Combining Image Recognition with Knowledge Graph Embedding for Learning Semantic Attribute of Images

Rajat Patel
UMBC
rpatel12@umbc.edu

Mohit Khatwani
UMBC
khatwan1@umbc.edu

Abstract

Linking entities in the knowledge graph has been an important problem. Learning of images in the open world using language model has attracted lots of interest over the year. The advent of knowledge graphs provides a unique representation of semantic relationships between entities. Through this paper, we propose a joint learning model to learn images along with image captioned entity attribute representation to learn the semantic relationships from the knowledge graph embedding model. The target model premises to help us understand the semantic relationship between the attribute entities and implicitly provide a link prediction for these entities through a knowledge graph embedding model.

1 Introduction

The problem that we are trying to solve revolves around using the knowledge representations in the form of knowledge graphs embedding to learn the semantic relations between the caption image entities. Although learning semantic relationship among entities and using them for image captioning has been attempted to solve via many methods, we would like to explore the space in which we try and find the semantic relationship between the image attributes using a knowledge graph embedding model by also incorporating external information of image to learning the model and the predict the link between the given entities

Our solution basically consists of 4 stages, the first stage involves a object detection model (Ren et al., 2015) which would help learn the attributes of the given image. Once these attributes of the images are available we extract relationship between these entities from a knowledge graph like Concept Net (Speer et al., 2017), however for the current model we have build a custom knowledge base of entities and relationship, where the node represents the entities and the edges represents the relations.An example of the custom defined knowledge base in give in Fig 1. Then in the third stage using this temporary database we train a knowledge graph embedding model based on (Liu et al., 2016) and proposed model based on convolution neural network to learn these relationships for solving a link prediction problem. The learning of the knowledge graph embedding model is aimed to be enhanced with usage of external information as input feature, which would be learned from a pretrained VGG net (Simonyan and Zisserman, 2014). Thus solution aims to understand how an image feature helps in enhancing the performance of the knowledge graph embedding model in terms of accuracy as compared to the baseline model where the image would be used as an input feature.The datasets being used and the proposed solution architecture has been mentioned in architecture section ahead.

Following paper is organized as follows, section 2 includes previous work. In Section 3 we explain the architecture of the proposed model. In section 4 we explain the detailed methodologies which is followed by our results of baseline model and proposed model in section 5. Section 6 points out the limitation of our method. In section 7 we discuss future work. The paper is concluded in section 8.

2 Previous work

There has been a wide range of work on image captioning problems in supervised settings (Vinyals et al., 2015). One of the most famous implementations of image captioning is by Vinyals and co. at Google called as Neural Image Caption Generator. There is also a lot of research done on using semantic embedding for image captioning. One of the interesting work that

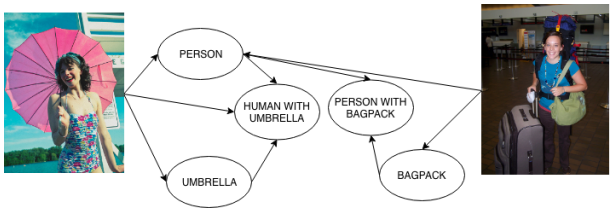

Figure 1: Knowledge Base description

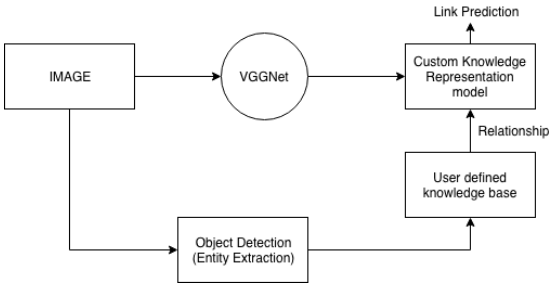

Figure 2: Architecture

we came across was done in 2017 by Li et al. (Li et al., 2017) for learning the semantic embedding for visual recognition for the Multi-label classification task. Another interesting work came from IBM research by Lonij et al. (Lonij et al., 2017) where they have built a link prediction for open-world visual recognition model using knowledge graphs. Our current research is very closely related to work done by IBM research (Lonij et al., 2017). The work demonstrated by Lonij and co. develops a solution to a link prediction problem by using a knowledge graph model with embedded entities and encoded image representation as inputs. Our current work is related to Lonij .et al; where we model our architecture design based on their implementation. While their implementation learns to solve a link prediction task using entity embeddings from the knowledge graph, we try and build our own small knowledge base with a closed world assumption such that we try and used the object recognized entities and the image in order to predict the actual class or relation between the image and the detected object entities. The proposed model of the knowledge base is mentioned in Fig. 1

3 Architecture

Fig.2 shows our proposed architecture. First step of this architecture is to extract entities in the image through an object detection model (FRCNN, YOLO)(Ren et al., 2015). The entities extracted from each image are converted to vectors using a word2vec model GloVe (Wikipedia contributors, 2018) , each image object recognized is an entity such that the resulting vector is the average over the number of object entities recognized. A customized closed world knowledge based has been used such that relationship between the image and the recognized by object entities could be predicted from this knowledge base as seen in Fig 1.The word entity embedding are given to custom knowledge representation model as prior along with the image in order to predict the image class or relationships. The baseline model is based on Neural Association Model(Liu et al., 2016), where we replace the output layer by a softmax layer function for multiclass classification. The proposed model is based on Convolution based knowledge representation model such that along with the image we also use prior information of the object entity embeddings to predict the link present in the closed world assumed knowledge base.

4 Methodology

The current model combines the information obtained from the object recognition model along with semantic information from a word embedding model to determine the semantic attributes of the images. In the process of building this we decided to build our own dataset of 7584 images distributed across 12 custom defined image label class extracted from COCO dataset (Lin et al., 2014) .The label categories for the images where defined on the basis of variability and strength of the images in the particular class. The predefined set of the image class labels along with their distributions in the training set is mentioned in Table 1.The images from the COCO image base have been selected with an effort on minimizing the effect of data skewness.

Objection detection model used here is state of art method YOLO (Redmon et al., 2015). YOLO uses a single CNN network for both classification and localizing the object using bounding boxes (Medium, 2018). The score for all the boxes detected is represented by a Tensor of shape 1×1. These scores are sorted in descending order and Non-max suppression algorithm is applied to remove any redundant boxes (Hosang et al., 2017). Intersection over union is used to remove redundant scores of bounding boxes. After this process we will be left with 3-5 boxes with maximum

Labels	Number of images
Human with umbrella	491
Tennis Racket	997
Baseball	942
Sportsball	530
person snowboarding	964
Kitchen electronics	31
Living Room	86
Traffic	98
Utensils	209
Person with bags	623
Animals	926
Human with animals	1699

Table 1: This table shows the labels considered and their respective count of images in training set.

score which are considered to give the classes of objects. Images to our YOLO v3 model are not directly passed, images are reshaped to size 416×416. Pre-trained weights were used for this YOLO model which was trained on COCO dataset. (thtrieu, 2018).

Image representation has been learned and extracted with a vector size of 4096 using a VGG net (Simonyan and Zisserman, 2014). The learned representation of these images would be further used for predicting the link between the object extracted entities and the image itself. Thus, we would use the image representation along with the object entity representation for our knowledge graph model for a link prediction task.

For creating word embeddings we have used GloVe, coined from Global Vectors. It is a model for distributed word representation (Wikipedia contributors, 2018). The distributed word representation model is based on skip gram based model for converting word to vector, each word spatial position is considered with respect left and right context words associated with current word. Thus, a vector embedding would define the syntactic as well as the semantic representation of the word. The embedding vector's dimension currently is hundred, however we can experiment with higher dimension of embedding vectors in future.

As a baseline model, we have built a feed-forward neural network with 3 hidden layer, and 2 Dropout layers in order to reduce the effects of over fitting. The model's hidden units dimension are 128, 64, 64 and 12, where the last layer provides the classification into 12 different image class types or image entity type. The baseline neural network models the input entity based word embedding vector, where the entities are derived from the image to predict the image class.

For our proposed model, we have used convolution neural network to process the compressed image representation which is followed by a concatenation of compressed image representation and word embedding. Further, the concatenated output is flattened and passed to couple of dense layers. To compress the image representation given by VGGNet we have used 5 convolution layers which are followed by max pool layers after every convolution layer. This output is concatenated and passed onto a single dense layer. This network is further ends with the output dense layer along with softmax activation function.

5 Results

5.1 Baseline Model

Fig. 5 and Fig. 6 show accuracy and loss value of our baseline model. This model only had the word embedding as input. Optimizer used for this model is Adam with learning rate of 0.01. Categorical cross entropy loss was used to predict the 12 class output. Architecture details of this model is given in table 3. The model was running for 200 epochs with a validation split of 0.2 i.e 20 percent of total data was taken as validation data and rest was used for training. We have also perform a 10 fold cross validation which receives accuracy 84.66%

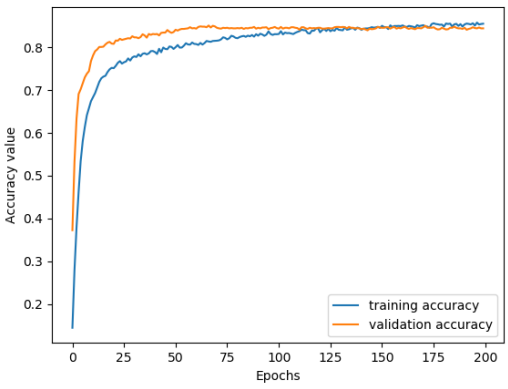

Figure 3: Accuracy for Baseline

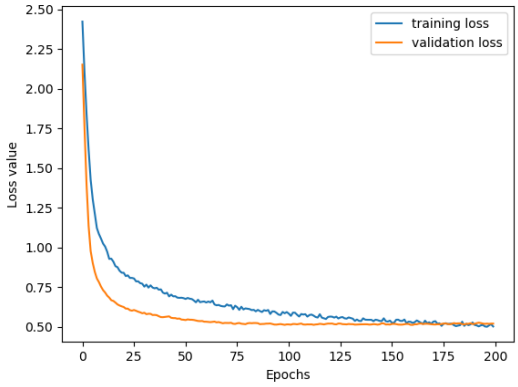

Figure 4: Loss for Baseline

5.2 Proposed Model

Proposed model has architecture as given in table 9. The input is given in two parts first is the 4096 vector of images and second is 100 sized vector of word embedding. This model is also optimized using Adam optimizer which has a learning rate of 0.01. We have also performed 10 fold cross validation and have received an accuracy of 80.13%. The proposed model work with prior of the embedding vector given the encoded information from an image to predict the class of the images class label which can be stated in terms of knowledge graph model as the relation between the predicted entity object and the image.

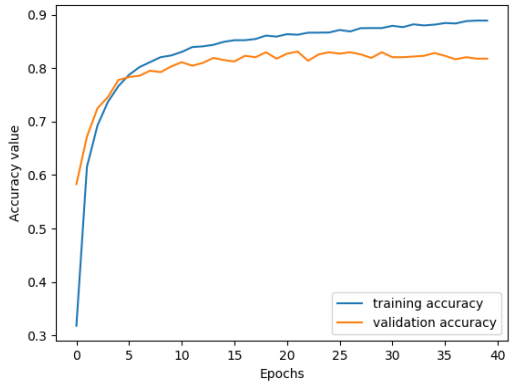

Figure 5: Accuracy for proposed model

6 Limitations

one of the major limitation of the current model is knowledge base that is defined with the proposed solution. The current knowledge base is based on closed world assumption such that knowledge

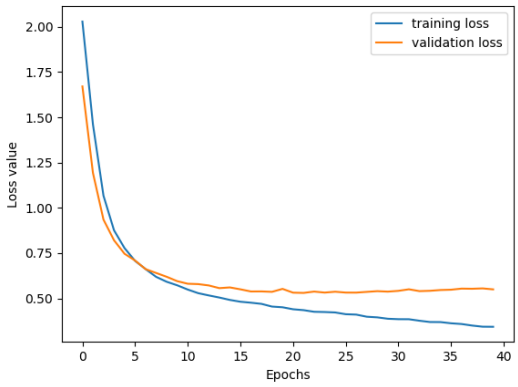

Figure 6: Loss for proposed model

Model	Accuracy (%)
Baseline	84.66
Proposed model	80.13

Table 2: Accuracy comparison of Baseline model and proposed model

only defines 12 given relation between the entities and the corresponding images acting as the entities. Another limitation if the amount of data that has been build in the custom knowledge has only 7584 images, due to which the proposed model does not give results that could be better than the proposed baseline model. Also, since we have tried and defined the our own knowledge there could be gaps and discrepancies in defining the relationships and entities for this knowledge base which might effect the performance of the knowledge base.

7 Discussion and Future Work

In future we would like to test this idea using a larger knowledge base with more classes and relations. ConceptNet is one of such examples which can be considered. Another future task can be to find a better way to select a subset of COCO dataset. This dataset should be inline with the relationships found in the knowledge graph found from ConceptNet.

8 Conclusion

In this paper, we propose a convolution neural network based knowledge representation model for solving the link prediction task. We compare our model with a baseline model which predicts the link using only the word embedding provided. Our

model uses information from both the image as well as the vector embedding based information of object entities recognized from a object recognition model (entities).

References

Jan Hendrik Hosang, Rodrigo Benenson, and Bernt Schiele. 2017. Learning non-maximum suppression. In *CVPR*, pages 6469–6477.

Dong Li, Hsin-Ying Lee, Jia-Bin Huang, Shengjin Wang, and Ming-Hsuan Yang. 2017. Learning structured semantic embeddings for visual recognition. *arXiv preprint arXiv:1706.01237*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Quan Liu, Hui Jiang, Andrew Evdokimov, Zhen-Hua Ling, Xiaodan Zhu, Si Wei, and Yu Hu. 2016. Probabilistic reasoning via deep learning: Neural association models. *arXiv preprint arXiv:1603.07704*.

Vincent Lonij, Ambrish Rawat, and Maria-Irina Nicolae. 2017. Open-world visual recognition using knowledge graphs. *arXiv preprint arXiv:1708.08310*.

Medium. 2018. yolo. `https://medium.com/diaryofawannapreneur/`. Accessed: 2018-12-17.

Joseph Redmon, Santosh Kumar Divvala, Ross B Girshick, and Ali Farhadi. 2015. You only look once: unified, real-time object detection. corr abs/1506.02640 (2015).

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*, pages 4444–4451.

thtrieu. 2018. Darkflow. `https://github.com/thtrieu/darkflow`.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Wikipedia contributors. 2018. Glove (machine learning) — Wikipedia, the free encyclopedia. [Online; accessed 18-December-2018].

9 Appendix

Layer name	Hidden Units/Dropout rate
Dense - 1	128
Dropout	0.5
Dense - 2	64
Dropout	0.5
Dense-3	64
Dropout	0.5
Dense - output	12

Table 3: Baseline model architecture

Layer name	Filter size /Options
Conv-1	4x1
Max Pool - 1	2x1
Conv-2	4x1
Max Pool - 2	2x1
Conv-3	4x1
Max Pool - 3	2x1
Conv-4	4x1
Max Pool - 3	2x1
Conv-5	4x1
Max Pool - 5	2x1
Flatten	-
Concatenation	img_input + emb_input
Dense	64
Dense	12

Table 4: Proposed model architecture