluster of yellow flowers
(3) a plant is growing in a garden
(4) a plant with green leaves and green leaves
(5) a plant with green leaves and purple flowers
(6) a cluster of yellow flowers in a garden
(7) a cluster of yellow flowers in a field

(5) a plant with green leaves and purple flowers
[purple | plant | green]

(b)

Fig. 5. Examples illustrating sentence re-ranking. Candidate sentences are shown in descending order on the left side, while the chosen sentences are shown on the right side. In square brackets are Flickr tags predicted by the tag relevance algorithm [11].

this condition, if the top-ranked sentence does not contain any of the "golden" concepts, the word in the top-ranked sentence with the closest distance to any one of the "golden" concepts will be replaced by that concept. The word distance is computed using a pre-trained word2vec[2] model [7].

[2] `code.google.com/p/word2vec`

3.2 Submitted Runs

We submitted six runs for the noisy track, and two runs for the clean track.

[**Noisy track**] **RUC_run1_dev2k** is our baseline trained on the Dev2k dataset without sentence re-ranking.

[**Noisy track**] **RUC_run2_dev2k_rerank-hierse** is trained on the Dev2k dataset, using the 251 concepts predicted by HierSE [4] for sentence re-ranking.

[**Noisy track**] **RUC_run3_dev2k_rerank-tagrel** is trained on the Dev2k dataset, using the Flickr tags predicted by the tag relevance algorithm [11] for sentence re-ranking.

[**Noisy track**] **RUC_run4_finetune-mscoco** is first trained on the MSCOCO dataset, and then fine-tuned (with a relatively low learning rate) on the Dev2k dataset, without sentence re-ranking.

[**Noisy track**] **RUC_run5_finetune-mscoco_rerank-hierse** is trained the same way as for RUC_run4_finetune-mscoco, using the 251 concepts predicted by HierSE for sentence re-ranking.

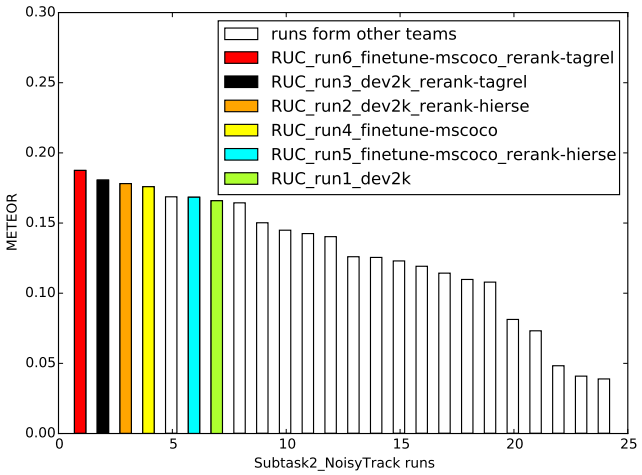
[**Noisy track**] **RUC_run6_finetune-mscoco_rerank-tagrel** is trained the same way as for RUC_run4_finetune-mscoco, using the Flickr tags predicted by the tag relevance algorithm for sentence re-ranking.

[**Clean track**] **RUC_run1_dev2k** is trained on the Dev2k dataset, with sentence re-ranking using the provided "golden" concepts.

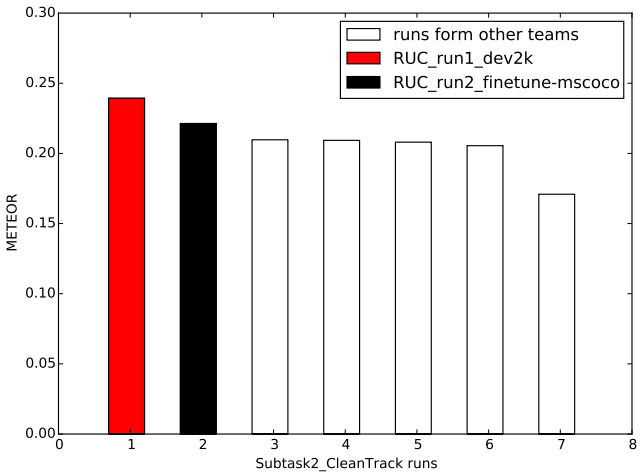
[**Clean track**] **RUC_run2_finetune-mscoco** is first trained on the MSCOCO dataset and then fine-tuned on Dev2k, with sentence re-ranking using the provided "golden" concepts.

Result Analysis The performance of the noisy track runs is shown in Fig. 6(a). We see that fine-tuning improves the system performance (RUC_run1_dev2k 0.1659 *versus* RUC_run4_finetune-mscoco 0.1759). Sentence re-ranking helps as well (RUC_run1_dev2k 0.1659 *versus* RUC_run2_dev2k_rerank-hierse 0.1781).

The performance of the clean track runs is shown in Fig. 6(b). It shows that the LSTM-RNN model trained on MSCOCO and fine-tuned on Dev2k is less effective than the model trained on Dev2k alone. This is somewhat contradictory to the results of the noisy track. For a more comprehensive evaluation, Table 2 shows precision, recall, and F1 scores of the two runs, from which we see that



(a) Noisy Track evaluation results



(b) Clean Track evaluation results

Fig. 6. Comparison of RUC-Tencent runs with other runs for the image sentence generation task. Performance metric: METEOR.

RUC_run2_finetune-mscoco obtains a higher precision score. This is probably because the reranking process based on the "golden" concepts makes the system more precision oriented.

Since the details of the test sets used in the two tracks are not available to us, we cannot do more in-depth analysis.

Table 2. Performance of the task 2 clean track.

Run	Mean METEOR	Mean Precision	Mean Recall	Mean F1
RUC_run1_dev2k	**0.2393**	0.6845	**0.4771**	**0.5310**
RUC_run2_finetune-mscoco	0.2213	**0.7015**	0.4496	0.5147

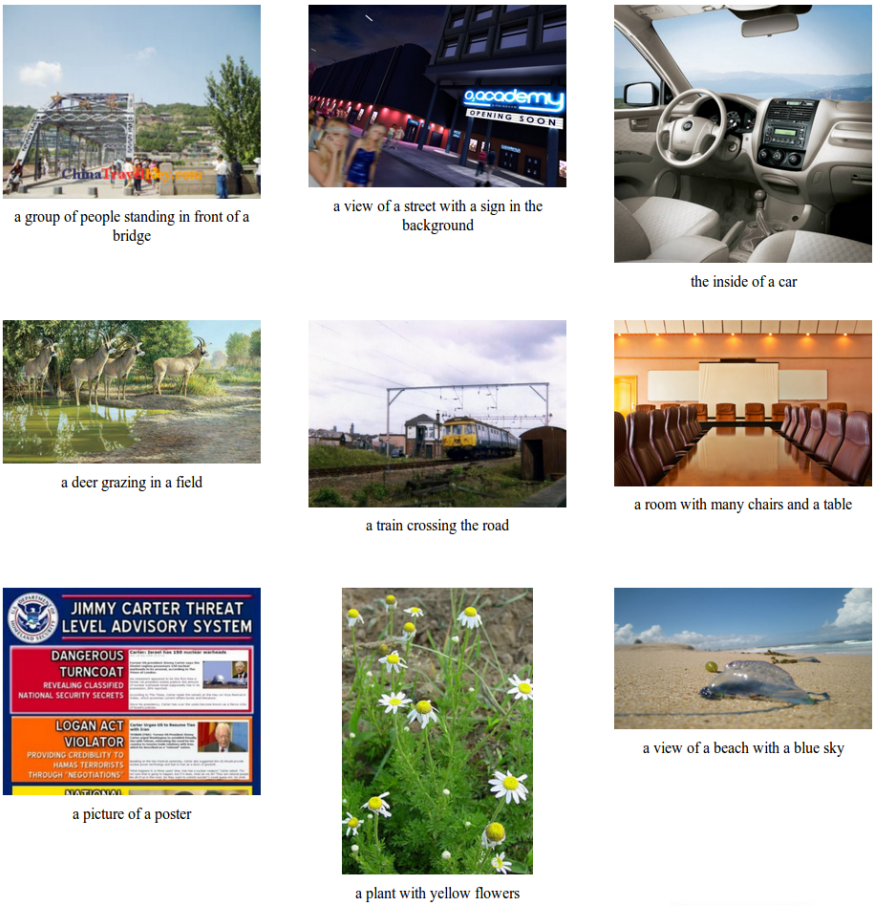


Fig. 7. Images with sentences generated by the RUC-Tencent image sentence generation system. The images are hand picked that showing our system performs well.

4 Summary and Discussions

For the concept detection and localization task, we find Hierarchical Semantic Embedding effective for resolving visual ambiguity. Negative Bootstrap improves the classification performance further. For the image sentence generation task, Google's LSTM-RNN model can be improved by sentence re-ranking.

Notice that for varied reasons including the limited number of submitted runs and the unavailability of detailed information about the test data, our analysis is preliminary, in particular a component-wise analysis is largely missing.

Acknowledgements. This research was supported by the National Science Foundation of China (No. 61303184), the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (No. 14XNLQ01), the Beijing Natural Science Foundation (No. 4142029), the Specialized Research Fund for the Doctoral Program of Higher Education (No. 20130004120006), and the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry. The authors are grateful to the ImageCLEF coordinators for the benchmark organization efforts [1, 20].

References

1. Gilbert, A., Piras, L., Wang, J., Yan, F., Dellandrea, E., Gaizauskas, R., Villegas, M., Mikolajczyk, K.: Overview of the ImageCLEF 2015 Scalable Image Annotation, Localization and Sentence Generation task. In: CLEF Working Notes. (2015)
2. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: CVPR. (2009)
3. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv:1408.5093 (2014)
4. Li, X., Liao, S., Lan, W., Du, X., Yang, G.: Zero-shot image tagging by hierarchical semantic embedding. In: SIGIR. (2015)
5. Lin, T., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. CoRR **abs/1405.0312** (2014)
6. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014)
7. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS. (2013)
8. Li, X., Liao, S., Liu, B., Yang, G., Jin, Q., Xu, J., Du, X.: Renmin University of China at ImageCLEF 2013 scalable concept image annotation. In: CLEF working notes. (2013)
9. Li, X., He, X., Yang, G., Jin, Q., Xu, J.: Renmin University of China at ImageCLEF 2014 scalable concept image annotation. In: CLEF working notes. (2014)
10. Li, X., Snoek, C., Worring, M., Smeulders, A.: Harvesting social images for bi-concept search. IEEE Transactions on Multimedia **14**(4) (Aug. 2012) 1091–1104
11. Li, X., Snoek, C., Worring, M.: Learning social tag relevance by neighbor voting. IEEE Transactions on Multimedia **11**(7) (Nov. 2009) 1310–1322
12. Li, X., Snoek, C., Worring, M., Koelma, D., Smeulders, A.: Bootstrapping visual categorization with relevant negatives. IEEE Transactions on Multimedia **15**(4) (Jun. 2013) 933–945
13. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research **9** (2008) 1871–1874

14. Li, X., Snoek, C.: Classifying tag relevance with relevant positive and negative examples. In: ACM MM. (2013)
15. Liao, S., Li, X., Shen, H.T., Yang, Y., Du, X.: Tag features for geo-aware image classification. IEEE Transactions on Multimedia **17**(7) (2015) 1058–1067
16. Uijlings, J., van de Sande, K., Gevers, T., Smeulders, A.: Selective search for object recognition. International Journal of Computer Vision **104**(2) (2013) 154–171
17. Felzenszwalb, Huttenlocher: Efficient graph-based image segmentation. International Journal of Computer Vision **59**(2) (2004) 167–181
18. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. arXiv:1409.0575 (2014)
19. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. CoRR **abs/1411.4555** (2014)
20. Villegas, M., Müller, H., Gilbert, A., Piras, L., Wang, J., Mikolajczyk, K., de Herrera, A.G.S., Bromuri, S., Amin, M.A., Mohammed, M.K., Acar, B., Uskudarli, S., Marvasti, N.B., Aldana, J.F., del Mar Roldán García, M.: General Overview of ImageCLEF at CLEF2015 Labs. Lecture Notes in Computer Science. (2015)