Abstract

Social relationships (e.g., friends, couple etc.) form the basis of the social network in our daily life. Automatically interpreting such relationships bears a great potential for the intelligent systems to understand human behavior in depth and to better interact with people at a social level. Human beings interpret the social relationships within a group not only based on the people alone, and the interplay between such social relationships and the contextual information around the people also plays a significant role. However, these additional cues are largely overlooked by the previous studies. We found that the interplay between these two factors can be effectively modeled by a novel structured knowledge graph with proper message propagation and attention. And this structured knowledge can be efficiently integrated into the deep neural network architecture to promote social relationship understanding by an end-to-end trainable Graph Reasoning Model (GRM), in which a propagation mechanism is learned to propagate node message through the graph to explore the interaction between persons of interest and the contextual objects. Meanwhile, a graph attentional mechanism is introduced to explicitly reason about the discriminative objects to promote recognition. Extensive experiments on the public benchmarks demonstrate the superiority of our method over the existing leading competitors.

1 Introduction

Social relationships are the foundation of the social network in our daily life. Nowadays, as intelligent and autonomous systems become our assistants and co-workers, understanding such relationships among persons in a given scene enables these systems to better blend in and act appropriately. In addition, as we usually communicate via social media like Facebook or Twitter, we leave traces that may reveal social relationships in texts, images, and video [Fairclough, 2003]. By automatically capturing this hidden information, the system would inform users about potential privacy risks. In image analysis tasks, most works are dedicated to recognizing visual attributes [Huang et al., 2016] and visual relationships [Lu et al., 2016]. However, the aforementioned applications require recognizing social attributes and relationships, which receives less attention in the research community. In this work, we aim to address the task of recognizing the social relationships of person pairs in a still image.

Reasoning about the relationship of two persons from a still image is non-trivial as they may enjoy different relationships in different occasions that contain different contextual cues. For example in Figure 1, given several persons with business wear and with some office supplies such as keyboard around, they are likely to be colleagues. In contrast, if the persons are sitting in a room that has some household supplies like the bed, they tend to be family members. Thus, modeling such correlations between social relationships and contextual cues play a key role in social relationship recognition. Existing works either merely fixate on the regions of persons of interest [Sun et al., 2017] or exploit category-agnostic proposals as contextual information [Li et al., 2017] to perform prediction. Despite acknowledged successes, they ignore the
semantic of contextual objects and the prior knowledge of their correlations with the social relationships. Besides, the interaction between the contextual objects and the persons of interest is also oversimplified.

Different from these works, we formulate a Graph Reasoning Model (GRM) that unifies the prior knowledge with deep neural networks for handling the task of social relationship recognition. Specifically, we first organize the prior knowledge as a structured graph that describes the co-occurrences of social relationships and semantic objects in the scene. The GRM then initializes the graph nodes with corresponding semantic regions, and employ a Gated Graph Neural Network (GGNN) [Li et al., 2015b] to propagate model message through the graph to learn node-level features and to explore the interaction between persons of interest and the contextual objects. As some contextual objects are key to distinguish different social relationships while some are non-informative or even interferential, we further introduce a graph attention mechanism to adaptively select the most discriminative nodes for recognition by measuring the importance of each node. In this way, the GRM can also provide an interpretable way to improve social relationship recognition by explicitly reasoning about relevant objects that provide key contextual cues.

In summary, the contributions can be concluded to three-fold. 1) We propose an end-to-end trainable and interpretable Graph Reasoning Model (GRM) that unifies high-level knowledge graph with deep neural networks to facilitate social relationship recognition. To the best of our knowledge, our model is among the first to advance knowledge graph for this task. 2) We introduce a novel graph attention mechanism that explicitly reasons key contextual cues for better social relationship understanding. 3) We conduct extensive experiments on the large-scale People in Social Context (PISC) [Zhang et al., 2015] and the People in Photo Album Relation (PIPA-Relation) [Sun et al., 2017] datasets and demonstrate the superiority of our methods over the existing state-of-the-art methods.

2 Related Work

We review the related works in terms of two research streams: social relationship recognition and graph neural network.

2.1 Social Relationship Recognition

Social relationships form the basic information of social network [Li et al., 2015a]. In computer vision community, social information has been considered as supplementary cues to improve various tasks including multi-target tracking [Choi and Savarese, 2012], human trajectory prediction [Alahi et al., 2016] and group activity analysis [Lan et al., 2012; Deng et al., 2016]. For instance, [Alahi et al., 2016] implicitly induce social constraint to predict human trajectories that fit social common sense rules. [Lan et al., 2012] and [Deng et al., 2016] exploit social roles and individual relation to aid group activity recognition, respectively.

The aforementioned works implicitly embed social information to aid inference, and there are also some efforts dedicated to directly predicting social roles and relationships. As a pioneering work, [Wang et al., 2010] use familial social relationships as context to recognize kinship relations between pairs of people. To capture visual patterns exhibited in these relationships, facial appearance, attributes and landmarks are extensively explored for kinship recognition and verification [Dibeklioglu et al., 2013; Xia et al., 2012; Chen et al., 2012]. To generalize to general social relationship recognition and enable this research, recent works [Li et al., 2017] and [Sun et al., 2017] construct large-scale datasets and employ deep models to directly predict social relationships from raw image input. Concretely, [Sun et al., 2017] build on Bugental’s domain-based theory [Bugental, 2000] which partitions social life into 5 domains. They derive 16 social relationships based on these domains and extend the People in Photo Album (PIPA) dataset [Zhang et al., 2015] with 26,915 relationship annotations between person pairs. A simple two-stream model is proposed for social domain/relationship recognition. Concurrently, [Li et al., 2017] follow relational models theory [Fiske, 1992] to define a hierarchical social relationship categories, which involve 3 coarse-level and 6 fine-level relationships. They also build a People in Social Context (PISC) dataset that consists of 22,670 images with 76,568 manually annotated person pairs from 9 types of social relationships, and propose a dual-glance model that exploits category-agnostic proposals as contextual cues to aid recognizing the social relationships.

2.2 Knowledge Representation

Representing extr/prior knowledge in the form of graph structure [Schlichtkrull et al., 2017; Lin et al., 2017] and incorporating this structure for visual reasoning [Malisiewicz and Efros, 2009; Zhu et al., 2014; Teney et al., 2017] has received increasingly attention. For example, Malisiewicz et al. [Malisiewicz and Efros, 2009] build a large graph, with the nodes referring to object instances and the edge corresponding to associations to types between nodes, to represent and reason about object identities and their contextual relationships. Lin et al. [Lin et al., 2017] introduce And-Or graphs for task representation, which can effectively regularize predictable semantic space for effective training. These methods usually involve hand-crafted features and manually-defined rules.

Recently, more works are dedicated to explore message propagation by learnable neural networks like [Chen et al., 2018; Wang et al., 2017] or neural network variants like Graph LSTM [Li et al., 2016; 2017] and Graph CNN [Duvenaud et al., 2015; Schlichtkrull et al., 2017; Kipf and Welling, 2016]. [Wang et al., 2017] exploits LSTM network to capture label dependencies by remembering previous information step by step. [Li et al., 2016] proposes a Graph LSTM network that propagates message through super-pixels over different level to model their contextual dependencies. Gated Graph Neural Network (GGNN) [Li et al., 2015a] is a fully differential recurrent neural network architecture for handling graph-structured data, which iteratively propagate node message through the graph to learn node-level or graph-level representation. Several works have successfully developed GGNN variants for various vision tasks including 3DGGN for RGBD semantic segmentation [Qi et al., 2017], and GSNN for multi-label image recognition [Marino et al., 2016]. For example, GSNN learns knowledge representation and concatenates it with image feature to improve multi-
Figure 2: An overall pipeline of our proposed graph reasoning model. Given an image and a person pair, the GRM initializes the relationship nodes with features extracted from regions of person pair and the object nodes with features extracted from corresponding semantic regions in the image. Then it employs the GGNN to propagate node message through the graph to compute node-level features, and introduces a graph attention mechanism to attend to the most discriminative object nodes for recognition by measuring their importance to each relationship node.

label classification. Different from these works, we apply the GGNN to encode the prior knowledge graph and explore the interaction between the person pair of interest and contextual objects to address a newly-raised problem, i.e., social relationship recognition.

3 Graph Reasoning Model

The graph refers to an organization of the correlations between social relationships and semantic objects in the scene, with nodes representing the social relationships and semantic objects, and edges representing the probabilities of their co-occurrences. Given an image and a person pair of interest from the image, the GRM follows [Li et al., 2017] to extract features from the regions of the person pair and initializes the relationship nodes with these features. And it uses a pre-trained Faster-RCNN detector [Ren et al., 2015] to search the semantic objects in the image and extract their features to initialize the corresponding object nodes. Then, the GRM employs the GGNN [Li et al., 2015b] to propagate node message through the graph to fully explore the interaction of the persons with the contextual objects, and adopts the graph attention mechanism to adaptively select the most informative object nodes to facilitate recognition by measuring the importance of each object node. Figure 2 presents an illustration of the GRM.

3.1 Knowledge Graph Propogation

GGNN [Li et al., 2015b] is an end-to-end trainable network architecture that can learn features for arbitrary graph-structured data by iteratively updating node representation in a recurrent fashion. Formally, the input is a graph represented as $\mathcal{G} = \{V, A\}$, in which $V$ is the node set and $A$ is the adjacency matrix denoting the connections among these nodes. For each node $v \in V$, it has a hidden state $h_v^t$, at timestep $t$, and the hidden state at $t = 0$ is initialized by input feature vectors $x_v$ that depends on the task at hand. At each timestep, we update the representation of each node based on its history state and the message sent by its neighbors. Here, we follow the computational process of the GGNN to learn the propagation mechanism.

The graph contains two types of nodes, i.e., social relationship and object nodes, and we initialize their input feature with different contents from the image. Since the regions of the person pair maintain the basic information for recognition, we extract features from these regions to serve as the input features for the social relationship nodes. Similar to [Li et al., 2017], we first crop three regions, among which one covers the union of the two persons and the other two contain the two persons respectively, and extract three feature vectors from these three regions. These feature vectors, together with the position information encoding the geometry feature of the two persons, are concatenated and fed into a fully connected layer to produce a $d$-dimension feature vector $f_v \in \mathbb{R}^d$, $f_v$ is then served as the input features for all the social relationship nodes. For the object nodes, we detect the object regions in the image using a pre-trained detector and extract features from these detected regions to initialize the nodes that refer to corresponding categories. As social relationship datasets, e.g., PISC [Li et al., 2017] and PIPA-Relation [Sun et al., 2017], do not provide object category and their position annotations, the detector cannot be directly trained on these datasets. Fortunately, COCO [Lin et al., 2014] is a large-scale dataset for object detection, and it covers 80 common categories of objects that occur frequently in our daily life; thus we get a faster RCNN detector [Ren et al., 2015] trained on this dataset for the collection of semantic objects. And we regard the object with a detected score higher than a pre-defined threshold $\epsilon_1$ as the semantic objects existed in the given image. For the node referring to the object $o$ that is detected in the image, its input feature is initialized by the features extracted from the corresponding region $f_o \in \mathbb{R}^d$, and otherwise, it is initialized by a $d$-dimension zero vector. In addition, we use a one-hot vector to explicitly distinguish the two node types, with $[1, 0]$ and $[0, 1]$ denoting the social relationship and object nodes, respectively, and concatenate them with the corresponding features to initialize the hidden
state at timestep \( t = 0 \), expressed as
\[
\begin{align*}
\mathbf{h}_v^0 &= \begin{cases} 
[1, 0], f_v & \text{if } v \text{ refers to a relationship} \\
[0, 1], f_v & \text{if } v \text{ refers to category } a \text{ that is detected} \\
[0, 0], 0 \text{ } & \text{otherwise}
\end{cases}
\end{align*}
\]
where \( \mathbf{0}_d \) is a zero vector with dimension of \( d \). At each timestep, the nodes first aggregate message from its neighbors, expressed as
\[
\mathbf{a}_v^t = \mathbf{A}_v \mathbf{h}_v^{t-1} + \mathbf{b},
\]
where \( \mathbf{A}_v \) is the sub-matrix of \( \mathbf{A} \) that denotes the connection of node \( v \) with its neighbors. Then, the model incorporates information from the other nodes and from the previous timestep to update each node’s hidden state through a gating mechanism similar to the Gated Recurrent Unit (Cho et al., 2014; Li et al., 2015b), formulated as
\[
\begin{align*}
\mathbf{z}_v^t &= \sigma (\mathbf{W}_a \mathbf{a}_v^t + \mathbf{U} \mathbf{h}_v^{t-1}) \\
\mathbf{r}_v^t &= \sigma (\mathbf{W}_r \mathbf{a}_v^t + \mathbf{U} \mathbf{h}_v^{t-1}) \\
\mathbf{h}_v^t &= \tanh (\mathbf{W} \mathbf{a}_v^t + \mathbf{U} (\mathbf{r}_v^t \odot \mathbf{h}_v^{t-1})) \\
\tilde{\mathbf{h}}_v^t &= (1 - \mathbf{z}_v^t) \odot \mathbf{h}_v^{t-1} + \mathbf{z}_v^t \odot \tilde{\mathbf{h}}_v^t,
\end{align*}
\]
where \( \sigma \) and tanh are the logistic sigmoid and hyperbolic tangent functions, respectively, and \( \odot \) denotes the element-wise multiplication operation. In this way, each node can aggregate information from its neighbors while transfer its own message to its neighbors, enabling the interaction among all nodes. An example propagation process is illustrated in Figure 2. After \( T \) interactions, the node message has propagated through the graph, and we can get the final hidden state for each node, i.e., \( \{ \mathbf{h}_1^T, \mathbf{h}_2^T, \ldots, \mathbf{h}_{|V|}^T \} \). Similar to [Li et al., 2015b], we employ an output network that is implemented by a fully-connected layer, to compute node-level feature, expressed by
\[
\mathbf{o}_v = o ( [ \mathbf{h}_1^T, \mathbf{x}_v ] ), v = 1, 2, \ldots, |V|.
\]

### 3.2 Graph Attention Mechanism

After computing features for each node, we can directly aggregate them for recognition. However, we found that some contextual objects play key roles to distinguish different relationships while some objects are non-informative or even incur interference. For example, the object “desk” co-occurs frequently with most social relationships; thus it can hardly provide information for recognition. To address this issue, we introduce a novel graph attention mechanism that adaptively reasons about the most relevant contextual objects according to the graph structure. For each social relationship and neighbor object pair, the mechanism takes their last hidden states as input and computes a score denoting the importance of this object to the relationship. We describe this module formally in the following.

For illustration convenience, we denote the social relationship nodes as \( \{ r_1, r_2, \ldots, r_M \} \) and the object node as \( \{ o_1, o_2, \ldots, o_N \} \), where \( M \) and \( N \) is the number of the two type nodes, respectively. And their hidden states can be denoted as \( \{ \mathbf{h}_1^r, \mathbf{h}_2^r, \ldots, \mathbf{h}_M^r \} \) and \( \{ \mathbf{h}_1^o, \mathbf{h}_2^o, \ldots, \mathbf{h}_N^o \} \) accordingly. Given a relationship \( r_i \) and an object \( o_j \), we first fuse their hidden states using low-rank bilinear pooling method [Kim et al., 2016]
\[
\mathbf{h}_{ij} = \tanh (\mathbf{U}^a \mathbf{h}_{r_i}) \odot \tanh (\mathbf{V}^a \mathbf{h}_{o_j}),
\]
where \( \mathbf{U}^a \) and \( \mathbf{V}^a \) are the learned parameter matrices. Then, we can compute the attention coefficient
\[
e_{ij} = a (\mathbf{h}_{ij} )
\]
that indicates the importance of object node \( o_j \) to relationship node \( r_i \). \( a \) is the attentional mechanism that is used to estimate the attention coefficient and it is implemented by a fully-connected layer. The model allows to attend on every object nodes but such mechanism ignores the graph structure. In this work, we inject the structure information into the attention mechanism by only computing the attention coefficient \( e_{ij} \) for object nodes \( j \in N_i \), where \( N_i \) is the neighbor set of node \( i \) in the graph. The coefficients are then normalized to \( (0, 1) \) using a sigmoid function
\[
\alpha_{ij} = \sigma (e_{ij} )
\]
where \( \alpha_{ij} \) is assigned to zero if object node \( j \) does not belong to \( N_i \).

Once obtained, we utilize the normalized attention coefficients to weight the output features of the corresponding nodes and aggregate them for final recognition. Specifically, we denote the output features of the social relationship nodes and the object nodes as \( \{ \mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_M \} \) and \( \{ \mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_N \} \). For relationship \( r_i \), we concatenate the features of its own node and the weighted features of context nodes to serve as its final features, that is
\[
\mathbf{f}_i = [ \mathbf{o}_i; \alpha_{i_1} \mathbf{o}_{i_1}, \alpha_{i_2} \mathbf{o}_{i_2}, \ldots, \alpha_{i_N} \mathbf{o}_{i_N} ].
\]
Then the feature vector \( \mathbf{f}_i \) is fed into a simple fully-connected layer to compute a score
\[
\mathbf{s}_i = \mathbf{W} \mathbf{f}_i + \mathbf{b}
\]
that indicates how likely the person pair is of social relationship \( r_i \). The process is repeated for all the relationship nodes to compute the score vector \( \mathbf{s} = \{ s_1, s_2, \ldots, s_M \} \).

### 3.3 Optimization

We employ the cross entropy loss as our objective function. Suppose there are \( K \) training samples, and each sample is annotated with a label \( y_k \). Given the predicted probability vector \( \mathbf{p}_k \)
\[
p_i^k = \frac{\exp (s_i^k)}{\sum_{i=1}^{M} \exp (s_i^k) }, i = 1, 2, \ldots, M,
\]
the loss function is expressed as
\[
\mathcal{L} = - \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{M} \mathbf{1} (y_k = i) \log p_i^k,
\]
where \( \mathbf{1}(\cdot) \) is the indicator function whose value is 1 when the expression is true, and 0 otherwise.
4 Experiments

4.1 Knowledge Graph Building

The knowledge graph describes the co-occurrences of social relationships and semantic objects in the scene. Building such a graph requires annotations of both social relationships for person pairs and objects existed in the images. However, there is no dataset that meets these conditions. As discussed above, detector trained on the COCO [Lin et al., 2014] dataset can well detect semantic objects that occur frequently in our daily life. Thus, we also use the faster RCNN [Ren et al., 2015] detector trained on COCO dataset to detect objects of the image on the social relationship dataset. We regard the detected object with a confidence score higher than a threshold $\epsilon_2$ as the semantic objects in the given image. Here, we utilize a high threshold to avoid incurring too much false positive samples (i.e., $\epsilon_2 = 0.7$). In this way, we can obtain several pseudo object labels for each image. We then count the frequency of the co-occurrences of each relationship-object pair over the whole training set. All scores are normalized to $[0, 1]$ and the edge with a small normalized score is pruned. Despite the mistakenly predicted labels, the obtained knowledge graph can basically describe the correlation between relationship and object by counting their co-occurrence over a quite large dataset. Besides, by explicitly reasoning about the most important nodes, our GRM can leverage the noise knowledge graph to aid recognition. Figure 3 illustrate an example knowledge graph for the PISC dataset.

4.2 Experiment Setting

Dataset. We evaluate the GRM and existing competing methods on the large-scale People in Social Context (PISC) [Li et al., 2017] and the People in Photo Album Relation (PIPA-Relation) [Sun et al., 2017] datasets. The PISC dataset contains 22,670 images and involves two-level relationship recognition tasks: 1) Coarse-level relationship focuses on three categories of relationship, i.e., No Relation, Intimate Relation, and Non-Intimate Relation; 2) Fine-level relationship focuses on six finer categories of relationship, i.e., Friend, Family, Couple, Professional, Commercial and No relation. For fair comparisons, we follow the standard train/val/test split released by [Li et al., 2017] to train and evaluate our GRM. Specifically, for the coarse level relationship, it divides the dataset into a training set of 13,142 images and 49,017 relationship instances, a validation set of 4,000 images and 14,536 instances and a test set of 4,000 images and 15,497 instances. For the fine level relationship, the train/val/test set consist of 16,828 images and 55,400 instances, 500 images and 1,505 instances, 1,250 images and 3,961 instances, respectively. The PIPA-Relation dataset partitions social life into 5 domains and derives 16 social relations based on these domains. Still, we focus on recognizing 16 relationships in the experiment. As suggested in [Sun et al., 2017], this dataset contains 13,729 person pairs for training, 709 for validation, and 5,106 for test. Serval examples of the samples and their relationship annotations from both two datasets are shown in Figure 4.

Implementation details. For the GGNN propagation model, the dimension of the hidden state is set as 4,098 and that of the output feature is set as 512. The iteration time $T$ is set as 3. During training, all components of the model are trained with SGD except that the GGNN is trained with ADAM following [Marino et al., 2016]. Similar to [Li et al., 2017], we utilize the widely used ResNet-101 [He et al., 2016] and VGG-16 [Simonyan and Zisserman, 2014] to extract features for person regions and semantic object regions respectively.

4.3 Comparisons with State-of-the-Art Methods

We compare our proposed GRM with existing state-of-the-art methods on both PISC and PIPA-Relation datasets.

Performance on the PISC dataset

We follow [Li et al., 2017] to compare our GRM with baseline and existing state-of-the-art methods on the PISC dataset. Concretely, the competing methods are listed as follow:

- **Union CNN** generalizes the model [Lu et al., 2016], which predicates general relations, to this task. It feeds the union region of the person pair of interest to a single CNN for recognition.
- **Pair CNN** [Li et al., 2017] consists of two identical CNNs that share weights to extract features for cropped image patches for the two individuals and concatenate them for recognition.
- **Pair CNN + BBox + Union** [Li et al., 2017] aggregates features from pair CNN, union CNN and BBox that encode the
Table 1: Comparisons of our GRM with existing state-of-the-art and baseline methods on the PISC dataset. We present the per-class recall for each relationships and the mAP over all relationships (in %).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Coarse relationships</th>
<th>Fine relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Intimate</td>
<td>Non-Intimate</td>
</tr>
<tr>
<td>Union CNN [Lu et al., 2016]</td>
<td>72.1</td>
<td>81.8</td>
</tr>
<tr>
<td>Pair CNN [Li et al., 2017]</td>
<td>70.3</td>
<td>80.5</td>
</tr>
<tr>
<td>Pair CNN + BBox + Union [Li et al., 2017]</td>
<td>71.1</td>
<td>81.2</td>
</tr>
<tr>
<td>Pair CNN + BBox + Global [Li et al., 2017]</td>
<td>70.5</td>
<td>80.0</td>
</tr>
<tr>
<td>Dual-glance [Li et al., 2017]</td>
<td>73.1</td>
<td>84.2</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>81.7</strong></td>
<td><strong>73.4</strong></td>
</tr>
</tbody>
</table>

Table 2: Comparison of the accuracy (in %) of our proposed GRM with existing methods on the PIPA-Relation dataset.

<table>
<thead>
<tr>
<th>Methods</th>
<th>accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Two stream CNN [Sun et al., 2017]</td>
<td>57.2</td>
</tr>
<tr>
<td>Dual-Glance [Li et al., 2017]</td>
<td>59.6</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>62.3</strong></td>
</tr>
</tbody>
</table>

Performance on the PIPA-Relation dataset

On this dataset, we compare our proposed GRM with two existing methods, i.e., Two stream CNN [Sun et al., 2017] that has reported the results on this dataset and Dual-Glance [Li et al., 2017] that performs best among existing methods on the PISC dataset (see Table 1). As dual-glance does not present their results on this dataset, we strictly follow [Li et al., 2017] to implement it for evaluation. The results are presented in Table 2. Still, our GRM significantly outperforms previous methods. Specifically, it achieves an accuracy of 62.3%, beating the previous best method 2.7%.

4.4 Ablation Study

Significance of knowledge graph

The core of our proposed GRM is the introduction of the knowledge graph as extra guidance. To better verify its effectiveness, we conduct an experiment that randomly initializes the adjacency matrix of the graph and re-train the model in a similar way on the PISC dataset. As shown in Table 3, the accuracies drop from 82.8% to 81.4% on the coarse-level task and from 68.7% to 63.5% on the fine-level task. These obvious performance drops clearly suggest incorporating the prior knowledge can significantly improve social relationship recognition.

<table>
<thead>
<tr>
<th>Methods</th>
<th>coarse-level</th>
<th>fine level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random matrix</td>
<td>81.4</td>
<td>63.5</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>82.8</strong></td>
<td><strong>68.7</strong></td>
</tr>
</tbody>
</table>

Table 3: Comparison of the mAP (in %) of our GRM that initialize the adjacency matrix by scores counting on the training set and by random scores.
### Analysis on the graph attention mechanism

Graph attention mechanism is a crucial module of our GRM that can reason about the most relevant contextual objects. Here, we further implement two baseline methods for comparison to demonstrate its benefit. First, we simply remove this module and directly concatenate features of the relationship nodes and all object nodes for recognition. As shown in Table 4, it suffers from an obvious drop in mAP, especially for the fine-level social relationship recognition that is more challenge. Second, we replace the learnt attention coefficients with randomly-selected scores and retrain the model in an identical way. It shows that the performance is even worse than that using features of all nodes as it may attend to nodes that refer to non-informative or interferential objects.

<table>
<thead>
<tr>
<th>Methods</th>
<th>coarse-level</th>
<th>fine level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random score</td>
<td>82.0</td>
<td>66.8</td>
</tr>
<tr>
<td>Ours w/o attention</td>
<td>82.6</td>
<td>67.8</td>
</tr>
<tr>
<td>Ours</td>
<td>82.8</td>
<td>68.7</td>
</tr>
</tbody>
</table>

Table 4: Comparison of the mAP (in %) of ours without attention, ours with random score and our full model.

### Analysis on the semantic object detector

The detector is utilized to search the semantic objects with confidence scores higher than threshold $\epsilon_1$. Obviously, a small threshold may incur false detected objects while a high threshold may lead to contextual cue missing. Here, we conduct experiments with different threshold values for selecting an optimal threshold. As shown in Table 5, we find that a relative threshold (i.e., 0.3) lead to best results despite disturbance incurred by the false detected objects. One possible reason is that setting $\epsilon_1$ as 0.3 can recall most contextual objects and the attentional mechanism can well suppress this disturbance, thus leading to better performance.

<table>
<thead>
<tr>
<th>$\epsilon_1$</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
</tr>
</thead>
<tbody>
<tr>
<td>mAP</td>
<td>67.8</td>
<td>68.7</td>
<td>67.2</td>
<td>67.3</td>
</tr>
</tbody>
</table>

Table 5: Comparison of the mAP (in %) of our GRM model using different $\epsilon_1$.

### 4.5 Qualitative Evaluation

In this subsection, we present some examples to illustrate how our GRM recognizes social relationships in Figure 5. Considering the first example, the semantic objects including desk, laptop, cup etc., are first detected and utilized to initialize the corresponding object nodes in the graph. However, the objects like desk and cup, co-occur frequently with most social relationships, and thus they can hardly provide informative message to distinguish these relationships. In contrast, office supplies like laptop provide strong evidence for the “professional” relationship. Thus, the attention mechanism assigns higher scores to these two nodes, and perform prediction successfully. For the second example, our GRM attends to the bowl and pizza that are the key cues for recognizing “friend” relationship.

![Figure 5: Two examples of how our GRM recognize social relationships. We visualize the original image, the regions of two persons (indicated by the red boxes), the regions of detected objects (indicated by green boxes) and the ground truth relationships in the left and predicted relationship and nodes referring to the detected objects in the right. The object nodes with top-2 highest scores are highlighted in orange. Best view in color.](image)

### 5 Conclusion

In this work, we propose a Graph Reasoning Model (GRM) that incorporates common sense knowledge of the correlation between social relationship and semantic contextual cues in the scene into the deep neural network to address the take of social relationship recognition. Specifically, the GRM consists of a propagation model that propagates node message through the graph to explore the interaction between the person pair of interest and contextual objects, and a graph attention module that measures the importance of each node to adaptively select the most discriminative objects to aid recognition. Extensive experiments on two large-scale benchmarks (i.e., PISC and PIPA-Relation) demonstrate the superiority of the proposed GRM over existing state-of-the-art methods.

### References

[Alahi et al., 2016] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In CVPR.


