Self-Supervision for Scene Graph Embeddings

Brigit Schroeder[1], Adam Smith[1], Subarna Tripathi[2]
[1]UC Santa Cruz, [2]Intel Labs

Scene graph embeddings are used in applications such as image retrieval, image generation and image captioning. Many of the models for these tasks are trained on large datasets such as Visual Genome [1], but the collection of these human-annotated datasets is costly and onerous [2]. We seek to improve scene graph embedding representation learning by leveraging the already available data (e.g. the scene graphs themselves) with the addition of self-supervision. In self-supervised learning, models are trained for pretext tasks which do not depend on manual labels and use the existing available data. However, it is largely unexplored in the area of image scene graphs. In this work, starting from a baseline scene graph embedding model trained on the pretext task of layout prediction, we propose several additional self-supervised pretext tasks. The impact of these additions is evaluated on a downstream retrieval task that was originally associated with the baseline model [3]. Experimentally, we demonstrate that the addition of each task individually and cumulatively improves on the retrieval performance of the baseline model, resulting in near saturation when all are combined.

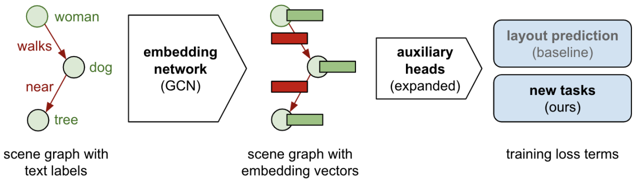
woman
walks
dog
near
tree
scene graph with text labels
embedding network (GCN)
scene graph with embedding vectors
auxiliary heads (expanded)
layout prediction (baseline)
new tasks (ours)
training loss terms

Figure 1: **Scene Graph Embedding Representation.**

A scene graph is a structured data format which encodes semantic and geometric relationships between objects. It contains a set of visual relationships containing a <*subject, predicate, object*> (”man walks dog”), where nodes representing objects identify the class of an object as well as provide a bounding box for the location of that object in an image. Edges representing relationships connect nodes. We use the human-annotated scene graphs from the Visual Genome (VG) dataset [1].

Following earlier work [3] (the baseline), we trained a graph convolutional neural network (GCN) (see Fig.1) on the pretext task of layout (bounding box) prediction. In our approach (similar to [4]), we add new auxiliary heads and self-supervised training losses to this otherwise unmodified backbone architecture (see Fig. 1 "new tasks") . For example, in Table 1, S_e,P_e,S_{box},P_{box} $\rightarrow O_{class}$ is the loss associated a pretext task that uses the embedding and bounding box parameters for each relationship's subject and predicate to predict the class of relationship's object. $O_{emb} \rightarrow O_{class}$ is a loss associated with the pretext task of predicting all classes of object nodes in a scene graph. masked SG$\rightarrow P_{class}$ is a loss which replaces the randomly selected 50% of a scene graph's predicate classes with a MASK token value and predicts withheld class label as a pretext task. Finally, supervised contrastive loss (Predicate SCL and Object SCL) [5] is applied to the predicate and object embeddings respectively (similar to the "encoded" representation of images traditionally used with SCL) to improve their learned representation.

Model	**R@1**	**R@25**	**R@50**	**R@100**
Baseline [3]	0.18	0.42	0.49	0.76
S_{emb},O_{emb},S_{box},O_{box} $\rightarrow P_{class}$	0.19	0.43	0.51	0.80
S_{emb},P_{emb},S_{box},P_{box} $\rightarrow O_{class}$	0.25	0.55	0.64	0.89
P_{emb},O_{emb},P_{box},O_{box} $\rightarrow S_{class}$	0.31	0.72	0.82	1.0
$O_{emb} \rightarrow O_{class}$	0.29	0.66	0.76	0.99
masked SG$\rightarrow P_{class}$	0.21	0.43	0.52	0.86
Predicate SCL	0.24	0.54	0.63	0.91
Object SCL	0.26	0.59	0.69	0.94
All above combined	**0.46**	**0.86**	**0.94**	**1.00**
Token match (ideal)	0.47	0.86	0.94	1.00
Random	0.00	0.00	0.01	0.06

Table 1: **Retrieval Evaluation Results.**

Our experiments evaluate the impact of the proposed new pretext tasks on a downstream visual relationship retrieval task. Queries, using visual relationships from the scene graphs of the form <*subject, predicate, object*>, are used to form an embedding vector (from the output embedding vectors in Fig. 1). A database of documents (in this case, images), where each contains a similar embedding-based visual relationship, is queried. Results are ranked by their L_2 distance to the query vector. For the purposes of evaluating relevance, we say that a query triple matches a candidate document triple if it has an exactly matching subject, predicate, and object. Summarizing retrieval performance using the Recall@k metric, Table 1 shows that each addition independently improves over the baseline, in some cases quite significantly, while their combination approaches ideal performance on this task (perfect matching of all documents and queries).

References

[1] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Li, D.A. Shamma, M.S. Bernstein, and F. Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *CoRR*, abs/1602.07332, 2016.

[2] Y. Liu, S. Pan, M. Jin, C. Zhou, F. Xia, and Philip S. Yu. Graph self-supervised learning: A survey. *arXiv preprint arXiv:2103.00111*, 2021.

[3] B. Schroeder and S. Tripathi. Structured query-based image retrieval using scene graphs. In *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[4] V. Vinay P. Maheshwari, R. Chaudhry. Scene graph embeddings using relative similarity supervision. In *AAAI*, 2021.

[5] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan. Supervised contrastive learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020.