Augmenting Visual Question Answering with Semantic Frame Information in a Multitask Learning Approach

Mehrdad Alizadeh
University of Illinois at Chicago
Chicago, Illinois, USA
Email: maliza2@uic.edu

Barbara Di Eugenio
University of Illinois at Chicago
Chicago, Illinois, USA
Email: bdieugen@uic.edu

Abstract—**Visual Question Answering (VQA) concerns providing answers to Natural Language questions about images. Several deep neural network approaches have been proposed to model the task in an end-to-end fashion. Whereas the task is grounded in visual processing, if the question focuses on events described by verbs, the language understanding component becomes crucial. Our hypothesis is that models should be aware of verb semantics, as expressed via semantic role labels, argument types, and/or frame elements. Unfortunately, no VQA dataset exists that includes verb semantic information. Our first contribution is a new VQA dataset (imSituVQA) that we built by taking advantage of the imSitu annotations. The imSitu dataset consists of images manually labeled with semantic frame elements, mostly taken from FrameNet. Second, we propose a multitask CNN-LSTM VQA model that learns to classify the answers as well as the semantic frame elements. Our experiments show that semantic frame element classification helps the VQA system avoid inconsistent responses and improves performance.**

I. INTRODUCTION

The goal of a Visual Question Answering (VQA) system is to answer user questions about an image [1]. In order to train neural-network based VQA models, many large-scale datasets have been created [2]. We have observed that a large portion of the questions available in current datasets, involve a verb other than ”*to be*” (i.e. 42% of VQA dataset). Questions including ”*to be*” as the primary verb are usually about *objects*, *object attributes*, *object presence*, *object frequency*, *spatial reasoning* and so on. These questions appear to be more visually than linguistically challenging. On the other hand, event verbs such as *cook* or *jump*, inherently provide semantic information that may help in answering questions about images describing such events. Semantic information about verbs includes the type of arguments a verb can take and how the arguments participate in the event expressed by a verb, but this information is missing in current VQA systems. We contend that, if a VQA system is aware of such semantic information, it can not only narrow down the possible answers but also avoid providing irrelevant responses. For example, the answer to the question ”*What is the woman cooking in the oven?*”, should belong to the *food* semantic category. However, neither do VQA datasets encode, nor has any VQA system taken advantage of this information. The question is how to incorporate such semantic information in VQA. Traditionally in linguistics, semantic information about a verb has been captured via so-called *thematic* or *semantic roles* [3], which may include roles like *agent* or *patient* as encoded in a resource such as VerbNet [4]. Semantic role labeling has been shown to improve performance in challenging tasks such as dialog systems, machine reading, translation and question answering [5], [6]. However, the difficulty of clearly defining such roles has given rise to other approaches, such as the abstract roles provided by PropBank [7], or the specialized frame elements provided by FrameNet [8]. In FrameNet, verb semantics is described by frames or situations. Frame elements are defined for each frame and correspond to major entities present in the evoked situation. For example, the frame *Cooking_creation* has four core elements, namely *Produced_food*, *Ingredients, Heating_Instrument*, *Container*. In order to create a VQA dataset with verb semantic information, we took advantage of the imSitu dataset [9], developed for situation recognition and consisting of about 125k images. Each image is annotated with one of 504 candidate verbs and its frame elements according to FrameNet [8]. A sample of images from the ImSitu dataset and their annotations can be found in Table I.[1]

In this paper, we first show how we created the new imSituVQA dataset, by employing a semi-automatic approach to create question-answer pairs derived from the imSitu dataset. We have recently publicly released the imSituVQA dataset[2]. Our second contribution is an augmented CNN-LSTM VQA model with semantic frame element information in a multi-task learning paradigm. The model is trained to classify answers as well as semantic frame elements. The two classifiers share the same weights and architectures up to the classification point. The experiments show that the frame element classification acts like a regularizer by reducing the inconsistencies between the two members of the predicted <*answer, frame element*> pair in order to provide accurate responses.

[1] imSitu substitutes some frame elements with more traditional thematic roles, for example *Agent* for *Cook*.

[2] https://github.com/givenbysun/imSituVQA

cooking				**buying**			
Agent	woman	Agent	boy	Agent	adolescent	Agent	woman
Food	vegetable	Food	meat	Goods	book	Goods	shoe
Container	pot	Container	wok	Payment	cash	Payment	credit card
Tool	knife	Tool	spatula	Seller		Seller	person
Place	kitchen	Place	kitchen	Place		Place	shoe shop

catching				**opening**			
Agent	bear	Agent	ballplayer	Agent	person	Agent	cat
Caughtitem	fish	Caughtitem	baseball	Item	can	Item	door
Tool	mouth	Tool	baseball glove	Tool	can opener	Tool	paw
Place	body of water	Place	outdoors				

TABLE I
Sample imSitu annotations of images about cooking, buying, catching and opening. [9]

II. Related Work

A. VQA Datasets

In order to train neural-network based VQA models, many large-scale datasets have been created. Datasets differ based on the number of images, the number of questions, complexity of the questions, reasoning required and content information included in the annotation for images, and questions. AQUAR [10] is among the first benchmarks released for the VQA task. It includes visual questions on color, number and physical location of an object. In the COCO-QA dataset [11] questions are generated from image captions describing the image. The VQA dataset [1], among widely used benchmarks, is a collection of diverse free form open ended questions. Visual7w [12] is a dataset with the goal of providing semantic links between textual descriptions and image regions by means of object-level grounding. FVQA [13] primarily contains questions that require external information to answer.

B. VQA Methods

Numerous baselines and methods have been proposed for the VQA task. The VQA task requires co-reasoning over both image and text to infer the correct answer. Most existing methods formulate VQA as a classification problem and impose the restriction that the answer can only be drawn from a fixed answer space. The current dominant baseline method proposed in [1] employs a CNN-LSTM-based architecture. It consists of a convolutional neural network (CNN) to extract image features and a long short term memory network (LSTM) to encode the question features. The method fuses these two feature vectors via an element-wise multiplication and then passes the result vector through fully connected layers to generate a softmax distribution over output answers.
The attention techniques learn to focus on the most discriminative regions rather than the whole image to guide the reasoning for finding the answer. Different attention techniques, such as stacked attention [14], co-attention between question and image [15], and factorized bilinear pooling with co-attention [16] have been shown to improve the performance of VQA.

III. The imSituVQA Dataset

In this section, we first briefly expand on our earlier description of imSitu, and explain how question-answer templates are generated. We then describe how they are filled with noun values from the imSitu annotated images. The process results in the creation of a new dataset, which we call imSituVQA. As we noted, the imSitu dataset [9] is tailored to situation recognition, a problem that involves predicting activities along with actors, objects, substances, and locations and how they fit together. imSitu utilizes linguistic resources such as FrameNet and WordNet in order to define a comprehensive space of situations. It provides representations helping to understand who (*AGENT*) did what (*ACTIVITY*) to whom (*PATIENT*), where (*PLACE*), using what (*TOOL*) and so on.

Verb	Question Template	Frame Element
cooking	Who is cooking?	AGENT
	What does the AGENT cook with TOOL?	FOOD
	What is the AGENT doing?	VERB
	What does the AGENT use to cook in CONTAINER?	TOOL
	Where does the AGENT cook FOOD in CONTAINER?	PLACE
buying	Who is buying GOODS?	AGENT
	What is the AGENT doing?	VERB
	What item does the AGENT buy with PAYMENT?	GOODS
	Who does the AGENT buy GOODS from?	SELLER
	Where does the AGENT buy GOODS?	PLACE
catching	Who catches at PLACE?	AGENT
	What is the AGENT doing?	VERB
	What item does the AGENT catch with TOOL?	CAUGHTITEM
	Where does the AGENT catch CAUGHTITEM?	PLACE
opening	What does the AGENT use to open ITEM?	TOOL
	Who opens ITEM?	AGENT
	What item does the AGENT open?	ITEM
	Where does the AGENT open ITEM with TOOL?	PLACE

TABLE II
A SUBSET OF QUESTION ANSWER TEMPLATES GENERATED FOR COOKING, BUYING, CATCHING AND OPENING.

Every situation in imSitu is described with one of 504 candidate verbs such as *cook*, *play*, *tattoo*, *wash*, *teach* and so on. Each verb has a set of FrameNet related frame elements[3]: for example, $S_r(cooking)$ ={ *AGENT, FOOD, CONTAINER, HEATSOURCE, TOOL, PLACE* } indicates semantic frame elements of the verb *cooking*. This set is also expressed by an abstract definition: *”an AGENT cooks a FOOD in a CONTAINER over a HEATSOURCE using a TOOL in a PLACE”*. imSitu includes 190 unique frame elements, some shared among verbs such as *AGENT* and some verb specific such as *PICKED*$\in S_r(PICKING)$.

Every image is labeled with one of 504 candidate verbs along with frame elements filled with noun values from WordNet. If an element is not present in the image its value is empty. There are about 250 images per verb and 3.55 frame elements per verb on average.

A. Question answer template generation

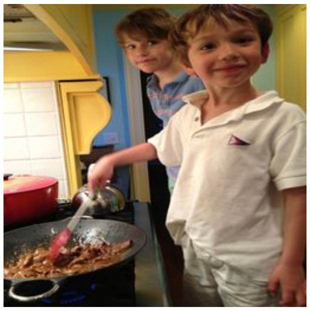
Before template generation, we mapped every frame element to a question word, for example, *AGENT* to *who*, *LOCATION* to *where*, *ITEM* , *FOOD* and *PICKED* to *what item*, *TOOL* to *what does [AGENT] use to* and so on. From 190 unique frame elements, 47 were mapped to *who*, 19 mapped to *where*, 53 mapped to *what* and the remaining were mapped to a question word starting with *what* such as *what item.*

There are 504 abstract definitions, each expressing a verb with its frame elements in a sentence. Given an abstract definition, we hold out one element as output frame element and use the remaining ones in order to generate question templates. For example, for *cook* the abstract definition is *”an AGENT cooks a FOOD in a CONTAINER over a HEATSOURCE using a TOOL in a PLACE”*. If we hold out *FOOD* then what remains is *”an AGENT cooks [X] in a CONTAINER over a HEATSOURCE using a TOOL in a PLACE”*. We created a recursive template question generation procedure that produces all possible combinations. For example, asking about *FOOD* requires templates staring with the *”What ...”* question word, then including or excluding other frame elements in the question results in different possible questions: *”What does AGENT cook?”*, *”What does AGENT cook with TOOL?”*, *”What does AGENT cook in CONTAINER?”* and so on. One advantage of this process is to generate many training samples useful for training deep models. A subset of templates for *cooking*, *buying*, *catching* and *opening* are shown in Table II. The abstract definitions for *buying*, *catching* and *opening* are: *”AGENT buys GOODS with PAYMENT from the SELLER in a PLACE”* , *”an AGENT catches a CAUGHTITEM with a TOOL at a PLACE”* and *”the AGENT opens the ITEM with the TOOL at the PLACE”*. In total, 6879 templates are generated, with on average 13.65 question-answer templates per verb.

1) Question answer pair realization: The template generation is based on 504 abstract definitions of the verbs. In order to build the real imSituVQA dataset, image annotations are used to substitute the frame elements in the templates. Each image is annotated with a verb and its frame elements with their fillers. Table I shows an example of such annotations for *cooking*, *buying*, *catching* and *opening*. All templates of a verb can be instantiated by filling frame elements with noun values from the annotation. If a verb has n templates, applying an image annotation results in n real <*question, answer*> samples of the image. Table III shows VQA samples for *cooking*, *buying*, *catching* and *opening*. This way, the size of the extracted dataset is the average number of templates times the number of images. This realization process results in 254k train, 88k development and 88k test samples. For the training set, the top 10 most frequent frame element classes among the existing 190 are shown in Table IV. Table V also shows the top 10 frequent answers. Because 60% of answers are about *place* and *agent*, the most frequent answers are usually values from these two frame elements. Figure 2

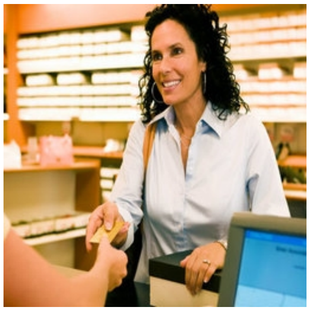
[3] As noted earlier, some are traditional thematic roles such as *AGENT* and not the corresponding FrameNet frame elements.

IMAGE about cooking



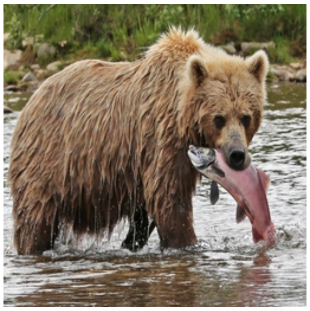
QUESTION	ANSWER	FRAME ELEMENT
Who is cooking?	boy	AGENT
What does the boy cook with spatula?	meat	FOOD
What is the boy doing?	cooking	VERB
What does the boy use to cook in wok?	spatula	TOOL
Where does the boy cook meat in wok?	kitchen	PLACE

IMAGE about buying



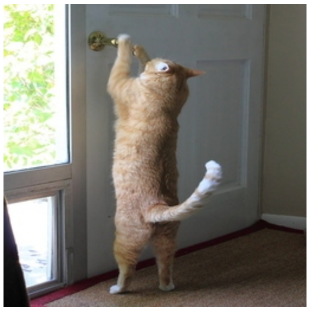
QUESTION	ANSWER	FRAME ELEMENT
Who is buying shoes?	woman	AGENT
What is the woman doing?	buying	VERB
What item does the woman buy with credit card?	shoe	GOODS
who does the woman buy shoe from?	person	SELLER
where does the woman buy shoe?	shoe store	PLACE

IMAGE about catching



QUESTION	ANSWER	FRAME ELEMENT
who catches at body of water?	bear	AGENT
what is the bear doing?	catching	VERB
where does the bear catch fish?	body of water	PLACE
what item does the bear catch? with mouth?	fish	CAUGHTITEM

IMAGE about opening



QUESTION	ANSWER	FRAME ELEMENT
what does the cat use to open the door?	paw	TOOL
who opens the door?	cat	AGENT
what item does the cat open?	door	ITEM

TABLE III
IMSITUVQA DATASET SAMPLES ABOUT COOKING, BUYING, CATCHING AND OPENING.

depicts the distribution of question template lengths in terms of the number of words. The questions are mostly between 4 to 7 words. Figure 1 shows the distribution of imSituVQA questions according to the first question word. As can be seen *”Where”* is more frequent than *”Who”* and *”What”*. This derives from *place* being the most frequent frame element, twice as frequent as *agent*, which is the second.

IV. Our VQA Model

Hyper-class augmented deep learning model has been shown to work well for fine-grained image classification. Instead of fine tuning a convolutional neural network (CNN) , [17] suggests a hyper-class augmentation formulated as multi-task learning in order to boost the recognition task. Similarly, we include frame element classification in parallel with answer classification.

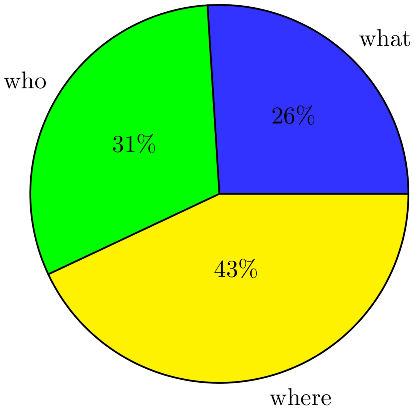


Fig. 1. Distribution of questions in imSituVQA.

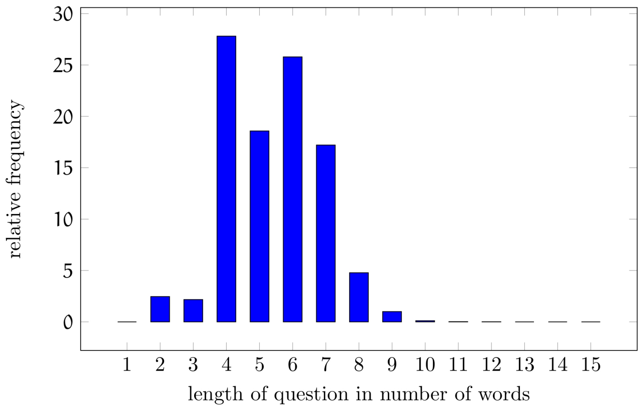


Fig. 2. Distribution of questions in imSituVQA based on length.

Frame element	frequency
PLACE	100,006
AGENT	49,976
ITEM	24,376
TOOL	13,908
VICTIM	3,932
TARGET	3,860
VEHICLE	3,706
DESTINATION	3,238
COAGENT	2,544
OBJECT	2,317

TABLE IV
TOP 10 FREQUENT FRAME ELEMENTS IN IMSITUVQA TRAINING SAMPLES.

Answer	frequency
outdoors	14,621
man	13,527
woman	10,763
people	9,228
room	8,323
outside	6,881
inside	6,679
person	5,625
hand	4,238
field	3,086

TABLE V
TOP 10 FREQUENT ANSWERS IN IMSITUVQA TRAINING SAMPLES.

Let $D_t = \{(x_1^t, y_1^t), ..., (x_n^t, y_n^t)\}$ be a set of training *<image, question>* paired samples with $y_i^t \in \{1, ..., C\}$ indicating the answers (e.g., *child*, *kitchen* and *cooking*) of *<image, question>* pair x_i^t , and let $D_a = \{(x_1^a, r_1^a), ..., (x_n^a, r_n^a)\}$ be a set of auxiliary frame element information, where $r_i \in \{1, ..., R\}$ indicates the frame element class of *<image, question>* pair x_a^t (e.g., *AGENT*, *FOOD* and *PLACE*). The goal is to learn a VQA model that correctly answers to an input *<image, question>* pair. In particular, we aim to learn a prediction function given by $\Pr(y|x)$, i.e., given the input *x:<image, question>* pair, we compute the probability that y is the answer. Similarly, we let $\Pr(r|x)$ denote the frame element classification model. Given the training *<image, question>* pairs and the answers with auxiliary frame element information, our strategy is to train a multi-task deep model. This model uses a shared CNN-LSTM VQA architecture up to the classification layer. Then sharing common features, it branches out to two different classifiers. One classifier classifies answers, and the other one, frame elements. Figure 3 summarizes the proposed multi-task learning model. In order to train the proposed VQA model, the total loss is the average of losses from these two classifiers.

V. EVALUATION

A. Experimental Setup

The proposed VQA model is evaluated by means of the CNN-LSTM-based architecture introduced in [18]. Training deep models requires significant time and resources. Consequently, we employ trained models such as GLOVE [19] and VGG-NET [20]. GLOVE provides a good word embedding layer initialization that generalizes well and provides a performance boost. GLOVE 300-dimensional weights are utilized in order to feed question words to a bidirectional long short term memory network (LSTM). The output of the LSTM is a 300 dimension question embedding which is mapped to 1024 dimensions by passing through a nonlinear layer. A VGG-NET-16 pre-trained model was used in order to extract image feature vectors. The 4096 image embedding is mapped to 1024 dimensions by passing through a nonlinear layer. The multimodal fusion of image and question embeddings occurs via pointwise multiplication, then after passing through two nonlinear layers with *tanh* activation function, the final embedding is fed to the frame element softmax layer and the answer softmax layer. The model is trained by minimizing the sum of the two cross-entropy loss functions using the rmsprop optimization algorithm [21]. The training data is passed with a batch size of 500 in 50 epochs.

B. Results and Discussions

Table VI shows the performance evaluation of the test samples. Using the most frequent answer (prior) in order to answer each question results in 5.65% accuracy. Selecting the most frequent answer per verb results in 22.15% accuracy. The CNN-LSTM model was trained with single answer softmax (39.58% accuracy) and multi-task, including both answer softmax and frame element softmax (44.90% accuracy). Augmenting VQA with frame element information boosts the accuracy up to 5%. This improvement in the generalization of the CNN-LSTM model indicates how well the multi-task approach acts like a regularizer. A chi-square test was performed in order to show statistically significant improvement of the model (Table VII).

Performance can be compared in terms of WUPS as well. Wu-Palmer (WUP) Similarity can be used as an alternative to accuracy [22]. [23] extended WUP similarity to the VQA

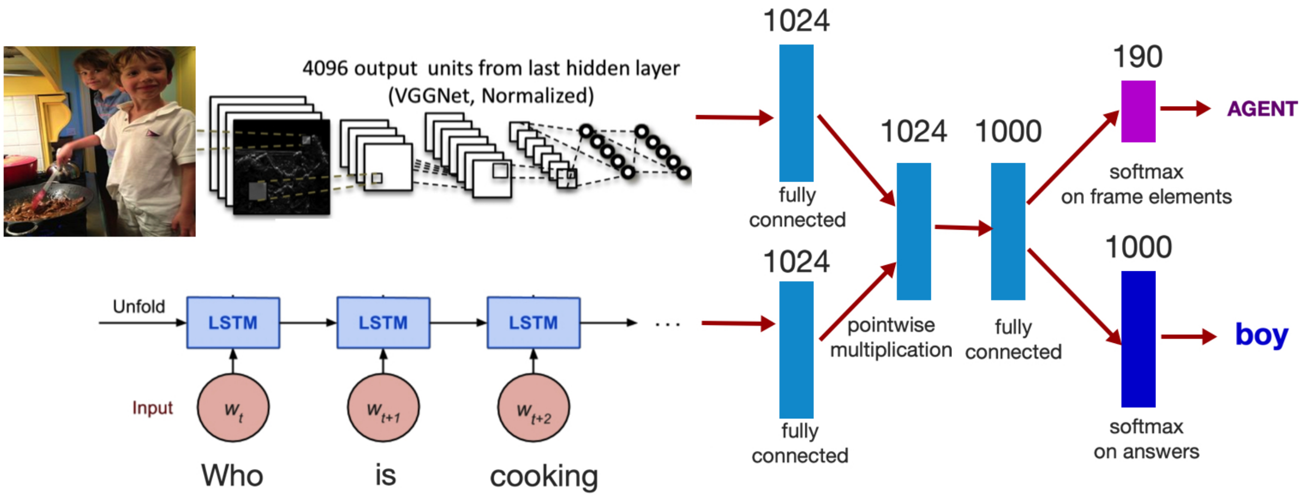


Fig. 3. Proposed multitask learning architecture for VQA

	Accuracy (%)	WUPS at 0.9 (%)
prior (”outdoors”)	05.68	11.87
per verb prior	22.15	27.65
CNN-LSTM	39.58	46.92
multi-task CNN-LSTM	**44.90**	**51.83**

TABLE VI
PERFORMANCE OF OUR VQA MODEL ON IMSITUVQA DATASET

task evaluation. WUP is based on how the predicted answer semantically matches the ground truth. Given a predicted answer and a ground truth answer, WUPS computes a value between 0 and 1 based on their similarity. It computes similarity by considering the depths of the two synsets in WordNet, along with the depth of the LCS (Longest Common Subsumer). WUPS is computed based on WUP. Given N number of samples with A being the ground truth answers and T predicted answers, the formula is as follows:

$$\mathrm{WUPS}(A,T) = \frac{1}{N}\sum_{i=1}^{N} \min \{\prod_{a\in A^i} \max_{t\in T^i} \mathrm{WUP}(a,t), \prod_{t\in T^i} \max_{a\in A^i} \mathrm{WUP}(a,t)\}.100 \quad (1)$$

Here are some examples of the pure WUP score to give intuitions about the range: *WUP(outside, outdoors)* = 0.92, *WUP(man,woman)* = 0.07, *WUP(land,earth)* = 1.0 *WUP(tree,water)*= 0.14 and *WUP(dog,wolf)* = 0.93.

”WUPS at 0.9” applies a threshold and considers a predicted answer correct if the WUPS score is higher than 0.9. ”WUPS at 1.0” corresponds to accuracy and [23] found that for VQA tasks a WUP score of around 0.9 is required for precise answers. Table VI shows performance in terms of ”WUPS at 0.9”. The improvement based on WUPS, using multi-task approach, is almost similar to that based on accuracy.

The accuracy of frame element classification is initially 90% and gets up to 99.68% at the end of the training. The performance on test data is 99.32%. This improvement,as we will discuss later, helps the model to provide more consistent responses and to regularize the model.

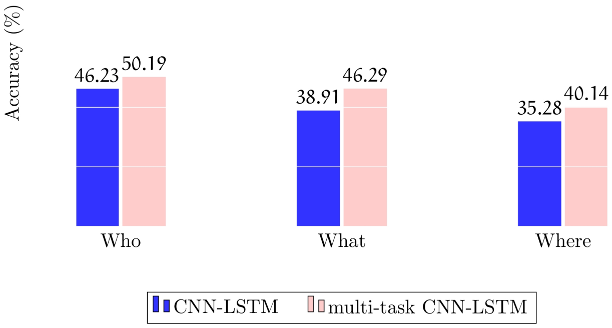


Fig. 4. Evaluation by wh-question type of the question

	Correct	Incorrect
CNN-LSTM	34905	53065
multi-task CNN-LSTM	39522	48448

TABLE VII
THE CHI-SQUARE STATISTIC IS 496.1854. THE P-VALUE IS < 0.01 AND THE RESULT IS SIGNIFICANT.

Frame element classification: The hyper-class augmentation model utilizes frame element classification for better representation learning of the VQA task. As discussed earlier, the accuracy of the frame element classification is 99.32%. One important reason for such high performance is the frame element dependency on the input question while it is independent of the input image. For example for the question *"who is cooking ?"* the frame element is always *AGENT* for all images about cooking. This results in a huge amount of data to train the frame element classification resulting in almost perfect performance.

It is interesting to know how frame element classification affects the predicted answer and how consistent it is with the correct answer and predicted answer. We consider the correct or predicted answer to be consistent with the frame element if there is at least one training sample labeled with both the answer and the frame element. For example <*bear, AGENT*> and <*bear, CHASEE*> are consistent but <*bear, PLACE*> and <*bear, TOOL*> are inconsistent. Figure 5 shows the frequency of distinct frame elements for a subset of answers. For example *man, car, telephone, bear* and *cafe* are fillers of 81, 37, 20, 8 and 1 distinct frame elements in the training samples respectively. An answer is consistent with the set of distinct frame elements it fills and inconsistent with others.

The almost perfect accuracy of the frame element classifier confirms its output is almost always consistent with the correct answer. Now the question is, how much does frame element classification help the predicted answer to be consistent with the semantic frame? Employing the consistency criterion, the consistency of the CNN-LSTM model is 97.56% and multi-task CNN-LSTM 99.94%. This shows a 2.38% improvement. In other words, augmenting the frame element classification decreases inconsistency in providing final responses. Consequently, the end-user would get more reasonable answers from the system.

Fine-grained evaluation. In order to perform a fine-grained analysis of the results, performance per question, per verb and per role are computed. Figure 4 shows a performance comparison based on the *wh-question type* of the question. Multi-task CNN-LSTM performs better for who (4%), what (8%) and where (5%) when compared to CNN-LSTM. Exploring performance per verb, we can see for example *cooking* improves from 30.12% to 44.58% and *buying* from 27.42% to 64.52%. Exploring performance per role, for example, the multi-task approach improves *AGENT* from 48.78% to 52.29%, *PLACE* from 34.75% to 39.52% and *ITEM* from 32.27% to 39.65%. Table VIII shows a different view of the performance difference between CNN-LSTM and the multi-task version. About 55% of verbs improve by less than 10%. *whipping, buying, sketching, scooping, making* improve by more than 30%. *spanking, ejecting, farming, hitting, harvesting, moistening* decline by more than 15%.

Accuracy Difference Range	Verb Frequency	Role Frequency
(-40%,-30%]		3
(-30%,-20%]	2	3
(-20%,-10%]	10	5
(-10%,0%)	67	24
0%	27	32
(0%,10%]	269	68
(10%,20%]	100	24
(20%,30%]	15	13
(30%,40%]	6	
(40%,50%]		4
(50%,60%]		2
...		
100%		1

TABLE VIII
PERFORMANCE EVALUATION GROUPED BY PERFORMANCE INTERVALS SHOWING VERB FREQUENCY AND ROLE FREQUENCY IN EACH GROUP.

VI. CONCLUSIONS

In this paper, we explained how we used the imSitu annotations to build a VQA dataset with semantic verb information. Furthermore, we proposed a multitask learning approach in order to augment a CNN-LSTM VQA model. The approach boosts performance and shows the benefit of using verb semantics in answering questions about images. The proposed model utilizes semantic frame elements in order to answer the input question about the image. We evaluated the proposed model showing 5% improvement in *accuracy* and "*WUPS at* 0.9".

We provided a justification of why the proposed hyper-class augmentation idea works and also explored its effect through different analysis. However additional theoretical analysis would enrich the proposed VQA model and system. One hypothesis would be to consider the semantic information equivalent to context or context-aware information [24] [25]. In this work we created a VQA dataset where questions are annotated with precise frame element information. Another approach would be to employ a semantic role labeling model in order to approximately extract frame element information for any question of an available VQA dataset, and then explore how frame element augmentation would work.

The hyper-class augmentation is a novel technique in the context of VQA. This idea can be generalized by augmenting the VQA models with answer types, task types, and other auxiliary information by means of the multi-task learning

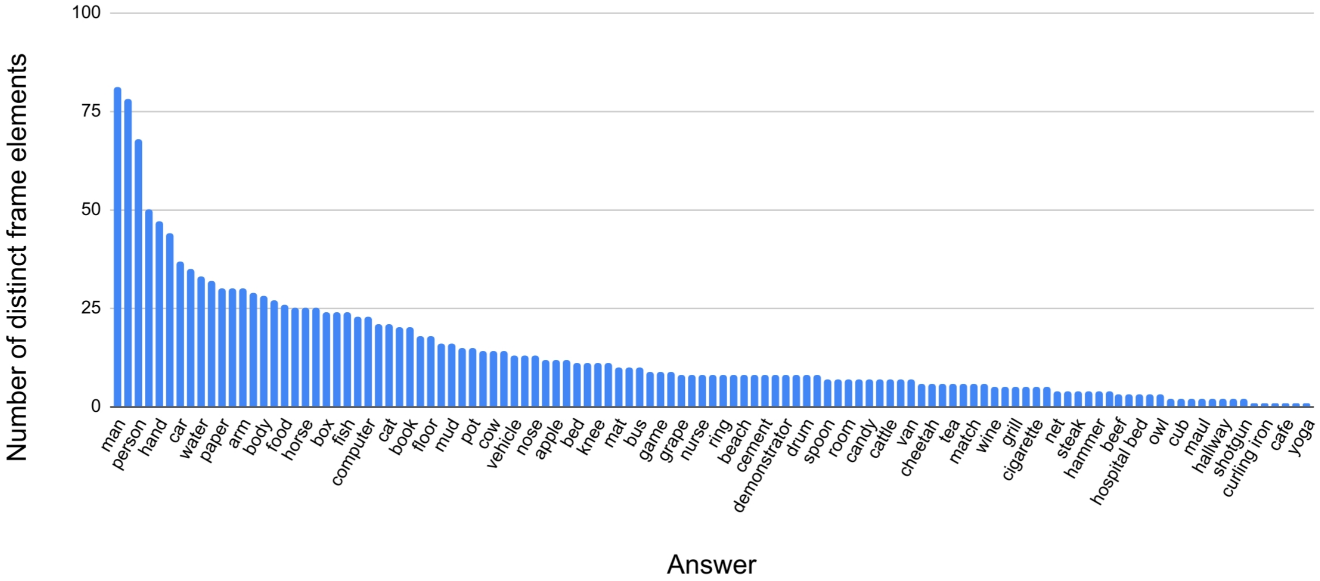


Fig. 5. Distinct frame element frequency for different answers.

paradigm.

REFERENCES

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "VQA: Visual Question Answering," in *International Conference on Computer Vision (ICCV)*, 2015.

[2] K. Kafle and C. Kanan, "Visual question answering: Datasets, algorithms, and future challenges," *Computer Vision and Image Understanding*, vol. 163, pp. 3–20, 2017.

[3] J. H. Martin and D. Jurafsky, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall, 2009.

[4] K. Kipper, A. Korhonen, N. Ryant, and M. Palmer, "A Large-Scale Classification of English Verbs," *Journal of Language Resources and Evaluation*, vol. 42, no. 1, pp. 21–40, 2008.

[5] E. Strubell, P. Verga, D. Andor, D. Weiss, and A. McCallum, "Linguistically-informed self-attention for semantic role labeling," *arXiv preprint arXiv:1804.08199*, 2018.

[6] D. Shen and M. Lapata, "Using semantic roles to improve question answering," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007.

[7] M. Palmer, D. Gildea, and P. Kingsbury, "The proposition bank: An annotated corpus of semantic roles," *Computational Linguistics*, vol. 31, no. 1, pp. 71–105, March 2005.

[8] C. J. Fillmore, C. R. Johnson, and M. R. Petruck, "Background to framenet," *International journal of lexicography*, vol. 16, no. 3, pp. 235–250, 2003.

[9] M. Yatskar, L. Zettlemoyer, and A. Farhadi, "Situation recognition: Visual semantic role labeling for image understanding," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[10] M. Malinowski and M. Fritz, "A multi-world approach to question answering about real-world scenes based on uncertain input," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., 2014, pp. 1682–1690.

[11] M. Ren, R. Kiros, and R. Zemel, "Exploring models and data for image question answering," in *Advances in neural information processing systems*, 2015, pp. 2953–2961.

[12] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei, "Visual7w: Grounded question answering in images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4995–5004.

[13] P. Wang, Q. Wu, C. Shen, A. Dick, and A. van den Hengel, "Fvqa: Fact-based visual question answering," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[14] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 21–29.

[15] J. Lu, J. Yang, D. Batra, and D. Parikh, "Hierarchical question-image co-attention for visual question answering," in *Advances In Neural Information Processing Systems*, 2016, pp. 289–297.

[16] Z. Yu, J. Yu, J. Fan, and D. Tao, "Multi-modal factorized bilinear pooling with co-attention learning for visual question answering," in *Proc. IEEE Int. Conf. Comp. Vis*, vol. 3, 2017.

[17] S. Xie, T. Yang, X. Wang, and Y. Lin, "Hyper-class augmented and regularized deep learning for fine-grained image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2645–2654.

[18] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2425–2433.

[19] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[22] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1994, pp. 133–138.

[23] M. Malinowski and M. Fritz, "A multi-world approach to question answering about real-world scenes based on uncertain input," in *Advances in neural information processing systems*, 2014, pp. 1682–1690.

[24] S. Li, A. Karatzoglou, and C. Gentile, "Collaborative filtering bandits," in *In Proceedings of the 39th International ACM SIGIR Conference on Information Retrieval*. SIGIR16, 2016.

[25] C. Gentile, S. Li, P. Kar, A. Karatzoglou, G. Zappella, and E. Etrue, "On context-dependent clustering of bandits," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1253–1262.