Towards Knowledge-Augmented Visual Question Answering

Maryam Ziaeefard
McGill University
Montréal, Canada

Freddy Lécué
Inria, Sophia Antipolis, France
Thales, Montréal, Canada

Abstract

Visual Question Answering (VQA) remains algorithmically challenging while it is effortless for humans. Humans combine visual observations with general and commonsense knowledge to answer a question about a given image. In this paper, we address the problem of incorporating general knowledge into VQA models while leveraging the visual information. We propose a model that captures the interactions between objects in a visual scene and entities in an external knowledge source. Our model is a graph-based approach that combines scene graphs with concept graphs, which learns a question-adaptive graph representation of related knowledge instances. We use Graph Attention Networks to set higher importance to key knowledge instances that are mostly relevant to each question. We exploit ConceptNet as the source of general knowledge and evaluate the performance of our model on the challenging OK-VQA dataset. Our code will be available at `https://github.com/ZiaMaryam/KVQA`

1 Introduction

The task of Visual Question Answering (VQA) (Antol et al., 2015) was introduced to bridge the gap between natural language processing and image understanding applications. Most VQA methods focus on the visual aspect of the VQA task and predict the answer by combining the question and image representations. However, visual-based approaches are practical when no insights beyond the visual content is required.

Incorporating external knowledge into VQA models combines visual observations with external knowledge (Garderes et al., 2020). Organizing the external knowledge and storing them in a structured database, such as a Knowledge Bases (KB), have become important resources for representing the general knowledge. A typical KB could be represented as a graph, which is usually a collection of triples also known as facts. The triples specify that two entities (nodes) are connected by a particular relation (edge), e.g., `(Shakespeare, writerOf, Hamlet)` where `Shakespeare` and `Hamlet` are entities and `writerOf` is the relation between the entities. Example of such graph structures are YAGO (Suchanek et al., 2007), DBpedia (Auer et al., 2007), NELL (Carlson et al., 2010), Freebase (Bollacker et al., 2008), and the Google Knowledge Graph (Steiner et al., 2012).

In recent years, a significant amount of research has been devoted to visual-based VQA while external knowledge remains unavailable. To address this challenge, recent work (Wang et al., 2017; Wang et al., 2018; Shah et al., 2019) generated new datasets for the purpose of evaluating the performance of VQA algorithms capable of answering higher knowledge level. We use OK-VQA dataset (Marino et al., 2019) for our experiments since it requires handling outside knowledge. In addition to understanding the question and the image, the model needs to learn what knowledge is necessary to answer the OK-VQA questions since no ground-truth facts are provided with the dataset.

In this work, our main motivation is to develop a technique to integrate general knowledge while leveraging the relations between objects in the visual scene and entities in the knowledge graph. Exploring

Proceedings of the 28th International Conference on Computational Linguistics, pages 1863–1873
Barcelona, Spain (Online), December 8-13, 2020

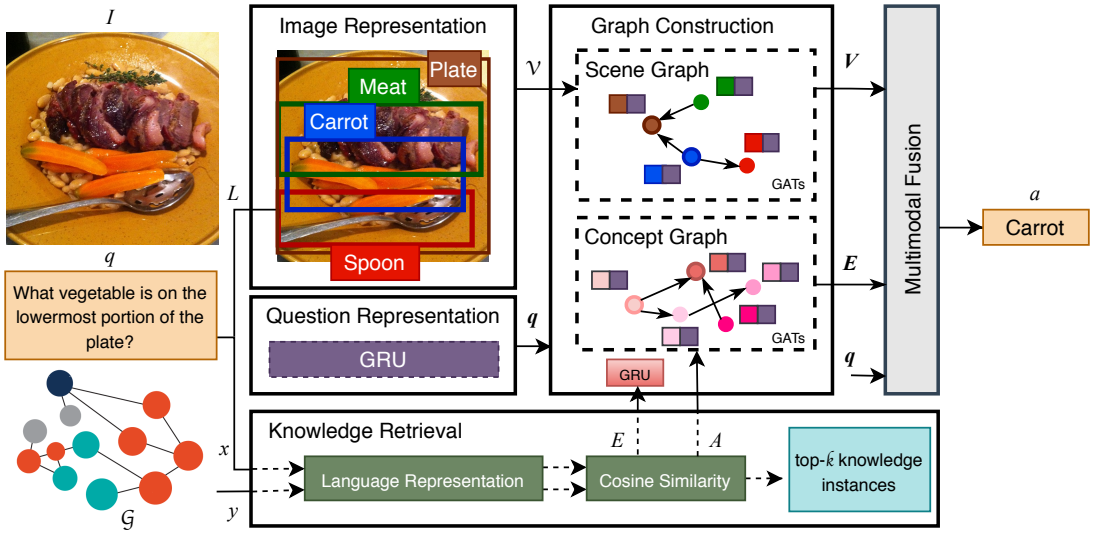


Figure 1: Model architecture of the proposed approach.

OK-VQA questions, we found questions that need information from visual concepts and beyond. For instance given question *"What vegetable is on the lowermost portion of the plate?"* (cf. Fig. 1), we need to learn what objects are located on the lowermost plate as well as what object among them is vegetable to retrieve the correct answer *"carrot"*. In order to capture this type of information, we need to represent the dynamics and interactions between different objects in an image and entities from a relevant external knowledge source.

The outline of our proposed model is depicted in Fig. 1. As shown in the figure, we first compute embeddings for the question and objects detected in the input image. We then construct a Scene graph (Li et al., 2019) which is a relation representation of visual concepts, where the nodes are the detected objects (e.g., plate, carrot) and the edges are the relationships between objects (e.g., located). To incorporate the general knowledge, we design a knowledge retrieval module based on sentence-level similarity scores and pass the retrieved knowledge entities to a Concept graph. We use Graph Attention Networks (GATs) (Velickovic et al., 2018) to assign higher weights to those objects and knowledge instances that are mostly relevant to each question. The outputs of these three steps (question embedding, Scene graph, and Concept graph) are fused to represent joint language-vision-knowledge embeddings and then fed to a classifier to predict the answer.

In summary, the main contributions of our work are: 1) Novel methodology to incorporate general knowledge to VQA models (Figure 1). Unlike existing models, we avoid the step of query construction and do not use ground-truth facts which makes it feasible to incorporate any knowledge resources to our model, 2) We use sentence level embeddings to retrieve knowledge instances rather than word level embeddings which capture the semantic context of the questions and knowledge instances (Section 4.1), 3) We develop Concept graphs using GATs which operate on neighboring to attend to key knowledge instances (Section 4.2), 4) We use both Scene graphs and Concept graphs to capture the relations between objects and entities.

2 Related Work

2.1 Scene Graph Representation for Visual Question Answering

Teney et al. (2017) propose a model to build graphs over the scene objects and over the question words. Each object in the scene corresponds to a node in the fully-connected scene graph, with each edge representing the relative position of the objects in the image. They evaluate their proposed model over a clip-art dataset. Shi et al. (2019) propose a model using scene graphs to represent objects as nodes and the pairwise relationships as edges for both abstract scenes and real images. The explicit relations

between objects are not taken into account in their work. Li et al. (2019) propose a VQA model that encodes each image into a graph and represents explicit and implicit inter-object relations using a graph attention mechanism. The graph for learning implicit relation is fully-connected and include relative geometry features. The explicit relation contains visual relationship extracted from Visual Genome dataset (Krishna et al., 2017). Norcliffe-Brown et al. (2018) propose a graph learner module using Spatial graph convolutions (Monti et al., 2017). Their model learns a graph representation of the input image that is conditioned on the question, and models the relevant interactions between objects in the scene.

2.2 Knowledge-based Visual Question Answering

Knowledge-based VQA is still relatively unexplored compared to visual-based VQA. Some methods (Wang et al., 2017; Wang et al., 2018) convert an input question into fixed templates to query an external KB and processes the returned knowledge to form the final answer. Narasimhan and Schwing (2018) and Narasimhan et al. (2018) propose models that retrieves most relevant facts to a question-answer pair based on word level GloVe embeddings. However, they require ground-truth facts per question to classify the relation that a given question refers to. Marino et al. (2019) propose ArticleNet for OK-VQA questions. ArticleNet retrieves articles from Wikipedia for each question-image pair and then predict whether and where the ground-truth answers appear in the article. While this method handles external knowledge resources, it requires an expensive hand-crafted process to extract Wikipedia articles including collecting possible search queries for each question-image pair, using the Wikipedia search API to get the top retrieved article for each query and extracting a subset of each article that is most relevant for the query.

3 Problem Formulation

Here is the problem definition of the knowledge-based VQA task: Given a question $q \in \mathcal{Q}$ grounded in an image $I \in \mathcal{I}$ and a knowledge base $\mathcal{G}$, the goal is to predict a meaningful answer $a \in \mathcal{A}$. Let Θ be the parameters of the model p that needs to be trained. Therefore, the predicted answer $\hat{a}$ of our model is:

$$\hat{a} = \arg\max_{a \in \mathcal{A}} \; p_{\Theta}(a|I, q, \mathcal{G}) \tag{1}$$

In order to retrieve the correct answer, we aim to learn a joint representation $z \in R^{d_z}$ of q, I, and $\mathcal{G}$ such that:

$$a^* = \hat{a} = \arg\max_{a \in \mathcal{A}} \; p_{\Theta}(a|z) \tag{2}$$

where a^* is the ground-truth answer. d_z is a hyperparameter that represents the dimension of the joint space z. d_z is selected based on a trade-off between the capability of the representation and the computational cost.

4 Our Approach

Our proposed approach is outlined in Fig. 1 which consists of Image Representation, Question Representation, Knowledge Retrieval, Graph Construction, and Multimodal Fusion modules. We use pre-trained Faster R-CNN features (Anderson et al., 2017) to extract visual information $\boldsymbol{\mathcal{V}} = \{v_i \,|\, i = 1, ..., n_v\}$, where each object i is associated with a visual feature vector $\boldsymbol{v_i} \in \mathbb{R}^{d_v}$ and bounding-box coordinates. For the Question Representation, we use a bidirectional RNN (GRU) and perform self attention on the sequence of RNN hidden states to generate question embeddings $\boldsymbol{q} \in \mathbb{R}^{d_q}$. The following sub-sections explain the Knowledge Retrieval, Graph Construction, and Multimodal Fusion modules in detail.

4.1 Knowledge Retrieval

Given a question-image pair (q, I), the knowledge retrieval module outputs a set of knowledge entities E and an adjacency matrix A to capture the relation between the entities. These outputs are obtained in four steps: i) generating a question-image instance; ii) generating knowledge instances; iii) instance representation; and iv) knowledge instance ranking. We discuss each of these steps in the following.

Generating question-image instance: We first obtain a question-image instance x by concatenating tokens of the question $q = \{w_i \mid i = 1, ..., n_t\}$ and object labels $L = \{l_i \mid i = 1, ..., n_v\}$ detected in the image, where n_t is the number of tokens. For instance given the question *"What vegetable is on the lowermost portion of the plate?"* and object labels *"spoon, carrot, meat, plate"* detected from the given image, we obtain $x =$" *what vegetable is on the lowermost portion of the plate spoon carrot meat plate*".

Generating knowledge instances: We use ConceptNet (Li et al., 2016) as the source of general knowledge which consists of a set of facts denoted as $\mathcal{G} = \{f_i \mid i = 1, ..., n_{kb}\}$, where n_{kb} is the number of facts. In ConceptNet, a fact f_i is represented as a triple of the form $f_i = (r, h, t)$, where r is a relation between two entities, h is a head entity, and t is a tail entity. We select 20 most frequent relations among 34 relations in ConceptNet, i.e., $r \in \mathcal{R} = \{$*RelatedTo, FormOf, IsA, PartOf, HasA, UsedFor, CapableOf, AtLocation, Causes, HasSubevent, HasFirstSubevent, HasProperty, HasPrerequisite, MotivatedByGoal, DerivedFrom, DefinedAs, SimilarTo, CausesDesire, MadeOf, Desires*$\}$.

We pre-process each triple and convert it to a semi-phrase to create knowledge instances y. For example, the triple *(IsA, carrot, orange vegetable)*, is converted to *"carrot is orange vegetable"* after the pre-processing step.

Instance embedding: We propose to use pre-trained language representation (LR) models to embed the question-image instance x and knowledge instances $\boldsymbol{y}$ generated in the previous steps. While the pre-trained LR model such as BERT (Devlin et al., 2018) is an emerging direction, there is little work on its fusion with the external knowledge in VQA tasks. We are particularly interested in LR models that are trained for sentence similarity tasks since we use the generated embedding vectors to compute a sentence level similarity between the question-image instance and each knowledge instances.

We use two state-of-the-art sentence embedding models, i.e, Universal Sentence Encoder (USE) (Cer et al., 2018) and Sentence-BERT (SBERT) (Reimers and Gurevych, 2019) to generate the question-image representation $\boldsymbol{x} \in \mathbb{R}^{d_s}$ and knowledge instance representations $\boldsymbol{y} \in \mathbb{R}^{n_{kb} \times d_s}$. USE presents two models for producing sentence embeddings. We use the transformer-based model since it works best in our model. Transformer-based USE uses an attention mechanism to compute context aware representations of words in a sentence that take into account both the ordering and identity of all the other words. On the other hand, SBERT is a modification of the pre-trained BERT network that uses Siamese and triplet network structures to derive semantically meaningful sentence embeddings.

Knowledge instance ranking: We obtain a similarity score per knowledge instance by computing the cosine similarity of the question-image representation $\boldsymbol{x}$ and knowledge instance representations $\boldsymbol{y}$. We rank the knowledge instances y based on the similarity scores and retrieve the top-k knowledge instances and the corresponding set of k facts, F_k, for each question.

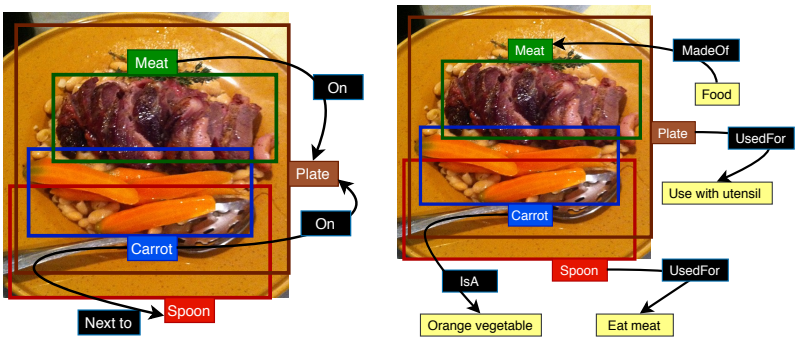
For every question, we extract unique entities $E = \{e_i \mid i = 1, ..., 2k\}$ from the facts F_k (at most 2 unique entities per fact) and construct an adjacency matrix A that shows the relation between the entities in each fact. The size of the matrix A is $2k \times 2k$. The value of an entry A_{ij} is between 0 and 20 depending on the type of the relation r between entity i and entity j (0 if entity i and entity j do not belong to the same fact).

4.2 Graph Construction

In order to capture both inter-object and knowledge entity relations, we construct a scene graph using objects detected in the image and their relations, as well as a concept graph that represents the explicit relations between entities. The details of the scene and concept graph construction are as follows.

Scene graph construction: We first construct a scene graph G_s, where nodes are a set of objects detected in the image and edges show the relation between objects. Given two object regions, the goal of the scene graph construction is to determine which relation exists between these two regions. We use pre-trained scene graphs generated by (Li et al., 2019) which includes 14 explicit relations from Visual Genome dataset (Krishna et al., 2017), with an additional no-relation class. An illustration of the scene graph is shown in Fig. 2(a).

Inspired by (Krishna et al., 2017), we use an attention mechanism to inject information from questions into the scene graphs. This is obtained by first concatenating the question embedding $\boldsymbol{q}$ with each of the



(a) Scene Graph (b) Concept Graph

Figure 2: Scene graph and concept graph illustration. The black arrows show the direction of relations (head $\rightarrow$ tail) with class labels of relations. Object labels are in green, blue, red, and brown boxes. Yellow boxes contain entities extracted from external knowledge.

visual features v_i to generate node embeddings. Self-attention is then performed on each node to obtain attended visual features based on the relations between a target object and its neighboring objects. The attended visual features are added to the original visual feature v_i to serve as final visual features $\boldsymbol{V}$.

Concept graph construction: Our proposed model is designed to incorporate the general knowledge to the VQA model by integrating the explicit relations between knowledge entities. For questions that need outside knowledge insights, knowledge instances might have different weights. Therefore in designing the proposed graph, we use a question-conditioned attention mechanism and dynamically assign higher weights to those knowledge instances that are mostly relevant to each question, instead of treating all the entities equally.

Considering each entity e_i as one node, we construct a concept graph G_e where edges are extracted from the adjacency matrix A obtained by the knowledge retrieval module (Section 4.1). An illustration of the concept graph is shown in Fig. 2(b).

We pass the entities e_i through a GRU model to generate entity embeddings $\boldsymbol{e}_i \in \mathbb{R}^{d_e}$ ($d_e = 1024$ in our experiment). To obtain a question-conditioned attention mechanism, we first concatenate entity embeddings $\boldsymbol{e}_i$ with the question embedding $\boldsymbol{q}$ to generate node embeddings:

$$\boldsymbol{e}'_i = [\boldsymbol{e}_i || \boldsymbol{q}] \quad \text{for } i = 1, ..., 2k \tag{3}$$

GATs are used to assign different weights to N neighbours of a target node. The graph attention mechanism is employed by applying multi-head attention with the attention mechanism as the following:

$$e^*_i = \phi(\sum_{j \in N_i} \alpha_{ij}(D\boldsymbol{e}'_j + b_{rel(i,j)})) \tag{4}$$

$\phi(.)$ is a nonlinear activation function such as ReLU. The attention weight α_{ij} depends on node embeddings as well as the relation between entity i and j, denoted as

$$\begin{aligned} \alpha_{ij} &= \frac{\exp(\alpha^e_{ij})}{\sum_{j \in N_i} \exp(\alpha^e_{ij})}), \\ \alpha^e_{ij} &= (B\boldsymbol{e}'_i)^\top . C\boldsymbol{e}'_j + d_{rel(i,j)}. \end{aligned} \tag{5}$$

where $B, C, D \in \mathbb{R}^{d_h \times (d_e + d_q)}$ are projection matrices. b, d are bias terms and $rel(i, j)$ represents the label of each edge. α^e_{ij} represents the similarity between the entity features, computed by scaled dot-product and is sensitive to the relation between entities.

After encoding all the entities via the above graph attention mechanism, the output features e^*_i of attention heads are concatenated and added to the original entity embeddings $\boldsymbol{e}_i$ to generate final entity features $\boldsymbol{E}$.

4.3 Multimodal Fusion

We fuse the outputs of question embeddings, scene graphs, and concept graphs to create a joint language-vision-knowledge representation. The fusion method needs to detect high-level interactions between these three features to provide a meaningful answer, without erasing the lower-level interactions extracted in the previous steps.

Popular fusion methods such as BAN (Kim et al., 2018) or MUTAN (Ben-younes et al., 2017) are not suitable for our work since we have three types of features to fuse. Therefore, we design a fusion method by applying the Compact Trilinear Interaction (CTI) (Do et al., 2019) to the question embeddings, scene graph visual features, and concept features and generate a vector to jointly represent the three features.

Given $\boldsymbol{V} \in \mathbb{R}^{n_v \times d_v}$, $\boldsymbol{Q} \in \mathbb{R}^{n_h \times d_q}$ where n_h is the number of hidden states ($n_h = 1$ in our experiments, i.e., $\boldsymbol{Q} = \boldsymbol{q}$), and $\boldsymbol{E} \in \mathbb{R}^{2k \times d_e}$ generated from previous steps, we generate a joint representation $z \in \mathbb{R}^{d_z}$. The joint representation z is computed by applying CTI to each $(\boldsymbol{V}, \boldsymbol{Q}, \boldsymbol{E})$:

$$z = \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} \sum_{m=1}^{2k} \mathcal{M}_{ijm} \left(\boldsymbol{V}_i W_{z_v} \circ \boldsymbol{Q}_j W_{z_q} \circ \boldsymbol{E}_m W_{z_e} \right) \tag{6}$$

where $\mathcal{M}$ is an attention map $\mathcal{M} \in \mathbb{R}^{n_v \times n_h \times 2k}$:

$$\mathcal{M} = \sum_{r=1}^{R} [\![\mathcal{G}_r; \boldsymbol{V} W_{v_r}, \boldsymbol{Q} W_{q_r}, \boldsymbol{E} W_{e_r}]\!] \tag{7}$$

where $W_{z_v}, W_{z_q}, W_{z_e}, W_{v_r}, W_{q_r}, W_{e_r}$ are learnable factor matrices, and $\circ$ is the Hadamard product. R is a slicing parameter, establishing a trade-off between the decomposition rate and the performance, and $\mathcal{G}_r \in \mathbb{R}^{d_{q_r} \times d_{v_r} \times d_{e_r}}$ is a learnable Tucker tensor.

The joint embedding computes more efficient and more compact representations than simply concatenating the embeddings. In addition, we overcome the issue of dimensionality faced with concatenating large matrices. The output of the fusion model is then fed to a classifier to predict the answer.

5 Experiments

We evaluate the performance of our proposed model using the standard evaluation metric recommended in the VQA challenge (Agrawal et al., 2017):

$$Acc(ans) = min \left(1, \frac{\#\{\text{humans provided ans}\}}{3} \right) \tag{8}$$

All experiments have been performed on OK-VQA dataset.

OK-VQA: is composed of 14,031 images and 14,055 questions. OK-VQA is divided into eleven categories: vehicles and transportation (VT); brands, companies and products (BCP); objects, materials and clothing (OMC); Sports and Recreation (SR); Cooking and Food (CF); Geography, History, Language and Culture (GHLC); People and Everyday Life (PEL); plants and animals (PA); science and technology (ST); weather and climate (WC). If a question was classified as belonging to different categories by different people, it was categorized as "Other".

Implementation details: Each image has a total of 36 image region features ($n_v = 36$), each represented by a bounding box and an embedding vector computed by pre-trained Faster R-CNN features where $d_v = 2048$. The input questions are embedded using GRU where $d_q = 1024$ and $n_t = 14$. We use USE and SBERT to embed instances. USE outputs a 512 dimensional vector ($d_s = 512$) where as $d_s = 768$ using SBERT. Our GATs include 16 attention heads. In the fusion model, $R = 32$ as suggested in (Do et al., 2019) and $d_z = 512$ since it leads to the best results in our model.

To train our proposed model, we use a binary cross-entropy loss with a batch size of 64 over a maximum of 20 epochs on 8 Tesla GPUs. We use the Adamax optimizer with an initial learning rate of 1e-3. A linear decay learning rate schedule with warm up is used to train the model. The details of different experimental setups and results are provided in the following subsections.

	@10	@20	@30	@40	@50
USE	26.08	28.83	**29.03**	28.48	27.92
SBERT	25.93	28.38	28.25	28.26	27.66

Table 1: Performance on OK-VQA validation set for different number of knowledge instances

Model	Overall	VT	BCP	OMC	SR	CF	GHLC	PEL	PA	ST	WC	Other
BAN	25.17	23.79	17.67	22.43	30.58	27.90	**25.96**	20.33	25.60	20.95	**40.16**	22.46
G_s + *BAN*	25.95	22.97	20.21	21.77	34.36	25.86	21.67	22.73	27.77	25.74	37.68	24.72
G_s+G_e	**29.03**	**26.11**	**24.15**	**26.36**	**36.94**	**30.92**	25.15	**24.83**	**29.58**	**29.38**	39.64	**26.34**

Table 2: Performance on Ok-VQA validation set for different graph setups.

5.1 Knowledge Retrieval Experiments

Our knowledge retrieval module uses sentence-level embeddings to represent question-image and knowledge instances. For this purpose, we use two different methods based on popular Transformer and BERT networks.

We first passed each question-image and knowledge instances through a bert-as-a-service model (Xiao, 2018) and derived a fixed sized vector by using the output of the special `CLS` token. This technique yielded rather worst performance. Our theory is that, since Bert model is not trained for sentence-level similarity tasks, it did not work well in our model. To bypass this limitation, we use SBERT which is a modification of the pre-trained BERT network to generate semantically meaningful sentence embeddings. We have tested SBERT with different backbones, e.g. BERT-base model with mean-tokens pooling and with `CLS` token pooling, BERT-large with mean-tokens pooling and with `CLS` token pooling, RoBERTa-base with mean-tokens pooling, and RoBERTa-large with mean-tokens pooling. The best results were achieved by BERT-base model with mean-tokens pooling. As the second approach, we compute the sentence embeddings using USE-large model which is trained with a Transformer encoder.

We then compute similarity scores which are the cosine similarity of a question-image instance embedding and knowledge instance embedings. Top-k knowledge instances with the highest similarity scores are retrieved as mostly relevant knowledge to each question. The accuracy of the VQA task using USE and SBERT for different number of k is reported in Table 1.

As indicated in Table 1, we observe a higher accuracy using USE technique. We chose $k = 30$ and USE to retrieve knowledge instances for the rest of experiments as this gives the best accuracy.

5.2 Graph Construction Experiments

In this subsection, we evaluate the advantage of capturing inter-object relations as well as incorporating general knowledge. Table 2 shows the results of this experiment as explained below:

BAN: In this setup, we fuse the question embedding $\boldsymbol{q}$ with object features $\boldsymbol{\mathcal{V}}$ using Bilinear Attention Networks (BAN). The output of the fusion network is then fed to the classifier to predict the answer. We found that setting γ (number of glimpses) to 2 in BAN model yields the best performance in our model. We do not use scene graphs nor concept graphs in this experiment.

G_s + *BAN*: The same fusion network as *BAN*, but visual features are $\boldsymbol{V}$ resulted by the scene graph. The question embedding $\boldsymbol{q}$ is fused with $\boldsymbol{V}$ and passed to the classifier to predict the answer.

$G_s + G_e$: This is the complete form of our model. $\boldsymbol{q}$, $\boldsymbol{V}$, and $\boldsymbol{E}$ are fused in this setup. CTI is used for the fusion technique as explained in Section 4.3.

From Table 2, consistent performance gain is obtained across most of categories by combining the scene graph and concept graph. Adding the scene graph alone improves the results in BCP, SR, PEL, PA, ST, and Other categories. Combining the scene graph and concept graph boosts the performance on the remaining categories except "Geography, History, Language and Culture" (GHLC) and "Weather and Climate" (WC).

Model	Overall	VT	BCP	OMC	SR	CF	GHLC	PEL	PA	ST	WC	Other
Q-Only	15.08	13.64	13.19	11.78	15.94	16.82	11.91	13.38	13.22	18.76	23.74	13.51
Q + G_e	20.43	19.14	18.69	16.28	21.09	21.42	17.42	19.52	19.78	22.28	30.24	18.87
MUT+AN	27.58	25.56	23.95	**26.87**	33.44	29.94	20.71	**25.05**	**29.70**	24.76	**39.84**	23.62
BAN + AN	25.63	24.45	19.88	21.59	30.79	29.12	20.57	21.54	26.42	27.14	38.29	22.16
XNM Net	25.06	25.73	21.86	18.22	33.02	23.93	23.83	20.79	24.81	21.43	37.74	24.39
Our model	**29.03**	**26.11**	**24.15**	26.36	**36.94**	**30.92**	**25.15**	24.83	29.58	**29.38**	39.64	**26.34**

Table 3: Performance on Ok-VQA validation set for ablation study and compared with SOTA.

5.3 Ablation Study and Comparison with SOTA

In Table 3, we compare two ablated instances of our model (Q-Only and Q + G_e) with its complete form. We also report the accuracy of the state-of-the-art baselines on OK-VQA dataset. ArticleNet (AN) (Marino et al., 2019) is a knowledge-based approach that retrieves articles from Wikipedia. Moreover, we applied XNM Net model (Shi et al., 2019) on Ok-VQA dataset and provide the results in the table.

Table 3 shows the accuracy on the OK-VQA validation set in the following setting:

Q-Only: Only question embedding $\boldsymbol{q}$ is fed to the classifier.

Q + G_e: Integrating general knowledge without using visual features in the pipeline. Questions embedding $\boldsymbol{q}$ and entity embedding $\boldsymbol{E}$ are fused using BAN and fed to the classifier. G_s is removed from the model in this experiment.

MUT + AN: SOTA- Incorporate hidden states of ArticleNet for the top retrieved sentences into MUTAN.

BAN + AN: SOTA- Incorporate hidden states of ArticleNet for the top retrieved sentences into BAN.

XNM Net: SOTA- Scene graph-based VQA model. The explicit relationships between objects are not considered in this work. Node embeddings are concatenated and used as edge features.

Our model: The complete form of our model. Question embedding $\boldsymbol{q}$, scene graph embedding $\boldsymbol{V}$, and concept graph embedding $\boldsymbol{E}$ are fused using CTI and fed to the classifier.

From the table, we observe that adding general knowledge to the model leads to a gain of 5.35% in the overall performance (*Q-Only* = 15.08 % vs. *Q + G_e* = 20.43 %).

We also note that the *G_s + BAN* setup (cf. Table 2) results in a better performance compared to *Q + G_e*. The reason is that most of the questions in OK-VQA datasets are related to objects found in the images. Therefore, the accuracy drops without providing the visual features.

Furthermore, we observe that our model surpasses the SOTA models in most of the categories. Our model performs especially well in SR, ST, and Other categories with a gain larger than 3%. Our performance is slightly below ArticleNet (*MUTAN + AN*) in OMC, PEL, PA, and WC. However unlike ArticleNet, our model does not require query construction and search APIs to retrieve relevant knowledge.

5.4 Qualitative Results

Fig. 3 depicts a few success and failure examples of our method. In the top two rows of examples, our model correctly retrieve the most relevant knowledge instances to the question and predict the correct answer. We only show top-3 retrieved knowledge instances in these examples due to the lack of space. Question, predicted answer, and provided answers are denoted by **Q**, **A**, and **GT**, respectively. (.) in **GT** shows number of humans that provided the answer. For example in the first example in the first row, 4 human annotators provided the answer "wool".

These results show incorporating external knowledge can improve the relational representation between entities in the question and objects in the image. For instance, the third example in the first row asks `What kind of animal this resemble?` To answer this questions, one requires to know that the question is referred to the airplane (visual information) and airplanes are designed to imitate birds (general knowledge).

The last row in Fig 3 shows failure cases. In the first example, our model fails because the retrieved knowledge instances are not mostly relevant to the question. The second example retrieves relevant knowledge instances, but it is not able to predict the correct answer. The answer predicted in the fourth

Q: What are these animals farmed for?
A: milk
GT: wool(4), milk(3), meat(3)
Top-3 retrieved knowledge instances:
1- sheep is animal that eat grass and produce wool and milk
2- sheep is farm animal
3- sheep used for shear

Q: What is the blue item used for?
A: fight fire
GT: put out fire(4), fight fire(3), water(3)
Top-3 retrieved knowledge instances:
1- fire hydrant used for fight fire
2- fire hydrant receives action use by fire department
3- fire engine used for put out fire

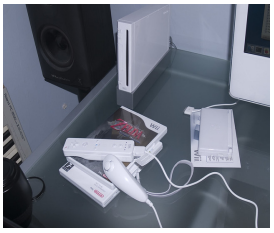
Q: What kind of animal this resemble?
A: bird
GT: bird (10)
Top-3 retrieved knowledge instances:
1- airplane used for imitate bird
2- this is us military airplane
3- most mammal not capable of fly

Q: What climate the yellow fruit grow?
A: tropical
GT: tropical(6), warm (4)
Top-3 retrieved knowledge instances:
1- fruit receives action grow in warm weather
2- fruit not capable of grow in cold place
3- banana is tropical fruit

Q: What do they call running around the bases on a single hit?
A: homerun
GT: homerun (10)
Top-3 retrieved knowledge instances:
1- home run is baseball term
2- score home run used for bat in other runner on base
3- bat used for hit baseball

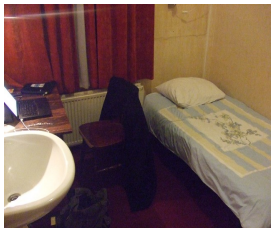
Q: Name the type of hat the man is wearing
A: cowboy
GT: cowboy(4), sombrero(4), stetson(2)
Top-3 retrieved knowledge instances:
1- cowboy capable of wear cowboy hat
2- hat used for slap your horse
3- this is hat man would wear

Q: What type of vessel would remain in this environment?
A: boat
GT: boat(10)
Top-3 retrieved knowledge instances:
1- boat that capable of travel under water
2- boat used for keep person out of water
3- vessel used for move person around



Q: What company created this gaming system?
A: nintendo
GT: nintendo (10)
Top-3 retrieved knowledge instances:
1- tv used for video game
2- nintendo is video game company
3- game created by programmer

Q: How is this beverage made?
A: bake
GT: brew(6), coffee maker(2), blender(2)
Top-3 retrieved knowledge instances:
1- cup on table used for drink from
2- cup on table used for fill with beverage
3- cup on table has subevent pour of drink



Q: Which item is to wash hands?
A: towel
GT: sink(10)
Top-3 retrieved knowledge instances:
1- sink used for wash your hand
2- sink used for wash hair
3- sink used for wash up face and hand

Q: What is the nickname of this city?
A: new york
GT: big apple(5), windy city(3), matrix(2)
Top-3 retrieved knowledge instances:
1- stop sign used for control traffic
2- stop sign used for stop
3- sign used for traffic direction

Q: What kind of gathering is this?
A: birthday
GT: party(6), family(2), wine taste(2)
Top-3 retrieved knowledge instances:
1- guest capable of bring wine to party
2- bottle wine at location party
3- party used for meet person

Figure 3: Qualitative results on OK-VQA validation set. The first two rows show success cases and the last row shows failure cases.

example does not belong to the list of the provided answers. However, the predicted answer could be correct. This example shows that we need a better evaluation metric for VQA tasks which covers semantic cases such as this example.

6 Conclusion

In this paper, we proposed a novel VQA model for questions which require knowledge from external content. We developed a knowledge retrieval model to extract most relevant facts to each question based on sentence level embeddings. We then combined visual observations with retrieved knowledge by learning graphs to captures the interactions between objects and knowledge entities. The experimental results have shown the performance of our proposed model on OK-VQA dataset.

For future work, we will explore how to integrate the fact retrieval module to the main VQA pipeline to have an end-to-end trainable model. We will also investigate how to capture the semantic similarity between provided answers and predicted answers in the evaluation metric.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2017. Vqa: Visual question answering. *Int. J. Comput. Vision*, 123(1):4–31, May.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2017. Bottom-up and top-down attention for image captioning and VQA. *CoRR*, abs/1707.07998.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives. 2007. Dbpedia: A nucleus for a web of open data. In Karl Aberer, Key-Sun Choi, Natasha Fridman Noy, Dean Allemang, Kyung-Il Lee, Lyndon J. B. Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, and Philippe Cudré-Mauroux, editors, *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer.

Hedi Ben-younes, Rémi Cadène, Matthieu Cord, and Nicolas Thome. 2017. MUTAN: multimodal tucker fusion for visual question answering. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2631–2639. IEEE Computer Society.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder. *CoRR*, abs/1803.11175.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Tuong Do, Huy Tran, Thanh-Toan Do, Erman Tjiputra, and Quang D. Tran. 2019. Compact trilinear interaction for visual question answering. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 392–401. IEEE.

Francois Garderes, Maryam Ziaeefard, Baptiste Abeloos, and Freddy Lecue. 2020. Conceptbert: Concept-aware representation for visual question answering. In *Findings of ACL: EMNLP*.

Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. 2018. Bilinear attention networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 1571–1581.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123(1):32–73.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Linjie Li, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Relation-aware graph attention network for visual question answering. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV*, pages 10312–10321. IEEE.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. OK-VQA: A visual question answering benchmark requiring external knowledge. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3195–3204. Computer Vision Foundation / IEEE.

Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodolà, Jan Svoboda, and Michael M. Bronstein. 2017. Geometric deep learning on graphs and manifolds using mixture model cnns. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5425–5434. IEEE Computer Society.

Medhini Narasimhan and Alexander G. Schwing. 2018. Straight to the facts: Learning knowledge base retrieval for factual visual question answering. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, volume 11212 of *Lecture Notes in Computer Science*, pages 460–477. Springer.

Medhini Narasimhan, Svetlana Lazebnik, and Alexander G. Schwing. 2018. Out of the box: Reasoning with graph convolution nets for factual visual question answering. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 2659–2670.

Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. 2018. Learning conditioned graph structures for interpretable visual question answering. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 8334–8343. Curran Associates, Inc.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Sanket Shah, Anand Mishra, Naganand Yadati, and Partha Pratim Talukdar. 2019. KVQA: knowledge-aware visual question answering. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 8876–8884. AAAI Press.

Jiaxin Shi, Hanwang Zhang, and Juanzi Li. 2019. Explainable and explicit visual reasoning over scene graphs. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019*, pages 8376–8384.

Thomas Steiner, Ruben Verborgh, Raphaël Troncy, Joaquim Gabarro, and Rik Van de Walle. 2012. Adding realtime coverage to the google knowledge graph. In *11th International Semantic Web Conference (ISWC 2012)*. Citeseer.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 697–706.

D. Teney, L. Liu, and A. Van Den Hengel. 2017. Graph-structured representations for visual question answering. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3233–3241.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations*. OpenReview.net.

Peng Wang, Qi Wu, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. 2017. Explicit knowledge-based reasoning for visual question answering. In Carles Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 1290–1296. ijcai.org.

Peng Wang, Qi Wu, Chunhua Shen, Anthony R. Dick, and Anton van den Hengel. 2018. FVQA: fact-based visual question answering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(10):2413–2427.

Han Xiao. 2018. bert-as-service. `https://github.com/hanxiao/bert-as-service`.