Tensor Composition Net for Visual Relationship Prediction

Yuting Qiang[1,2]
qiangyuting.new@gmail.com

Yongxin Yang[2]
yongxin.yang@ed.ac.uk

Xueting Zhang[2]
Xueting.Zhang@ed.ac.uk

Yanwen Guo[1]
ywguo@nju.edu.cn

Timothy Hospedales [2,3]
t.hospedales@ed.ac.uk

[1] Department of Computer Science and Technology
Nanjing University
Nanjing, China

[2] School of Informatics
The University of Edinburgh
Edinburgh, UK

[3] Samsung AI Research Centre
Cambridge, UK

Abstract

We present a novel Tensor Composition Net (TCN) to predict visual relationships in images. Visual Relationship Prediction (VRP) provides a more challenging test of image understanding than conventional image tagging, and is difficult to learn due to a large label-space and incomplete annotation. The key idea of our TCN is to exploit the low-rank property of the visual relationship tensor, so as to leverage correlations within and across objects and relations, and make a structured prediction of all visual relationships in an image. To show the effectiveness of our model, we first empirically compare our model with Multi-Label Image Classification (MLIC) methods, eXtreme Multi-label Classification (XMC) methods and VRD methods. We then show that, thanks to our tensor (de)composition layer, our model can predict visual relationships which have not been seen in training dataset. We finally show our TCN's image-level visual relationship prediction provides a simple and efficient mechanism for relation-based image-retrieval even compared with VRD methods.

1 Introduction

Building on recent progress in image classification [14] and object detection [20], understanding the high order semantics of complex images is becoming increasingly topical in machine perception. This requires not only recognizing individual objects, but also predicting visual relationships [17, 22]. Extracting visual relationships in images is critical to comprehend the visual world beyond mere object classification.

To this end, many recent works address *Visual Relationship Detection (VRD)* [17, 27, 30]. These works commonly formulate visual relationships as *subject-predicate-object* triplets (e.g. *person-ride-bike*, *dog-next to-cat*) [27, 30]. With the supervision of both the category

and location (bounding box) of each object, they train or fine-tune an object detection network (e.g. Faster R-CNN) first, then a predicate prediction step is followed. In this paper, we promote the task of *Visual Relationship Prediction (VRP)*, where visual relationships are predicted, but not registered to specific bounding boxes. Accordingly, our VRP problem only require image level visual relationship annotations without any localization.

Due to the lack of object level annotations (bounding boxe and object category), we could simply regard each unique visual relationship triple as a distinct category and formulate VRP as a multi-label image classification (MLIC) problem. However, one critical challenge is that the space of potential relationships is combinatorially large. There are $n \times n \times m$ potential relationships for a dataset with n subjects, n objects and m predicates. A simple heuristic for managing this growth in label-space is to filter out visual relationships never seen during training. However the number of unique seen relationships is already large (over 20000 for Visual Genome dataset), which is orders of magnitude larger than that of common MLIC datasets (20 for PASCAL and 80 for MSCOCO).

More fundamentally, this strategy prevents us from predicting unseen visual relationships at runtime. In many object classification problems the train and test dataset easily share exactly the same label space. But due to huge label space and long tail distribution of visual relationships, it is common to have testing relationships that are unseen during training. Therefore, an effective VRP model should be capable of predicting unseen/rarely seen relationships. Fortunately, different relationships are semantically related [17]. For example, *person-ride-horse* might be understandable without any training examples, given prior experience of "person-ride-bike" and "horse" object, if the model can re-combine the riding predicate and horse object.

To enable such compositional reasoning, and to exploit correlations within- and across relationship triples, we model the space of relationships to predict as a three-way tensor instead of a long vector. This provides two benefits: (i) It enables zero-shot prediction of novel relationships at runtime, (ii) it provides knowledge sharing between every occurrence of any given subject, object, or predicate among all unique relationships in which it occurs. To achieve these benefits our model termed Tensor Composition Net for Visual Relationship Prediction (TCN-VRP) exploits the low rank property of the proposed tensor by incorporating a tensor composition layer into a neural network so that it can predict all potential relationships without inducing a combinatorial number of parameters.

To summarize our contributions: (i) We propose a simple and elegant neural tensor network to predict visual relationships in an image without filtering out unseen relationships. To the best of our knowledge, this is the first work to employ a neural tensor network to solve VRP based on image-level annotations; (ii) Thorough experiments show that TCN-VRP provides better visual relationship prediction compared to prior MLIC and XMC alternatives, especially for unseen or few-shot relationships. (iii) We show that TCN-based VRP provides a rich yet efficient query modality for image retrieval.

2 Related Work

Visual Relationships   With the release of visual relationship datasets [13, 17] and the continued advances of object detection methods [4], a growing body of work [27, 30, 51] has studied detecting *generic* visual relationships. These methods usually decompose VRD task as object detection and predicate recognition so that the problem is tractable. Message passing technology (e.g. graph convolution, RNN) is used in these methods to boost both object

detection and predicate recognition. Although recently popular, these methods require RPNs, object detectors and expensive message passing techniques for inference, which makes them complicated and computationally costly to apply in practice. Crucially, due to reliance on object detection, these models require high-quality annotations of object localization.

In this paper, we propose to address the VRP rather than VRD task. Although VRP produces less detailed information (no bounding boxes), it only requires image-level annotation to train. Furthermore, one key motivating application of visual relationship detection is relationship-based image-retrieval. This particular capability does not depend on detection and can be driven by VRP, making it simultaneously simpler and cheaper to train.

Multi-Label Classification   Multi-Label Image Classification(MLIC) aims to annotate images with multiple labels. Early attempts [1] tackle this problem by reformulating it as single-label classification. Recent studies [16, 26, 28] pay more attention on label dependencies to facilitate multi-label prediction. Various strategies including exploiting graph structure [3, 16], matrix completion [2] and recurrent neural networks [26] have been explored to model label correlations. However, many previous strategies can not be directly applied to VRP. For example, [3] use a predefined matrix to represent label co-occurrence, and information propagation is conducted by multiplying the feature map with this matrix. In our VRP context, the size of this matrix would be over $1e^{10}$, making it computationally intractable.

MLIC typically only considers dozens of object categories, another group of work focus on eXtreme Multi-label Classification (XMC), where the label space is extremely large (usually over 1 million). These works usually employ label embeddings or tree to build correlation between different labels and make a structed prediction. For our VRP task, one crucial problem is to model the semantic correlation among different labels, which has not been considered by previous XMC methods.

Tensor Decomposition   Tensor decomposition methods are widely used in relationship induction for knowledge-bases [18, 24]. Popular approaches include training a low-rank approximation of a label tensor to induce missing labels [18], or training low-rank factors to estimate a label tensor [24] using a neural network. These methods address completing the missing element in a single relationship tuple – without any grounding to an image. In contrast, we use tensor composition to generate a unique image-specific label tensor that predicts all relations in an input image. Few studies have used tensor methods in the context of VRD. [9] constructs a visual relationship tensor as prior to boost visual relationship detection [27]. It is completely different to ours in that we dynamically predict the tensor corresponding to each input image.

3   Method

Problem Definition and Notation   We assume a dataset of N images, annotated with image-level visual relationships, based on n object and m predicate types. Each image is annotated with a number of relationship triplets, and a triplet is represented as $\langle o_i, p_k, o_j \rangle$ to denote the fact that, in this image, the object $i \in \{1, 2, \ldots, n\}$ has the predicate $k \in \{1, 2, \ldots, m\}$ with the object $j \in \{1, 2, \ldots, n\}$. The set of relationship triplets can be naturally encoded as a 3-way relation tensor $\mathbf{T} \in \mathbb{R}^{n \times n \times m}$[1]. A tensor entry $\mathbf{T}_{ijk} = 1$ denotes the existence of a relationship triplet $\langle o_i, p_k, o_j \rangle$ in the image, otherwise $\mathbf{T}_{ijk} = 0$. Our overall goal is to predict

---

[1]Assuming that every object in an image can potentially take both subject and object role in a relation

a relationship tensor that describes the full set of relationships that exists in a given image. That is, given a set of images and associated label tensors $(\mathbf{x}^{(i)}, \mathbf{T}^{(i)})_{i=1}^N$, we want to learn a model $f(\cdot)$ that predicts the label tensor corresponding to an image $f(\mathbf{x}^{(i)}) \to \hat{\mathbf{T}}^{(i)} \approx \mathbf{T}^{(i)}$.

Computational Challenge   A naïve predictor for $\mathbf{T}$ given $\mathbf{x}$ contains $d \times (n \times n \times m)$ parameters, where d is the image feature dimension. In modern CNNs, d is usually larger than 1000, and for the commonly used benchmark dataset, n or m is at the scale of 100. This means the number of parameters in predictor alone scales up to 1 billion easily, which is 15 times larger than the *whole* ResNet-152. Clearly, the computational cost is prohibitive, and we need to find a lightweight proxy of $\mathbf{T}$ instead of predicting it directly.

Tensor Decomposition   Since $\mathbf{T}$ is inherently a tensor, a route to a more compact form is tensor decomposition. Tucker-decomosition [15] is a popular and safe choice for tensor decomposition. It decompose $\mathbf{T}$ as $\mathbf{T} = \mathbf{S} \times_1 A \times_2 B \times_3 C$. Here $\times_i$ denotes tensor-matrix dot product at the ith axis of tensor. $\mathbf{S}$ is the 3-way tensor sized $r_1 \times r_2 \times r_3$ (called *core* tensor), and A, B, C are matrices sized $r_1 \times n$, $r_2 \times n$, $r_3 \times m$, respectively (called factor matrices). (r_1, r_2, r_3) is called Tucker rank of tensor, and they are hyper-parameters in our context. Since we usually set $r_1 \ll n$, $r_2 \ll n$, $r_3 \ll m$, the number of entities in $\{\mathbf{S}, A, B, C\}$ is much smaller than the $\mathbf{T}$, but we can reconstruct $\mathbf{T}$ from $\{\mathbf{S}, A, B, C\}$ for the final prediction.

Knowledge Sharing   We can propose to predict $\{\mathbf{S}, A, B, C\}$ then reconstruct $\mathbf{T}$ instead of predicting $\mathbf{T}$ directly.   A question that arises, is whether it is necessary to deduce all of $\{\mathbf{S}, A, B, C\}$ from the image feature?

In Tucker-decomposition, A, B, or C can be understood as word embedding matrices defining three *distinct* latent spaces. The use of three distinct spaces is intuitive because: (i) The fact that A and B are not tied, i.e., $A \neq B$, reflects that words have different embeddings depending on whether they are subject or object [25]. (ii) The fact that $r_1 \neq r_2 \neq r_3$ reflects that we do not need the same capacity to encode the words from different vocabularies.

If we exhaustively compute the outer products of the columns of A, B, C, we get a stack of $r_1 \times r_2 \times r_3$ tensors, each sized $n \times n \times m$, and the role of core tensor $\mathbf{S}$ is simply to rescale those $n \times n \times m$ tensors in order to reconstruct the original tensor. Based on this understanding, each column in the factor matrix encodes embedding of a specific word, and the core tensor contains the coefficients to recover the relationships of the given image with those basis. Therefore, we predict the core tensor $\mathbf{S}$ from the given image only, and factor matrices A, B, C are shared among all images, i.e., not conditioned on image features.

3.1   Tensor Composition Network

Network Design   Based on the reasoning above, we present our model – Tensor Composition Network (TCN), illustrated in Fig. 1. TCN consists of three components: (i) the image feature extractor $g_\theta(\mathbf{x})$ that produces a d-dimensional fused feature vector for a given image $\mathbf{x}$ (ii) the predictor that deduces the core tensor from image feature $\mathbf{S} = h_\phi(g_\theta(\mathbf{x}))$ (iii) the composer that generates the final prediction $t_{A,B,C}(S) = \mathbf{S} \times_1 A \times_2 B \times_3 C$. Here h_ϕ is a fully-connected layer that links every neuron produced by g_θ to every neuron in $\mathbf{S}$, which results in $d \times r_1 r_2 r_3$ parameters. The trainable parameters are $\{\theta, \phi, A, B, C\}$.

Similar to many related studies [14, 20], we use a CNN to extract image features. To exploit features at different levels of abstraction that are helpful to predict relationships in scenes, we employ Global Average Pooling (GPA) to extract image features at multiple depths. The fused feature could enable the subsequent relationship prediction module to be aware of both local and global features.
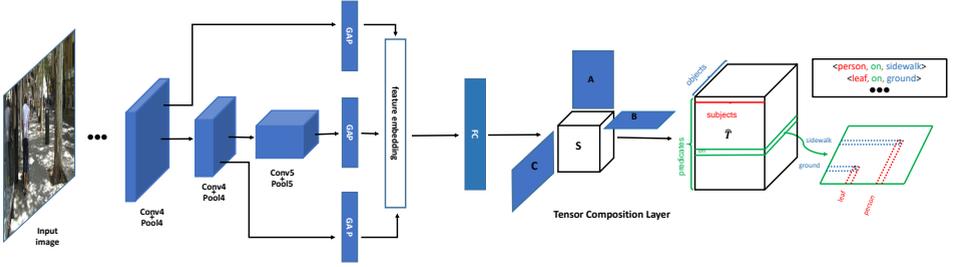
Figure 1: Overview of our Tensor Composition Net for Visual Relationship Prediction model. An image (left) is to be annotated with relevant ⟨subject, predicate, object⟩ relationship triplets (right). These triplets can be encoded as a relation tensor $\hat{\mathbf{T}}$, where a positive entry at $\hat{\mathbf{T}}_{i,j,k}$ indicates the corresponding triplet exists. The tensor is generated via Tucker composition where the core tensor $\mathbf{S}$ is predicted from the image and contracted with three factor matricies A, B, C that learn subject, predict and object embeddings. Blue blocks are learned parameters, and uncolored blocks (i.e., feature embedding, $\mathbf{S}$ and $\hat{\mathbf{T}}$) are activations.

With a fully connected layer, an intermedia vector is predicted from extracted image feature and it would be further folded as the core tensor ($\mathbf{S}$). Finally, the visual relationship tensor is generated via Tucker composition where the core tensor $\mathbf{S}$ is contracted with three factor matricies A, B, C that learn subject, predict and object embeddings.

Loss Function When we get the predicted tensor $\hat{\mathbf{T}} = t(h(g(\mathbf{x})))$, we adopt to softmax [6] loss, which has been proven effective for multi-label prediction problem. Softmax loss is defined as,

$$\ell(\hat{\mathbf{T}}, \mathbf{T}) = -\text{sum}(\frac{\text{flatten}(\mathbf{T})}{\text{sum}(\mathbf{T})} \odot \log(\text{softmax}(\text{flatten}(\hat{\mathbf{T}})))) \tag{1}$$

where $\odot$ denotes element-wise product. This is essentially a cross entropy loss with ground truth probability of $\frac{\text{flatten}(\mathbf{T})}{\text{sum}(\mathbf{T})}$.

4 Experiments

As far as we know, this is the first work to address image level visual relationship prediction. To evaluate of our model, we compare it with classic and state of the art MLIC and XMCmethods, with a particular focus on unseen and few-shot relationship prediction, and relation-based image retrieval (RBIR).

Datasets VR [17] and Visual Genome [13] are commonly used benchmark datasets for scene understanding [27, 30, 31]. As a small and early benchmark, VR only contains 5000 images with 37993 relationship instances in total. In contrast, Visual Genome contains more than 100k images but its raw annotations are very noisy, so data cleansing is needed before using it for visual relationship prediction. Among several cleaned version of this dataset [13, 27, 31], we use VG200 [31] since it involves a relatively large number of images, object and predicate categories. We provide detailed statistics and low-rank analysis of the two datasets in supplemental material.

Implementation Details For fair comparison, we employ VGG-16 [23] pretrained on

ImageNet [21] as backbone network to extract features for all compared methods. We set relative error $\varepsilon = 0.02$ (VR) and $\varepsilon = 0.05$ (VG200) for HOSVD to initialize A, B, C. Our model is trained for 40 epoches, using SGD with momentum, batchsize of 16 and an initial learning rate of 2×10^{-3}, which is divided by 2 every 10 epoches. To address the problem of imbalanced distribution of visual relationsip labels, we use weighted sampling during batch generation. Specifically, we first assign each existing visual relationship label l_i ($i = 1, 2, \ldots, L$) with a weight w_{l_i}, $w_{l_i} = \sum_{j=1}^{L} t_{l_j} / t_{l_i}$, t_{l_i} denotes the number of times that the label l_i occurs in the dataset. And the sampling weight for each image is $w_I = \sum_{l \in I} w_l$, so that images with rare relationship types or multiple relationships are prioritized during training.

4.1 Visual Relationship Prediction

Setup   Given an image, we aim to predict the set of *subject-predicate-object* triplets in it. Based on the predicted tensor $\hat{\mathbf{T}}$, we sort all potential triplets by their scores $\hat{\mathbf{T}}_{ijk}$ and select the top K triplets as our predicted relationships.

Metrics   As the annotated relationships in VR and Visual Genome are known to be incomplete [17, 27], we follow previous works [27, 30, 31] by using Top-K recall to measure performance. Specifically, we use *Recall@K* ($K = 20$, 50 and 100) and calculate the proportion of correctly predicted ground truth relationships in the top K most confidently predicted relationships.

Competitors   We compare TCN-VRP with both MLIC and XMC methods. Note that many state-of-the-art MLIC works employ a relational matrix to represent label dependencies which is not feasible for VRP due to the large label space, thus we compare: (1) WARP [5], which explores different loss functions for multi-label classification. We evaluate all three loss functions they consider (i.e. softmax [5], pairwise ranking [11] and weighted approximate ranking (WARP) losses). (2) Att-Consist [7], a two-branch network which introduces a new loss that measures the attention heatmap consistency between origin image and its transformed image to their network. (3) CNN-RNN [26] proposed a multi-label RNN model to sequentially predict the labels for each image, and the recurrent neurons in their model can capture high-order label co-occurrence dependency. For XMC competitors, we compare with 3 state-of-the-art methods : (4)Slice[10], a large-scale multi-label classification algorithm for low-dimensional, dense, deep learning features. (5) ExMLDS [8] leverage word embedding techniques for extreme multi-label classification. (6) Parabel [19] promote XMC problem based on balanced label tree. For fair comparison, we employ the same feature extraction module as our TCN-VRP for all XMC competitors. (7) VTransE[31] exploits embedding translation from the natural language processing to enable object-relation knowledge transfer and improve relationship detection. (8) VSPNet [29] proposed a bipartite message passing framework to detect visual relationships in an image. A graph-alignment algorithm is employed to enable the network to be trained without bounding-box annotation. Note that unlike the other competitors VTransE uses strong supervision (bounding boxes) during training, which provides an advantage.

Results   Table 1 shows that our model is competitive with state-of-the-art alternatives. We attribute this success to our low rank Tensor composition layer, which compresses the label space and models label correlations with few learnable parameters. In comparison, CNN-RNN [26] models label correlation through sequential recurrence, outputting a label at each time-step given the previous steps' prediction. A wrong prediction at one step can affect all later predictions, a problem which is exacerbated in datasets such as VG200 with

	Model	VR			VG200		
		R@20	R@50	R@100	R@20	R@50	R@100
MLIC	SoftmaxLoss [5]	23.73	36.26	46.31	30.32	43.32	50.85
	PairwiseLoss [5]	24.94	37.56	46.91	30.72	45.05	56.41
	WARPLoss [5]	23.70	34.18	41.93	30.28	43.23	52.94
	CNN-RNN [26]	23.80	35.90	44.94	26.13	39.67	50.74
	Att-Consist [7]	18.71	27.74	36.13	9.89	15.81	21.44
XMC	Slice[10]	23.29	32.26	39.69	29.43	40.12	44.37
	ExMLDS [8]	16.61	23.16	27.08	22.45	32.21	38.46
	Parabel [19]	24.31	29.48	34.64	28.77	39.43	43.35
VRD	VTransE* [31]	21.75	30.72	37.26	15.49	22.73	28.08
	VSPNet [29]	-	-	-	10.08	14.95	18.71
	Our TCN-VRP	**29.42**	**43.54**	**53.46**	**32.97**	**47.82**	**59.04**

Table 1: Comparison of methods for relationship prediction. We re-evaluated Att-Consist [7] using their official code but replacing the backbone with VGG16 newtork. Since CNN-RNN [5] and [26] (WARP, Pairwise, Softmax) don't release their code, we implement their network according to their papers. * Strongly supervised.

more relations per image. WARP [5] introduces weighted approximate ranking loss to optimize top-k annotation accuracy, which means the advantage of WARP loss heavily rely on complete annotation for each image. However, acquiring a completely annotated VRP dataset is particularly expensive since there could be hundreds of relationships in each image. Att-Consist [7] addresses MLIC by penalizing the discrepancy between heatmaps of original image and transformed image in loss function. However, the increased complexity of optimization makes it easy to be trapped in a local minima.

VRD methods also provide potential competitors. In particular weakly-supervised VRD methods are fair competitors which use the same supervision (image-level triples) as the other competitors. Our results show that VRD methods do not necessarily perform well for our VRP problem. VSPNet [29] outperformed many alternatives on weakly supervised VRD. However, it achieves comparatively low recall on our VRP problems. Our TCN can even outperforms the strongly supervised VRD method VTransE. This is because the VRD based methods heavily rely on the object detection module, If an object in the scene is missed by their detector (an event that is especially risky in the weak supervision case), VTransE and VSPNet can never recover. All relations involving that object are automatically missed.

Qualitative Results Qualitative results of our model's relationship prediction in Figure 2 suggest that our predictions are better than the quantitative results imply, since many predictions that do not exist in the ground-truth (blue) are actually correct. On the other hand, small objects like "leaf" in the bottom-left image are not recognized by our model.

4.2 Few-shot and Zero-shot Relation Learning

Due to the long tail distribution and large label space, it is infeasible to build a VRP dataset with sufficient annotated instances for each relationship category. Some relationships may only occur few times or even never in the training dataset. Therefore, it is important for a
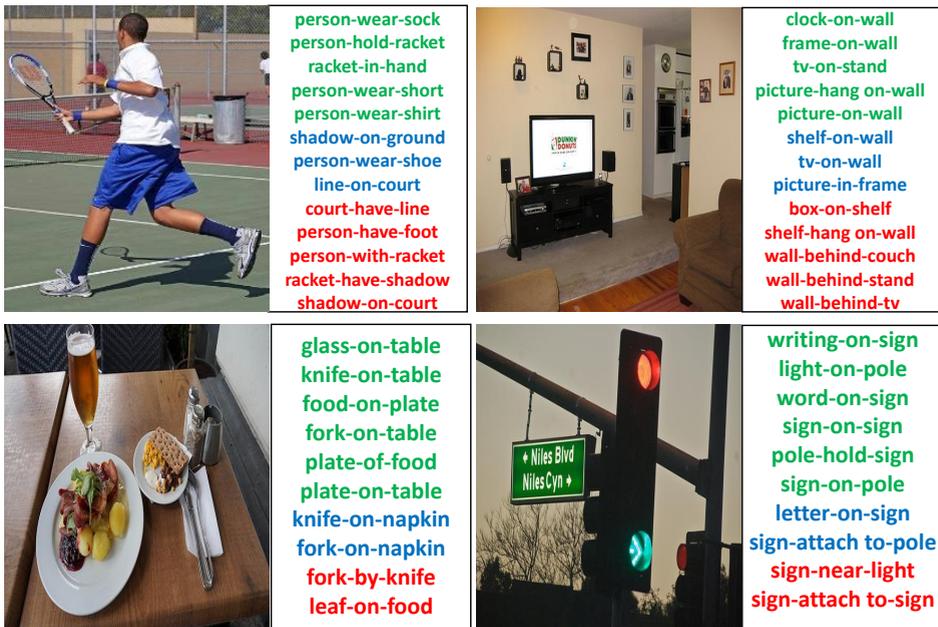
Figure 2: Qualitative examples of TCN-VRP relationship prediction using Recall@8. Sub-figure: Input image (left) and top 8 relationship predictions (right). Green: True positive predictions. Blue: False positive predictions. Red: False negative predictions.

VRP model to be able to predict visual relationships with few examples.

Setup, Metrics and Competitors   We compare the competing methods on *m*-shot ($m = 0, 1, 5$) learning, by selecting those triplets which exist in the test dataset but occur $\leq m$ times in training. We use the sparser VR dataset for this experiment, and compare the same set of methods using *Recall@K* evaluation as per general VRP. Existing VRD methods [? ?] tackling the zero-shot setting are trained under a strongly supervised condition and therefore are not directly comparable.

Results   Table 2 presents results on few-shot visual relationship prediction. XMC methods build correlations between different labels through label embeddings or a balanced tree, which helps them to perform better for few-shot relation prediction than MLIC methods. Our TCN-VRP outperforms both MLIC and XMC methods by a large margin on few-shot learning, confirming that our tensor composition layer helps our model to learn from few examples. The top 100 recall of TCN-VRP is at least 1.4 times better then the competitors. And the advantage is greater for 1 shot learning, while MLIC competitors achieve less than 1% recall@100 for 1-shot relationship prediction, compared to our 6% for this task.

4.3 Relation-Based Image Retrieval (RBIR)

A key benefit of visual relationship prediction is to enable image retrieval by more sophisticated relationship-triplet queries (e.g., *person-push-bike*, *dog-ride-surfboard*). Recently,

	Model	0-shot		1-shot		5-shot	
		R@50	R@100	R@50	R@100	R@50	R@100
MLIC	SoftmaxLoss[5]	-	-	0.37	0.81	4.35	7.96
	PairwiseLoss[5]	-	-	0.12	0.31	1.14	2.94
	WARPLoss[5]	-	-	0.00	0.00	0.00	0.00
	CNN-RNN [26]	-	-	0.00	0.12	0.63	2.43
	Att-Consist[7]	-	-	0.00	0.00	0.00	0.00
XMC	Slice[10]	-	-	2.04	3.84	6.90	12.11
	ExMLDS[8]	-	-	0.0	0.00	0.0	0.0
	Parabel[19]	-	-	2.11	4.28	4.63	8.43
	Our TCN-VRP	**1.03**	**3.24**	**2.56**	**6.03**	**7.74**	**16.02**

Table 2: Few-shot visual relationship prediction results on VR. "-": Previous MLIC and XMC methods can't be applied to zero-shot visual relationship prediction.

VRD-based methods have been applied to this task. We explore whether VRP is sufficient to perform RBIR well, given its appealing cheaper annotation requirements, and simpler inference method.

Setup and Metrics   Given a query triplet and a set of test images, we predict a relation score tensor for each image. We then sort all the images according to their scores for the query triplet. There could be multiple images containing the query triplet, so we regard an image as a correct hit if it has at least one query triplet in its annotation. We adopt the same setting as [51] which used Median rank (**Med r**) as metric. Median rank refers to the the median rank of the most confident correctly retrieved image. To show that our VRP model is sufficient for RBIR though it doesn't localize relationships, we use the existing protocol of evaluating using the *top 1000-frequent* triplets as queries [51].

Competitors   To show the effectiveness of our TCN-VRP for RBIR, we compare against VRD competitors. Specifically, we compare with VisualPhrase [22], DenseCap [12], LP [17] and VTransE [51]. Note that these are all strongly supervised methods that benefit from box-level supervision during training that is not used by our TCN.

Results   Tab. 3 shows that our TCN-VRP performs relation-based image retrieval comparably or better than state of the art VRD methods on both VR and VG200 benchmarks. One reason for this is the sparse annotation in the VR dataset, which makes it hard to finetune an accurate object detector on it, and that further affects the performance of VRD methods. In contrast, our TCN-VRP is robust to sparse annotation and still achieves good performance on this dataset. Qualitative results of our model for RBIR are provided in supplemental material.

Dataset \ Method	VisualPhrase	DenseCap	LP	LP-VLK	VTransE	TCN-VRP
VR [17]	204	199	211	137	41	**18**
VG200 [51]	18	13	-	-	7	**6**

Table 3: RBIR (Relation-Based Image Retrieval) results versus VRD alternatives (Top-1000 frequent triplets). Metric is median rank ($\downarrow$) of the target image.   Results for VisualPhrase [22], DenseCap [12], LP [17] and VTransE are reported in [51].

4.4 Ablation Study

We finally perform some further analysis to better understand our network. To anwser wheter Tucker factorization provides a good low-rank assumption to use as a layer in our model, we compare with a canonical polyadic (CP) alternative [15] (TCN-VRP-CP). CP decomposes a three way tensor $\mathbf{T} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$ as a sum of R rank one tensors, i.e. $\mathbf{T} = \sum_{r=1}^{R} u_r^{(1)} \circ u_r^{(2)} \circ u_r^{(3)}$, where $\circ$ denotes outer product. We can predict all these $u_r^{(i)}$ (i=1,2,3) based on the extracted features and then perform CP tensor composition to construct the relation tensor $\mathbf{T}$. We then attempt to regard core tensor (i.e. $\mathbf{S}$) as parameter and predict factor matrics (i.e. A, B, C) for each image (TCN-VRP-ABC). To evaluate the feature extraction strategies, We compare: (i) TCN-VRP-FC. which takes the more common strategy [23] of flattening the feature map of last max pooling layer. (ii) TCN-VRP-FL. Instead of just flattening the feature from last max-pooling layer, we use a GAP layer to embed the feature from last layer.

Method	VR			VG200		
	R@20	R@50	R@100	R@20	R@50	R@100
TCN-VRP-CP	18.21	25.13	31.29	24.21	35.06	44.01
TCN-VRP-ABC	19.78	26.93	32.64	27.67	39.59	48.72
TCN-VRP-FC	26.70	39.00	48.03	30.25	45.32	55.76
TCN-VRP-FL	27.86	41.01	50.87	30.05	43.80	54.57
TCN-VRP	29.43	43.54	53.46	32.97	47.82	59.04

Table 4: Ablation study of our TCN-VRP model

Results in Table 4 show that: (i) Tucker is much better than CP-composition for relationship prediction. Our Tucker decomposition model explicitly separates common knowledge about object/predicate from specific information about an image, which makes it more reliable and stable to train. (ii) Core tensor ($\mathbf{S}$), rather than factor matrics (A, B, C) should be regarded as image-specific. This corroborates our earlier analysis about Tucker-decomposition for visual relationship tensor. (iii) The GAP feature extraction helps our model resist overfitting by extracting a lower dimensional representation. (iv) Our TCN-VRP can be further improved by fusing features from different pooling layers to combine global and local information.

5 Conclusion

We proposed a neural tensor network to predict visual relationships in images. By introducing a Tensor composition layer, we compressed the relation tensor label-space and leverage semantic correlations between relationships. We achieve better visual relationship prediction compared to related MLIC methods. We also achieve competitive relation-based image-retrieval performance with strong supervised VRD works. In future we will explore scaling annotation by using existing caption data together with text parsing to automatically acquire weak relation annotation at large scale.

References

[1] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern recognition*, 37(9):1757–1771, 2004.

[2] R. Cabral, F. D. l. Torre, J. P. Costeira, and A. Bernardino. Matrix completion for weakly-supervised multi-label image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):121–135, Jan 2015. ISSN 0162-8828. doi: 10.1109/TPAMI.2014.2343234.

[3] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *CVPR*, pages 5177–5186, 2019.

[4] Ross Girshick. Fast r-cnn. In *ICCV*, 2015.

[5] Yunchao Gong, Yangqing Jia, Thomas Leung, Alexander Toshev, and Sergey Ioffe. Deep convolutional ranking for multilabel image annotation. *arXiv preprint arXiv:1312.4894*, 2013.

[6] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*, 2009.

[7] Hao Guo, Kang Zheng, Xiaochuan Fan, Hongkai Yu, and Song Wang. Visual attention consistency under image transforms for multi-label image classification. In *CVPR*, pages 729–739, 2019.

[8] Vivek Gupta, Rahul Wadbude, Nagarajan Natarajan, Harish Karnick, Prateek Jain, and Piyush Rai. Distributional semantics meets multi-label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3747–3754, 2019.

[9] Seong Jae Hwang, Sathya N. Ravi, Zirui Tao, Hyunwoo J. Kim, Maxwell D. Collins, and Vikas Singh. Tensorize, factorize and regularize: Robust visual relationship learning. In *CVPR*, 2018.

[10] Himanshu Jain, Venkatesh Balasubramanian, Bhanu Chunduri, and Manik Varma. Slice: Scalable linear extreme classifiers trained on 100 million labels for related searches. In *WSDM '19, February 11–15, 2019, Melbourne, VIC, Australia*, 2019.

[11] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM. ISBN 1-58113-567-X. doi: 10.1145/775047.775067. URL http://doi.acm.org/10.1145/775047.775067.

[12] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016.

[13] Joshua, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *International Journal of Computer Vision*, 123 (1):32–73, May 2017. ISSN 1573-1405. doi: 10.1007/s11263-016-0981-7. URL https://doi.org/10.1007/s11263-016-0981-7.

[14] Pooja Kamavisdar, Sonam Saluja, and Sonu Agrawal. A survey on image classification approaches and techniques. *International Journal of Advanced Research in Computer and Communication Engineering*, 2(1):1005–1009, 2013.

[15] T. Kolda and B. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3): 455–500, 2009.

[16] Qiang Li, Maoying Qiao, Wei Bian, and Dacheng Tao. Conditional graphical lasso for multi-label image classification. In *CVPR*, pages 2977–2986, 2016.

[17] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016.

[18] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *ICML*, 2011.

[19] Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. 2018.

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[21] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[22] Mohammad Amin Sadeghi and Ali Farhadi. Recognition using visual phrases. In *CVPR*, 2011.

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

[24] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with neural tensor networks for knowledge base completion. In *NIPS*, 2013.

[25] Cunchao Tu, Han Liu, Zhiyuan Liu, and Maosong Sun. Cane: Context-aware network embedding for relation modeling. In *ACL*, 2017.

[26] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu. CNN-RNN: A unified framework for multi-label image classification. In *CVPR*, 2016.

[27] Danfei Xu, Yuke Zhu, Christopher Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017.

[28] Hao Yang, Joey Tianyi Zhou, and Jianfei Cai. Improving multi-label learning with missing labels by structured semantic correlations. In *ECCV*, pages 835–851. Springer, 2016.

[29] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Weakly supervised visual semantic parsing. In *CVPR)*, June 2020.

[30] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018.

[31] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017.