A Corpus for Visual Question Answering Annotated with Frame Semantic Information

Mehrdad Alizadeh, Barbara Di Eugenio
Department of Computer Science,
University of Illinois at Chicago,
Chicago, IL USA
{maliza2, bdieugen}@uic.org

Abstract

Visual Question Answering (VQA) has been widely explored as a computer vision problem, however enhancing VQA systems with linguistic information is necessary for tackling the complexity of the task. The language understanding part can play a major role especially for questions asking about events or actions expressed via verbs. We hypothesize that if the question focuses on events described by verbs, then the model should be aware of or trained with verb semantics, as expressed via semantic role labels, argument types, and/or frame elements. Unfortunately, no VQA dataset exists that includes verb semantic information. We created a new VQA dataset annotated with verb semantic information called imSituVQA. imSituVQA is built by taking advantage of the imSitu dataset annotations. The imSitu dataset consists of images manually labeled with semantic frame elements, mostly taken from FrameNet.

Keywords: Visual Question Answering, Verb Semantics, FrameNet, Data Augmentation

1. Introduction

The goal of a Visual Question Answering (VQA) system is to answer user questions about an image (Antol et al., 2015b). Neural network based VQA models require large datasets in order to be trained efficiently (Kafle and Kanan, 2017). Currently available datasets approach the task mostly from a visual point of view. The questions are usually about *objects*, *object attributes*, *object presence*, *object frequency*, *spatial reasoning* and so on. However we believe the language component should play a major role as well. Exploring currently available VQA datasets, we realized a considerable portion of questions involve a verb other than "*to be*". We analyzed the VQA dataset (Antol et al., 2015a) since it has been widely used and also includes open-ended free-form questions. As shown in Figure 1, 43% of questions involve a verb other than "*to be* in the VQA dataset. In order to compute the distribution of verbs, we used an automatic semantic role labeler. For example the output labels for the question: *"What is the bride wearing on her head?"* is: *V:wear.01* *A0:bride* *AM-LOC:on* *R-A1:what*. *V:verb.0x* pattern is used to check whether *verb* equals *'be'* or not. Therefore, we hypothesize that event verbs such as *cook* or *catch*, 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. A VQA system aware of or trained with such semantic information can narrow down possible responses. For example, the answer to the question "*What is the man cooking?*", should be constrained to be about *food*. However, neither do VQA datasets encode, nor has any VQA system taken advantage of this information.

Traditionally in linguistics, semantic information about a verb has been captured via so-called *thematic* or *semantic roles* (Martin and Jurafsky, 2009), which may include roles like *agent* or *patient* as encoded in a resource such as VerbNet (Kipper et al., 2008). Semantic role labeling has been shown to improve performance in challenging tasks such as dialog systems, machine reading, translation and question answering (Strubell et al., 2018; Shen and Lapata, 2007). However, the difficulty of clearly defining such roles has given rise to other approaches, such as the abstract roles provided by PropBank (Palmer et al., 2005), or the specialized frame elements provided by FrameNet (Fillmore et al., 2003). 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*.

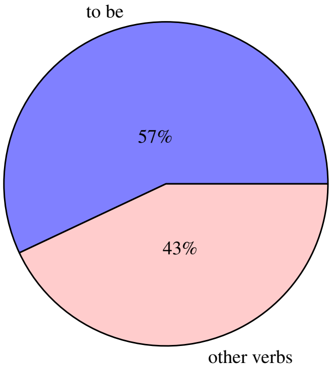


Figure 1: Distribution of verbs in the VQA dataset (Antol et al., 2015a), *'to be'* versus *'other verbs'*.

In order to create a VQA dataset with verb semantic information, we took advantage of the imSitu dataset (Yatskar et al., 2016), developed for situation recognition. The im-

fixing		**cooking**		**falling**		**buying**	
Agent	man	Agent	boy	Agent	leaf	Agent	woman
Object	roof	Food	meat	Source	tree	Goods	shoe
Part	tile	Container	wok	Goal	land	Payment	credit card
Tool	hammer	Tool	spatula			Seller	person
Place	roof	Place	kitchen			Place	shoe shop

catching		**painting**		**attaching**		**opening**	
Agent	bear	Agent	man	Agent	woman	Agent	cat
Caughtitem	fish	Item	boat	Item	fabric	Item	door
Tool	mouth	Tool	roller	Tool	hand	Tool	paw
Place	body of water	Place	outside	Place	workstation		

Table 1: Sample imSitu (Yatskar et al., 2016) annotations of images about different events described by semantic frame.

Situ dataset consists of about 125k images. Each image is annotated with one of 504 candidate verbs and its frame elements according to FrameNet (Fillmore et al., 2003). A sample of images from the ImSitu dataset and their annotations are shown in Table 1.[1]

In this paper, we describe how we created the imSituVQA dataset. First, we explain how we exploited imSitu abstract verb definitions in order to create question templates with relevant response frame elements. An abstract verb definition encodes a verb with its possible set of frame elements. For example, the abstract definition for the verb *cook* is "*an AGENT cooks a FOOD in a CONTAINER over a HEATSOURCE using a TOOL in a PLACE*". Each verb abstract definition of its frame elements is used to create question templates. Second, employing imSitu annotations, actual question answers are created. Each image is labeled with one verb. So for the templates generated for that verb, filling frame elements with noun values results in question answer pairs for each image. We have recently publicly released the imSituVQA dataset[2].

2. Related work

In this section, we review datasets created for the task of VQA. Datasets differ based on the number of images, the number of questions, complexity of the questions, reasoning required and content information included via annotation for images, and questions. The performance is usually measured by *accuracy*, but it might be data-specific as well.

The DAtaset for QUestion Answering on Real-world images (DAQUAR) (Malinowski and Fritz, 2014) was the first dataset and benchmark released for the VQA task. The images are taken from NYU-Depth V2 dataset (Silberman and Fergus, 2012). The images are all of indoor scenes. NYU-Depth V2 is annotated with semantic segmentation, meaning every pixel of an image is labeled with an object class (or no object) out of 894 possible classes. DAQUAR includes 1449 images (795 training, 654 test). Question answer pairs are collected in two ways: (1) manually by human annotators with focus on colors, numbers and objects; (2) using predefined templates to generate from the NYU dataset (*"How many [object] are in [image id]?"*). In total, 12,468 question answer pairs were collected (6,794 training, 5,674 test). Unfortunately, DAQUAR is restricted as the answers are among a predefined set of 16 colors and 894 object categories. It also suffers from bias resulting from humans focusing on a few prominent objects such as tables and chairs in the image. Beside *accuracy*, the authors proposed *WUPS* in order to measure performance. *WUPS* is defined based on *WUP* (Wu and Palmer, 1994). *WUP(a, b)* measures similarity based on the depth of two words *a* and *b* in a taxonomy such as WordNet. *WUPS* generates a score between 0 and 1. It is typically thresholded by 0.9 indicating whether an answer is correct or not.

Many VQA datasets utilize Microsoft Common Objects in Context (MS-COCO) (Lin et al., 2014) image dataset. The MS-COCO consists of 2.5 M instances of 91 object types for object recognition. The images are taken from complex everyday scenes of common objects in a natural context.

The COCO-QA dataset (Ren et al., 2015) is a dataset based

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

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

on the MS-COCO dataset. It was one of the first attempts in increasing the scale of the dataset for the VQA task. The <question, answer> pairs are automatically generated from MS-COCO caption annotations. The questions generally fall in four categories: *Object*, *Number*, *Color* and *Location*. For each image, there is one question with a single word answer. The dataset contains a total of 123,287 samples (72,783 training and 38,948 testing). Performance is assessed via either *accuracy* or *WUPS* score. The automatic conversion of captions results in a high repetition rate of the questions. Also since captions are describing the main information of the image, it does not provide detail specific questions.

The VQA dataset (Antol et al., 2015a) is the most widely used dataset for the VQA task. It is mostly because of the free-form and open-ended design of the questions and answers. For open-ended questions, potentially major AI capabilities are needed to answer: fine-grained recognition (e.g., *"What kind of food is served?"*), object detection (e.g., *"How many zebras are there?"*), activity recognition (e.g., *"Is this man playing tennis?"*), knowledgebase reasoning (e.g., *"Is this a hybrid car?"*), and commonsense reasoning (e.g., *"Does this person follow the rules?"*). Real images are selected from the MS-COCO dataset. Questions and answers were generated by crowd-sourced workers. For each question image pair, 10 answers were obtained from unique workers. Answers are usually a single word or a short phrase. Almost 38% of the questions are Yes/No, 12% Number and 50% Others. The original VQA dataset has 204,721 images with 614,163 questions, 3 questions per image on average (248,349 training, 121,512 validation, 244,302 testing). The second version of the VQA 2.0 has also been proposed (Goyal et al., 2017). It extends the VQA dataset by balancing Yes/No type of questions. A machine response is evaluated via a VQA specific accuracy measure. An answer is considered correct if it matches the answers of at least three annotators.

The Visual Genome QA (Krishna et al., 2017) is the largest dataset for VQA, (1.7 M question/answer pairs). It includes structured annotations known as scene graphs. These scene graphs specify visual elements, attributes, and relationships between elements. Questions were created by human subjects. Questions start with one of the 7 possible question words (*Who, What, Where, When, Why, How,* and *Which*). A major advantage of the Visual Genome QA dataset for VQA is the structured scene annotations. The diversity of the answers is also larger in comparison to VQA. The Visual7W dataset (Zhu et al., 2016) is a subset of the Visual Genome dataset with additional annotations. Objects mentioned in the question were drawn with bounding boxes in the image in order to resolve textual ambiguity and to enable answers of a visual nature. The questions are evaluated in a multiple choice way with 4 candidate answers of which only one is correct. The dataset contains 47,300 images and 327,939 questions.

The Compositional Language and Elementary Visual Reasoning diagnostics dataset (CLEVR) (Johnson et al., 2017) was proposed to alleviate the biased problem of VQA benchmarks. This way it prevents the models from exploiting the situation in order to answer questions without reasoning. It challenges visual reasoning capabilities such as counting, logical reasoning, comparing, and storing information in memory. It is designed so that accessing external knowledge bases and using common sense may not help in order to answer the questions. Images are annotated with ground-truth object positions and attributes (*shape, size, color, material*). Questions are generated automatically using textual templates (i.e. *"How many <Color> <Material> things are there?"*) from 90 question families. CLEVR has 100K rendered images (simple 3D shapes) and about one million questions of which 853K are unique.

The focus of many VQA datasets is on questions that require direct analysis of an image in order to answer. There are many questions that require common sense, or basic factual knowledge to be answered. FVQA (Fact-based VQA) (Wang et al., 2018) was proposed by appending supporting fact information to VQA (<image, question>,answer) samples. The supporting fact is represented as a triplet such as <Cat, CapableOf, ClimbingTrees>. 2190 images were sampled from the MS-COCO. Each image is annotated with visual concepts (objects, scenes, and actions) using available resources and classifiers. The knowledge about each visual concept is extracted from structured knowledge bases, such as DBpedia, ConceptNet, and WebChild. Annotators created 5,826 questions in which answering each question requires information from both the image and selected supporting facts.

3. The imSituVQA Dataset

This section briefly describes the imSitu dataset and explains the process of a novel VQA dataset creation (imSituVQA) from the currently available imSitu dataset. The process is composed of two primary steps: (1) *Question answer template generation*: Question answer templates are generated from imSitu abstract verb definitions. (2) *Question answer pair realization*: The templates are filled with noun values from the imSitu annotated images.

The imSitu dataset (Yatskar et al., 2016) is tailored to situation recognition and consists of about 125k images. Situation recognition is 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[3] (Fillmore et al., 2003) and WordNet[4] (Miller, 1995) 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. A semantic frame is a conceptual structure describing an event or relation and the participants in it.

A sample of images from the imSitu dataset and their annotations can be found in Table 1. Every situation in imSitu is described with one of **504 candidate verbs** such as *cooking*, *fixing*, *falling*, *opening*, *attaching* and so on. Each verb

[3] The FrameNet database contains over 1200 semantic frames. A semantic frame is a description of a type of event, relation, or entity and the participants in it.

[4] WordNet is a lexical database of English. Words are grouped into synsets (sets of synonyms), each expressing a concept. These concepts are connected by means of conceptual-semantic and lexical relations forming WordNet.

Abstract definition from imSitu dataset	Sample Generated Question Templates	Reponse Frame Element
An AGENT cooks a FOOD in a CONTAINER over a HEATSOURCE using a TOOL in a PLACE.	Who is cooking? What does the AGENT cook with TOOL? What is the AGENT doing ? What does the AGENT use to cook in CONTAINER ? Where does the AGENT cook FOOD in CONTAINER ?	AGENT FOOD VERB TOOL PLACE
The AGENT buys GOODS with PAYMENT from the SELLER in a PLACE	Who is buying GOODS ? What is the AGENT doing ? What item does the AGENT buy with PAYMENT ? Who does the AGENT buy GOODS from? Where does the AGENT buy GOODS ?	AGENT VERB GOODS SELLER PLACE
An AGENT catches a CAUGHTITEM with a TOOL at a PLACE.	Who catches at PLACE ? What is the AGENT doing ? What item does the AGENT catches with TOOL Where does the AGENT catches CAUGHTITEM ?	AGENT VERB CAUGHTITEM PLACE
The AGENT opens the ITEM with the TOOL at the PLACE.	What does the AGENT use to open ITEM ? Who opens ITEM ? What item does the AGENT opens ? Where does the AGENT opens ITEM with TOOL	TOOL AGENT ITEM PLACE

Table 2: A subset of Question Answer templates generated for cooking, buying, catching and opening.

has a set of FrameNet related frame elements. For example S_r*(cooking)* ={ *AGENT, FOOD, CONTAINER, HEATSOURCE, TOOL, PLACE* } provides the set of semantic frame elements of the verb *cook*. The 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"*. As another example S_r*(buying)* ={ *AGENT, GOODS, PAYMENT, PLACE* } includes a set of semantic frame elements of the verb *buy*. The abstract definition is : *"the AGENT buys GOODS with PAYMENT from the SELLER in a PLACE"*. Table 1 shows sample image annotations of some verbs such as *cook* and *buy*. The interested reader may refer to the imSitu online browser in order to explore the dataset. [5]
imSitu includes 190 unique frame elements, some shared among verbs such as *AGENT* and *TOOL*, while some are verb-specific such as *PICKED*∈ S_r*(picking)*. Every image is labeled with one of the 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 roles per verb on average.

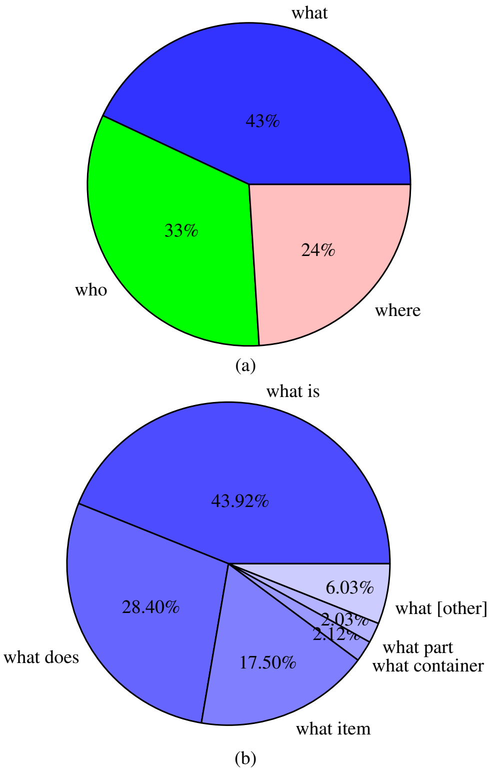


Figure 2: Distribution of questions in templates. (a) covers all questions while (b) includes questions starting with question word "what"

3.1. Question answer template generation

The main idea behind question template generation is to ask question about one of the frame elements of a given verb based on its abstract definition. For example a question about *cooking* can ask about *AGENT*, *FOOD*, *CONTAINER*, *HEATSOURCE*, *TOOL* or *PLACE*. [6] Each frame element requires a relevant question word to be used. Consequently, we mapped every frame element to a question

[5] `http://imsitu.org`

[6] The only exception is question template "What is the AGENT doing?" that the response frame element is labeled with VERB.

Question Word	Frame Elements
Who	*COMPETITOR, VICTIM, LISTENER, INDIVIDUALS, MOURNER, FOLLOWER, COAGENT, VOTEFOR, PERFORMER, EXPERIENCER, TICKLED, SELLER, EATER*
Where	*PLACE, TARGET,ADDRESSEE, SURFACE, GROUND, END, SOURCE, SHELTER, SURFACE, RECIPIENTS,CONTAINER, GOAL, STAGE, SCAFFOLD*
What	*OBJECT, HUNTED, BORINGTHING,FOCUS, OCCASION, SUBSTANCE, CLOTH COMPONENTS, DEPICTED, REFERENCE, AGENTTYPE, FOOD, CENTER, CLOTH*
What item	*ITEM, SIGNEDITEM, CAUGHTITEM, TURNEDITEM, GOODS, HIDINGITEM DRENCHEDITEM, REMOVEDITEM, DEFLECTEDITEM, WRAPPEDITEM*
What part	*PART, BODYPART, YANKEDPART, VICTIMPART, ITEMPART, RECIPIENTPART, AGENTPART, OBJECTPART, COAGENTPART*
What [Frame Element]	*VEHICLE, CONTAINER, SKILL, SHAPE, PATH, LIQUID IMITATION, MATERIAL, INSTRUMENT, PHENOMENON, OBSTACLE, EVENT*
What does the [AGENT] use to	*CROWN, BRUSH, CONNECTOR, GLUE, WRAPPINGITEM, COMPONENT, LOCK, COVER, DYE, PARACHUTE, ACTION, SEALANT*

Table 3: A subset of frame elements and the question words they are mapped to.

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*. Table 3 shows a subset of frame elements and the question word they are mapped to.

As shown in the first column of Table 2, in imSitu, each verb is described by an abstract statement including all its frame elements. Therefore, there are 504 abstract definitions in total. An abstract definition defines a natural form of how prepositions and punctuations are used along frame elements. 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"*. We can easily segment the statement to *"[an AGENT] cooks [a FOOD] [in a CONTAINER][over a HEATSOURCE] [using a TOOL] [in a PLACE]"*. Now in order to ask a question about a specific frame element, we hold out its segment. For example 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]"*. Then, we should decide which other segments should be included in the question. The only exception is *AGENT* and it will always be included (with article *the AGENT*) if not held out as response frame element. Approximately, we considered all possible subsets of segments. For example: *"[an AGENT] cooks"* and *"[an AGENT] cooks [X] [in a CONTAINER][in a PLACE]"* are two possible combinations when *FOOD* is the response frame element. Finally the relevant question word is appended at the beginning of each combination and verb form is modified accordingly. For example: *"What does the AGENT cook?"* or *"What does the AGENT cook in CONTAINER in PLACE?"*. A subset of question templates and their response frame elements for *cooking*, *buying*, *catching* and *opening* are shown in Table 2. In total, 6879 templates are generated, with on average 13.65 question-answer templates per verb. Figure 2 shows the distribution of template questions in terms of question words.

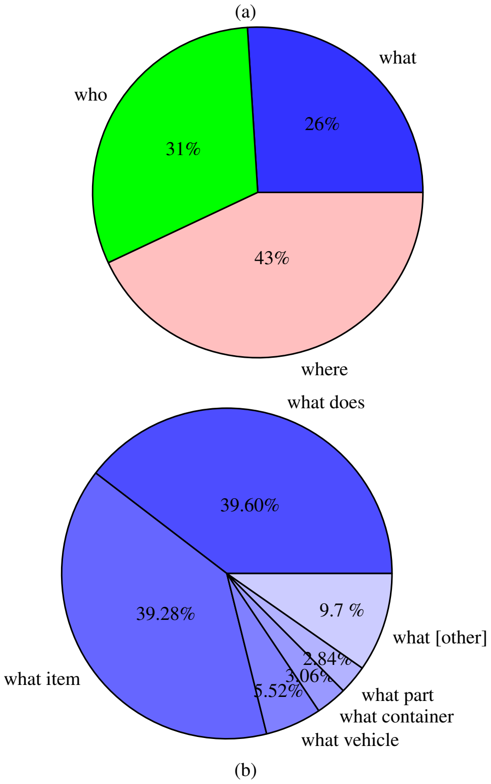


Figure 3: Distribution of questions in imSituVQA. (a) covers all questions while (b) includes questions starting with question word "what"

3.2. Question answer pair realization

The previous step generates templates for all 504 candidate verbs. As each image in imSitu is annotated with

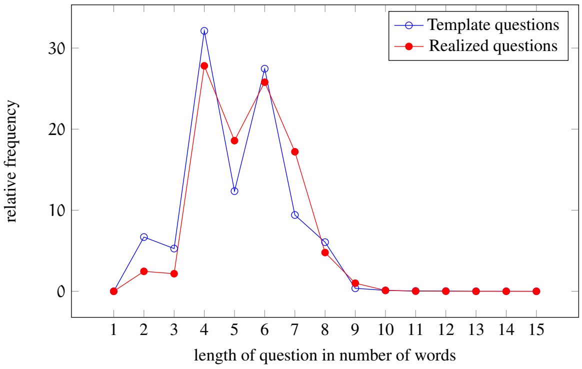
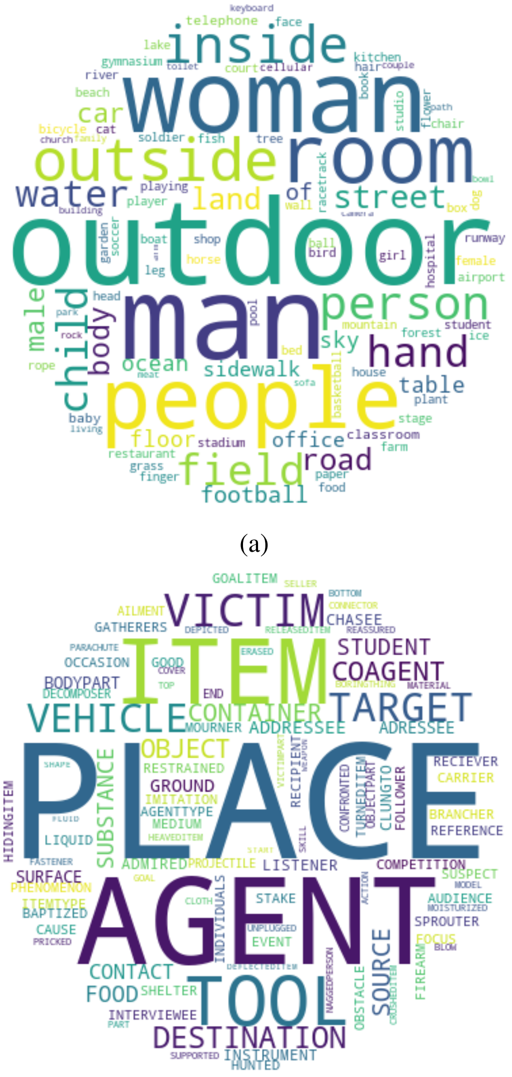


Figure 4: Distribution of template questions vs realized questions based on length.

one verb, the templates of the annotated verb are considered for the image. Templates that include frame elements that are missing or empty in the image annotation are excluded. Then, given each question template and response frame element, the frame elements are filled with the noun values from the annotation. This realization process can be applied to all imSitu images. The final dataset is called imSituVQA. Each sample in imSituVQA is a <*image,question*> input pair that is labeled with an <*answer*> as output. For example given the image about *cooking* from Table 1, applying the realization process on *"What does the AGENT cook in CONTAINER in PLACE?"* results in *"What does the boy cook in wok in kitchen?"*. Realizing the response frame element *FOOD* results in *"meat"* as answer. These three items compose a sample (<*image,question*>:<*answer*>) for VQA task. Table 4 shows VQA samples for *cooking*, *buying*, *catching* and *opening*. As can be seen, the dataset not only includes the typical question answer pairs but also frame element annotations as well.

If a verb has n templates, applying an image annotation results in n real <*question, answer*> samples of the image. 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 5. Table 6 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 5 visualizes the relative frequency of answers and response frame elements in terms of word clouds. The questions are mostly between 4 to 7 words. Figure 3 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. Figure 4 depicts the distribution of template questions and realized questions lengths in terms of the number of words. The distributions are very similar, showing the majority of questions are 4 to 7 words.



(a)

(b)

Figure 5: imSituVQA word clouds of (a) answers and (b) frame elements.

4. Conclusion

In this paper, we explained how we used the imSitu annotations to build a VQA dataset with verb semantic information. The goal is to enhance the VQA language processing component especially for questions describing events via verbs. Integrating or training VQA models with semantic frame information remains as a research problem to be explored.

One important question is how the VQA model performs on imSituVQA. We did experiment with VQA task on imSituVQA, and we quickly sketch some results here, Since the main goal of this paper was to discuss the process of imSituVQA creation. 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. Training the CNN-LSTM model proposed in (Antol et al., 2015a) results in 39.58% accuracy.

IMAGE about cooking

IMAGE about buying

QUESTION	**ANSWER**	**QUESTION**	**ANSWER**
Who is cooking ? VERB	boy AGENT	Who is buying shoes ? VERB ITEM	woman ? AGENT
What does the boy cook with spatula ? AGENT VERB TOOL	meat FOOD	Where does the woman buy shoes ? AGENT VERB GOODS	shoe store PLACE
Where does the boy cook meat in wok ? AGENT VERB FOOD CONTAINER	kitchen PLACE	Who does the woman buy shoes from ? AGENT VERB ITEM	person SELLER

IMAGE about catching

IMAGE about opening

QUESTION	**ANSWER**	**QUESTION**	**ANSWER**
What is the bear doing ? AGENT	catching VERB	Who opens the door VERB ITEM	cat AGENT
Where does the bear catch fish ? AGENT VERB CAUGHTITEM	body of water PLACE	What does the cat use to open the door ? AGENT VERB ITEM	paw TOOL
What item does the bear catch ? AGENT VERB	fish CAUGHTITEM	What item does the cat open ? AGENT VERB	door ITEM

Table 4: imSituVQA dataset samples about cooking, buying, catching and opening. The imSituVQA dataset includes frame element annotations for each question answer pair.

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 5: 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 6: Top 10 frequent answers in imSituVQA training samples.

Acknowledgements

This work is partially supported by a UIC University Scholar award to Di Eugenio

5. References

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., and Parikh, D. (2015a). Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

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

Fillmore, C. J., Johnson, C. R., and Petruck, M. R. (2003). Background to framenet. *International journal of lexicography*, 16(3):235–250.

Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., and Parikh, D. (2017). Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. (2017). Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE.

Kafle, K. and Kanan, C. (2017). Visual question answering: Datasets, algorithms, and future challenges. *Computer Vision and Image Understanding*, 163:3–20.

Kipper, K., Korhonen, A., Ryant, N., and Palmer, M. (2008). A Large-Scale Classification of English Verbs. *Journal of Language Resources and Evaluation*, 42(1):21–40.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollr, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Malinowski, M. and Fritz, M. (2014). A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in neural information processing systems*, pages 1682–1690.

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

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–105, March.

Ren, M., Kiros, R., and Zemel, R. (2015). Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961.

Shen, D. and Lapata, M. (2007). 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)*.

Silberman and Fergus. (2012). Indoor segmentation and support inference from rgbd images. In *ECCV*.

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

Wang, P., Wu, Q., Shen, C., Dick, A., and van den Hengel, A. (2018). Fvqa: Fact-based visual question answering. *IEEE transactions on pattern analysis and machine intelligence*, 40(10):2413–2427.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics.

Yatskar, M., Zettlemoyer, L., and Farhadi, A. (2016). Situation recognition: Visual semantic role labeling for image understanding. In *Conference on Computer Vision and Pattern Recognition*.

Zhu, Y., Groth, O., Bernstein, M., and Fei-Fei, L. (2016). Visual7w: Grounded question answering in images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4995–5004.