A Tale of Two Encodings: Comparing Bag-of-Words and Word2vec for VQA

Berthy Feng
Princeton University '19
bfeng@princeton.edu

Divya Thuremella
Princeton University '18
divyat@princeton.edu

Abstract

In this paper, we analyze the effect of two text encoding methods on a VQA classifier model. We compare a naïve bag-of-words encoding with a semantically meaningful word2vec encoding. We evaluate performance through ablation studies, manipulation of text and image inputs, and visualization of textual and visual attention. We find that the word2vec-based model learns to utilize both textual and visual information, whereas the bag-of-words-based model learns to rely more on textual input. Our analysis methods and results provide insight into how VQA models learn depending on the types of inputs they receive during training.

1. Introduction

Visual question answering (VQA) is the task of answering a question about a given image. Many baseline VQA methods employ the following general approach:

1. Extract text features from the input question.
2. Extract visual features from the input image.
3. Train a classifier that takes text features + visual features as input and outputs a probability distribution across answers.

Surprisingly, a naïve bag-of-words text input achieves impressive performance on the VQA dataset [1], outperforming more complicated text encoding methods [9]. We propose that using word2vec as the text input should improve performance by incorporating the semantic meaning of the question.

Our work compares a bag-of-words encoding and word2vec encoding for VQA performance. In doing so, we provide insights into why a bag-of-words model performs so well and how the text encoding method impacts what the model learns.

2. Related Work

iBOWIMG [9] provides a good baseline for VQA. When compared with nine other VQA methods, including an LSTM-based embedding, iBOWIMG outperforms most on both open-ended and multiple-choice questions. Our work aims to understand *why* a model like iBOWIMG performs so well.

iBOWIMG uses a learned embedding layer, which does not take advantage of NLP word embedding methods, such as word2vec [4] and GloVe [5]. Our work suggests using word2vec for text encoding and explains why and how this impacts performance. Research in NLP has compared the effectiveness of embedding methods for encoding semantic meaning, but we provide an in-depth analysis of the effect of text encoding methods on VQA specifically. Our analysis is different from evaluating the encoding method itself because our goal is to understand how the encoding method influences both semantic and visual understanding.

Since iBOWIMG, state-of-the-art VQA models have emerged, including those using bi-directional LSTMs [3] and neural module networks [2]. However, we believe there is still work to be done to fully understand and explain the baseline model.

3. Project Overview

3.1. Implementation

Classifier architecture. We borrow from the classifier architecture used in iBOWIMG [9]. The input to the network is a concatenation of the word feature and image feature. The input goes through a fully-connected layer and then a softmax layer, as shown in Figs. 1 and 2. The output vector size is the number of words in the answer vocabulary, and the word corresponding to the most probable class is the predicted answer.

The input image feature is the 4096-dimensional feature vector extracted from the fc2 layer of a VGG16 net [7] trained on ImageNet [6]. The textual features are described next.

Bag-of-words. The bag-of-words (BOW) encoding of a question is the sum of the one-hot encoding vector of each

word in the question. The size of the encoding vector is the number of words in the vocabulary. Fig. 1 is a diagram of the BOW-based classifier.

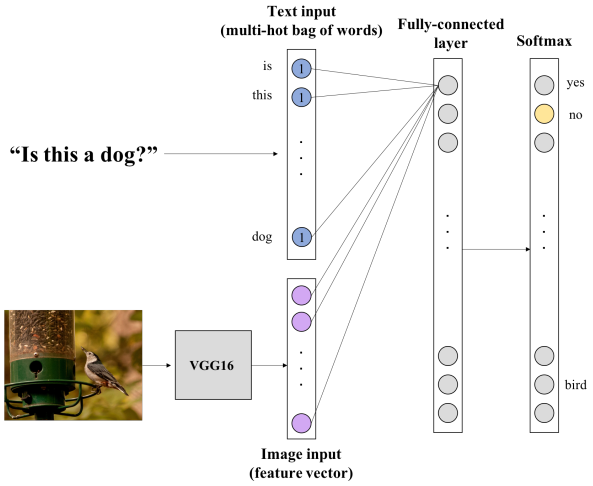


Figure 1: BOW classifier model

Word2vec. The word2vec (W2V) embedding of a question is created by using a two-layer network with a 300-dimensional hidden layer that becomes the encoding of the word. We use a word2vec embedding matrix pre-trained on the Google News dataset. To form the question feature, we get the word2vec embedding of each word in the question and sum the embeddings. Fig. 2 is a diagram of the W2V-based classifier.

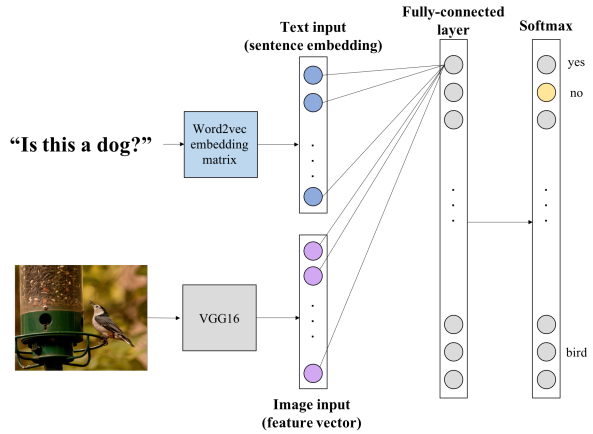
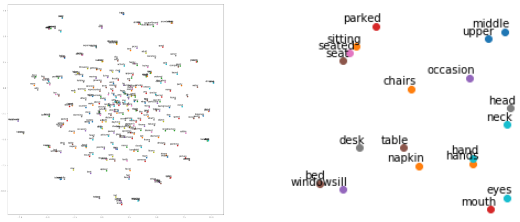


Figure 2: W2V classifier model

Whereas BOW is a naïve encoding, W2V encodes semantic relationships and analogies between words: words that are closer in the 300-dimensional space are more semantically related, and words that are analogous have similar distance vectors. Fig. 3 provides a visualization of the relationships that W2V encodes in the VQA vocabulary.



(a) Full space (b) Zoom-in

Figure 3: Projection of 300-d word2vec space in 2-d, using words from the VQA dataset. Words that are similar in semantic meaning are grouped together.

3.2. Goals

Our goal is to compare two different text encoding methods on a baseline VQA classifier. More than simply comparing accuracy, we aim to understand *how* and *what* the model learns.

3.3. Evaluation Methods

We train both models on the VQA train2014 set and evaluate on the val2014 set [1]. We use three general evaluation approaches:

1. Ablation studies to understand how the model depends on each input type.
2. Manipulating text and visual input to understand how the model responds to altered inputs.
3. Extracting textual and visual attention to understand which input features activate the predicted answer.

3.4. Insights

We found that word2vec teaches the VQA model to better integrate textual and visual input. We show this in our ablation studies (Section 4) and manipulations of the inputs (Section 5).

We also propose novel methods to evaluate VQA models: semantic precision (Section 4) and a variant of class activation mapping (Section 6).

4. Measuring Accuracy and Semantic Precision

In this section, we compare performance using top-1 accuracy and investigate the impact of removing either the visual or textual input. We also propose a metric called "semantic precision."

4.1. Top-1 Accuracy

We evaluate both models on val2014 using top-1 accuracy. The W2V+IMG classifier achieves an overall accuracy of 35.2% compared with 33.7% for BOW. W2V does

better on most types of questions answered by a noun (i.e., "what" questions). In general, BOW recognizes "yes/no" questions better than W2V. Table 1 provides accuracy comparisons for the overall dataset and for specific question types.

Question Type	BOW	W2V
Overall	33.7	**35.2**
"What is the"	0.1	**30.2**
"What are the"	0.0	**10.9**
"What is this"	**13.5**	0.0
"What animal is"	**71.8**	0.0
"What kind of"	6.3	**14.9**
"What"	1.4	**6.4**
"Why"	0.0	**4.8**

Table 1: Top-1 accuracy (%) on val2014

4.2. Ablation Studies

The following ablation studies remove text or visual input and measure the resulting decrease in performance.

Text only. We perform text-only ablation studies by removing the visual input (replacing the 4096-d image feature vector with all zeros). Despite the lack of visual information, both BOW and W2V still perform relatively well, achieving 32.5% and 31.5%, respectively. This indicates that the text input provides the most necessary information. This shows that questions in the VQA dataset contain significant bias and are easy to learn based only on language priors. Both the BOW and W2V models learn to recognize question types extremely well, and in Section 5 we demonstrate that changing the input question type influences the type of answer that the networks predict. For example, changing "What does the label on the bottle mean?" to "Does the label on the bottle mean?" causes both networks to change their answer from a noun to "yes/no." Furthermore, the dataset contains a significant portion of "yes/no" questions (val2014 contains about 22.9% "yes" questions and 15.1% "no" questions), making it easy to guess the correct answer for this type of question.

Image only. For image-only analysis, we remove the text input by replacing the text feature vector with all zeros. As expected, accuracy declines significantly for both networks: BOW accuracy drops from 33.3% to 22.3%, and W2V accuracy drops from 35.0% to 19.1%. Interestingly, W2V suffers significantly more than BOW without text input.

Discussion. Our ablation studies suggest that the W2V-based classifier learns to use both textual and visual information, whereas the BOW-based classifier depends overwhelmingly on the text input. For both text-only and image-only, BOW outperforms W2V. However, when given both types of input, W2V achieves 35% accuracy compared to 33.3% by BOW. For text-only, BOW accuracy is high (32.5%). The BOW classifier depends mostly on the text input to make a prediction, whereas the W2V classifier learns to use both types of input.

When we looked more closely at the BOW predictions for image-only, we saw that BOW guessed "yes" 95.3% of the time and "no" 3.8% of the time. Therefore, given no information about the question type, BOW assumes that it is a "yes/no" question and guesses "yes." W2V, on the other hand, guessed "yes" 75.1% of the time and "no" 9.4% of the time when given no text. This suggests that the BOW classifier learns and memorizes the dataset bias, whereas W2V learns to utilize both its given inputs.

Model	Accuracy
BOW IMG+TXT	33.3
BOW TXT	32.5
BOW IMG	22.3
W2V IMG+TXT	35.0
W2V TXT	31.5
W2V IMG	19.1

Table 2: Ablation studies, measured by top-1 accuracy (%) on val2014

4.3. Semantic Precision

Model	Semantic Precision
BOW IMG+TXT	**57.6**
BOW TXT	61.8
BOW IMG	39.1
W2V IMG+TXT	**60.1**
W2V TXT	61.0
W2V IMG	31.7

Table 3: Semantic precision (%), the average semantic similarity between predicted answer and ground truth, on val2014

We propose another metric of VQA performance: semantic precision. This metric quantifies the semantic closeness between predictions and ground-truth answers. Semantic precision offers an alternative to top-1 accuracy performance. It is a more lenient metric because it rewards a method for correctly understanding the semantic meaning of a question even though the answer is not technically correct. Suppose that in response to the question "How many apples are there?" Method 1 answers "2" and Method 2 answers "on the table," and the true answer is "3." By accuracy, Method 1 suffers the exact same amount as Method 2,

but by semantic precision, Method 1 performs much better.

To calculate semantic precision, we use the pre-trained W2V embeddings and measure the distance between the normalized embedding vectors of two words. We rate the semantic similarity of these vectors between 0.0 and 1.0. For each predicted answer, we calculate the similarity between the prediction and ground-truth answer and determine the average semantic similarity score for both BOW and W2V. As expected, the W2V model outperforms BOW in semantic precision due to the semantic information encoded in W2V.

5. Altering Semantic and Visual Input

Next, we examine which parts of the semantic and visual inputs influence a model's prediction. The semantic importance of each question word can be examined by successively removing various key words from the question and observing how the answer changes in response. The importance of each object in an image can be examined by zeroing out the portion of the image that contains that object, extracting the VGG-16 feature vector of the new image, and evaluating how the network changes its prediction. The results of semantic and visual alterations to three different question-image pairs are shown and described below.

(a) Original image (b) Altered image

Figure 4: Images associated with the question "How many fruits and veggies are there?" and answer "lot"

Example 1. We first examined the image in Fig. 4(a) and corresponding question "How many fruits and veggies are there?" (answer: "lot"). BOW predicted "4," while W2V predicted "hundreds," which is closer to the correct answer. In both cases, the network knew to respond with a quantity.

We then altered the image by zeroing out the main focus of the image (i.e., the part with the fruits at the bottom). As a result, BOW predicted "5," while W2V predicted "contest." Interestingly, BOW still predicted a number, but W2V no longer predicted a quantity. This suggests that changing the image affects the W2V model more than it affects the BOW model.

Altering the question to remove key words also changed the predicted answers. When we removed words to form the questions "many fruits and vegetables are there" and "fruits and vegetables are there," both BOW and W2V predicted "yes." Note that "yes" is technically correct for these altered questions because word order is not encoded in either model.

When we removed "fruits and vegetables" from the sentence to form "how many are there," BOW still predicted the "4," whereas W2V predicted "2." This shows how important the question words are to both BOW and W2V. Both methods depend strongly on the words indicating the type of question.

However, BOW depends on the question type more than W2V does. When phrases like "many" and "are there" were removed to form the questions "how fruits and vegetables are there" and "how many fruits and vegetables," W2V predicted "oranges" for both questions, indicating that it was concentrating on the "fruits and vegetables" phrase and connecting it with the oranges found in the image. This demonstrates that when the question type is ambiguous, W2V reverts to using visual information. Meanwhile, BOW did not revert to using visual information. BOW predicted "yes" for the question "how fruits and vegetables are there" and "4" for the question "how many fruits and veggies." This shows that BOW was not concentrating on "fruits and veggies" in the original question, but was instead trying to categorize the question into a yes/no or quantity type of question. For the question "how fruits and veggies are there," BOW seems to interpret it as the question "are there fruits and veggies" so answers "yes." Note that both the words "how" and "many" are necessary for BOW to consider the question a quantity question (we further demonstrate this in Section 6).

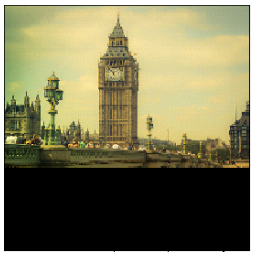
Example 2. We next examined the image in Fig. 5(a) and the question "What time does the clock read?" (answer: "11:55"). BOW predicted "cloudy," while W2V predicted "4:00." The W2V prediction is very close to ground truth. It seems that W2V associated the words "time" and "clock," as well as the image of the clock, with the answer "4:00."

When we altered the image to hide the clock tower, BOW predicted "4:25" while W2V predicted "bridge." This time, BOW recognized the type of question, while W2V didn't. Interestingly, BOW recognized the question type only when the object in the question was hidden, which suggests that BOW gives more importance to the question words than to the image features. Conversely, W2V clearly gives more importance to the image because it predicted "bridge" when the tower was hidden. For W2V, seeing the clock tower was essential to recognizing that the question was asking for a specific time, but without the clock tower, W2V moved its attention to the next most salient object in the image, the bridge.

We then removed the bridge. As a result, BOW predicted "horse" while W2V predicted "big ben." Neither the BOW

(a) Original image

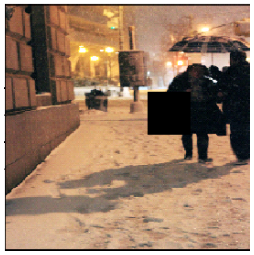
(b) Image without clock



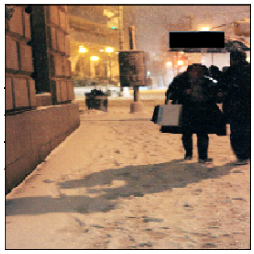
(c) Image without bridge

Figure 5: Images associated with the question "What time does the clock read?" and answer "11:55"

(a) Original image



(b) Image without bag



(c) Image without umbrella

Figure 6: Images associated with the question "What is over the woman's right arm?" and answer "bags"

nor the W2V network understood that the question was asking for a time. However, the W2V model clearly has a better understanding of the image because it correctly recognizes Big Ben.

Removing various words in the question provided some insight as to how the text influences the predicted answer. Similar to the above example, when we removed the first word to form the question "time does the clock read," both W2V and BOW answered "yes." Their prediction is technically correct for the altered question. Clearly, both W2V and BOW pay careful attention to the type of question. We also found that "read" does not seem to be an important word, as removing the word to make the question "what time does the clock" did not change either model's answer. Surprisingly, removing any of the middle words in the question ("time," "does," "the," "clock") caused both W2V and BOW to predict "cloudy." This indicates that these middle words are essential in making the W2V model understand that the question is asking for a time. In the absence of information about question type, both models reverted to simply recognizing the most salient part of the image (apparently the clouds). Perhaps the simplicity of the BOW-based model allows it to recognize only the most frequent question types, making it miss rare questions types such as time.

Example 3. We next examined the image in Fig. 6(a) and question "What is over the woman's right arm?" (answer: "bags"). Both models predicted "bag," which is extremely close to correct.

We first altered the image to remove the bag. As a result, BOW predicted "black" and W2V predicted "left." This suggests that both networks depend on the image content to some extent. Without being able to see the bag, the W2V model simply predicted a word that was similar to the question words. BOW, however, successfully focused its attention on the right side of the woman and saw the black square. The BOW prediction reminds us that zeroing out parts of the image may not be the best way of removing objects, as the zeros indicate the color black, not a lack of information.

Next we altered the image to remove the umbrella. In response, BOW predicted "bag" and W2V predicted "left." One explanation is that the BOW model knows only to look near the woman's right arm because it pays more attention to the question words than W2V does. W2V does not seem to understand what the woman's right arm means and simply predicts some word that seems similar to the question words. This is supported by examining which words influence the answer more (using the textual attention method outlined in Section 6). When we examined the textual attention of the two networks, we saw that the word that influenced the answer most for the BOW model was "item." The words that influenced the answer most for the W2V model were "side," "hand," "arm," and "lefthand." Therefore, BOW seems to pay more attention to the "what" part of the question, whereas W2V seems to pay attention to the more visually meaningful part of the question ("right arm").

We also tried removing some key words from the ques-

tion. When the altered question was "is over the woman's right arm," both methods predicted "yes" with high confidence, as expected. We also tried removing "over," "woman's," and "woman's right arm" to make the following three questions: "what is the woman's right arm," "what is over the right arm," and "what is over." For all three of these questions, BOW still predicted "bag." On the other hand, W2V predicted "umbrella." This, too, suggests that "what" is the key word for BOW, since changing any word other than "what" did not change the predicted answer. W2V, on the other hand, perceives other objects such as the umbrella, but seems to use the phrases "over" and "woman's right arm" to know where in the image to look.

To summarize these evaluations, we found that both the image and question influence both networks' answers. However, BOW pays more attention to the question words, specifically those that indicate the question type, while the W2V method generally relies more on the image features. Interestingly, even though BOW concentrates more on the question type, it seems that when the question type is "what," the image features determine the answer. For example, in Examples 2 and 3, both of which contained the word "what" in their questions, BOW's predictions were based on the image. This suggests that the BOW model modulates how much it should concentrate on the image based on the question type. Meanwhile, W2V also pays attention to the type of question, but it uses the image information to determine what other words to concentrate on. As shown in the results for Examples 1 and 2, when parts of the image related to specific question words were hidden, the answer given by the W2V model was gibberish. In Example 3, the W2V model perceived all the objects in the image (woman, bag, umbrella, etc.), but when the indicative phrases "over" or "woman's right arm" were removed, it chose an incorrect object. While the BOW model chooses which parts of the image to focus on based on the question type, W2V takes in the whole image as well as the question type and tries to match the rest of the words in the question accordingly.

6. Extracting Visual and Textual Attention

Inspired by the Class Activation Mapping (CAM) [8] method, we propose a method of extracting both visual and textual attention in the network. Traditional CAM works by taking the activated word (the argmax of the softmax output) and backpropagating it through the network [8]. For our VQA classifier, we also first get the activated word and focus on the weights tied to the activated word in the fully connected layer (the "activated weights"). For example, for the W2V-based classifier, the input size has $4096 + 300 = 4396$ dimensions, so there are 4396 activated weights. Of these weights, the first 300 are tied to the word input, and the last 4096 are tied to the visual input, as shown in Fig. 7(b).

6.1. Textual Attention

The goal of textual attention is to identify the "key word" that influences the network output. For example, if the predicted answer is "yes," we want to know which input word most significantly caused that output.

To extract the key word, we start with the activated class and extract the weights tied to the activated word to focus on the activated weights associated with the text input. For W2V, we take the first 300 activated weights, and for BOW, we take the first 5535 weights (there are 5535 question vocabulary words in the training set).

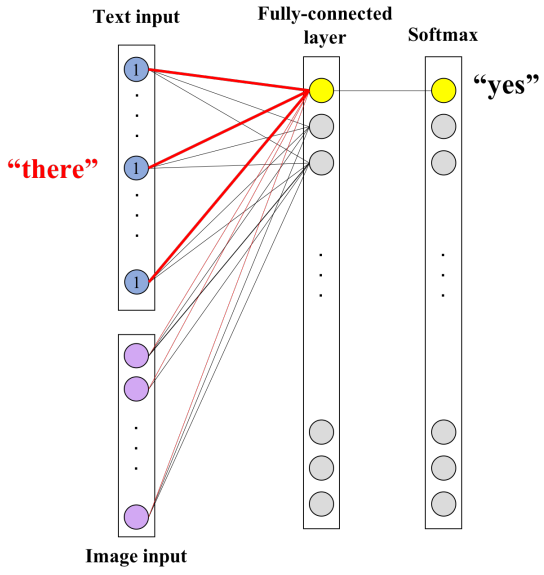
For BOW, we take the argmax of the 5535-d vector of activated text weights to identify the index of the most influential word in the input. Since the input is just a one-hot bag of words, the argmax of the activated text weights corresponds directly with the key word's index in the vocabulary.

For W2V, it is trickier to identify the key word, since the text input is a 300-d vector in the word2vec embedding space. We propose the following method: we treat the activated text weights, which themselves constitute a 300-dimensional vector, as a vector in the word2vec embedding space. We normalize the vector of activated text weights and calculate its 10 closest words. The intuition is that the weight on each of the 300 dimensions corresponds to the importance of that dimension on the network's answer, and since each dimension is in the word2vec semantic space, we can parse the semantic "meaning" of the weights, treating the weight vector as a word itself. Amazingly, this method produces highly interpretable results (Figs. 9 and 10).

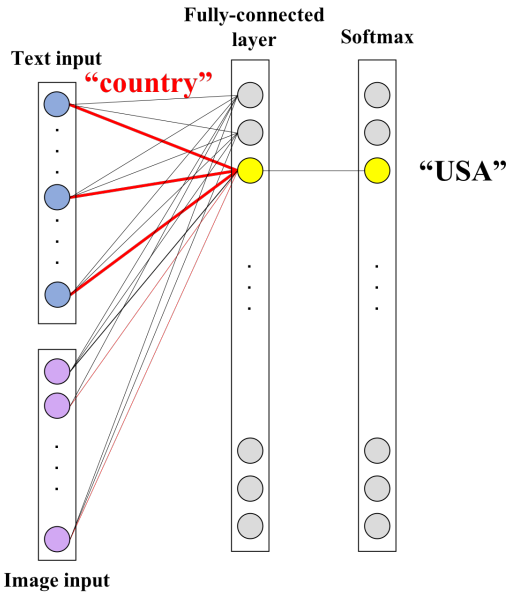
6.2. Visual Attention

To examine visual attention, we extract the activated weights associated with the visual input. We then take the argmax of these weights to get the "activated dimension" of the 4096-d image feature vector. Starting from the activated dimension, we backpropagate through VGG16 to the conv5 output. The conv5 layer uses 512 filters of size 7×7, and we take the filter with the largest weight (the "activated filter"). The activated filter is the filter with the most impact on the prediction of the VQA classifier. It is visualized as a 7×7 heatmap over the image. Fig. 8 provides an example.

Our visual attention mapping method currently produces rather coarse results. Further improvements can be made by averaging over the 512 filters (instead of taking the argmax) to generate a more detailed heatmap. We can do the same for the visual weights that influence the activated class in the VQA classifier; that is, instead of taking the activated dimension of the 4096-d feature vector, we can average over all 4096 dimensions and backpropagate through VGG16. While there are improvements to be made, we believe our visual attention mapping method provides valuable insight into the visual attention of the network.

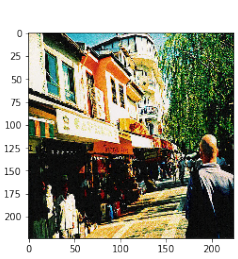


(a) BOW activating word

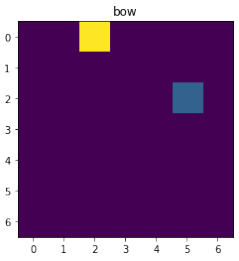


(b) W2V activating word

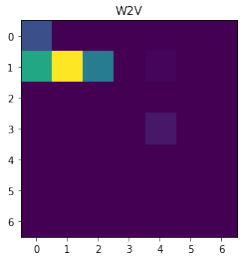
Figure 7: Visualization of our class activation mapping method to identify activating input words.



(a) Image



(b) BOW activated filter



(c) W2V activated filter

Figure 8: Activated filters for input image and question "Is this a Spanish town?" Both models predict the correct answer ("yes").

BOW. Most answers predicted by the BOW-based classifier are related to their "key words." For example, whenever BOW predicts "yes" as the answer, the activated input word is "there." This is because the word "there" occurs frequently in "yes/no" questions, such as those of the form "Is there..." Partially as a result of dataset bias, the BOW-based classifier learns to more probably predict "yes" if it sees the word "there" in the input question. Whenever BOW predicts a number, such as "3" or "2," the activated class maps back to the word "many." This is because most questions that are answered by a number contain the phrase "how many." This supports our insight that the BOW-based model heavily relies on the type of question.

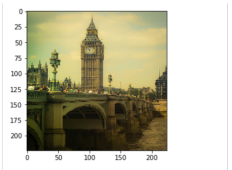
W2V. Extracting the activated word for the W2V-based model yields more mixed results. When the classifier predicts "yes," the activated word embedding is semantically close to random words in the word2vec dictionary, such as "Splendid_Friend." This could be due to the noise in the word2vec dictionary itself, since it was trained on the large Google News dataset. Another explanation is that the W2V-based model learns a more robust representation of the word "yes," where "yes" could be the answer to a more diverse set of questions than in the BOW method's understanding of the world. This would mean that in order to answer "yes," W2V requires more information (perhaps visual features) than just one word.

Interestingly, for other answers, the activated word embedding closely resembles the predicted answer. For example, when the W2V-based model predicts "2" or "3," the activated word embedding's 10 closest words include "many," "number," and "how." For the answer "USA," the activated word embedding is semantically closest to "country" and "flag." For the answer "beer," the activated word embedding is close to "drinking," "drink," and "beverage." Figs. 9 and 10 provide more examples.

Question: how many legs does the stool have
Answer: 4

	Prediction	Activating Word(s)
BOW	yes	there
W2V	2	many, number

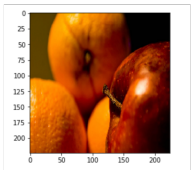
(a)



Question: what time does the clock read
Answer: 11:55

	Prediction	Activating Word(s)
BOW	cloudy	weather
W2V	4:00	time

(b)

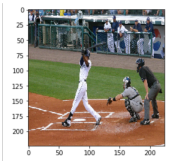


Question: what is the red fruit
Answer: apple

	Prediction	Activating Word(s)
BOW	flowers	on
W2V	apple	brand, fruit

(c)

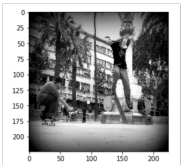
Figure 9: Examples of textual activation mapping. For each example, we provide the method's prediction and the activating word.



Question: what is the function of the man in the black shirt
Answer: umpire

	Prediction	Activating Word(s)
BOW	swinging	doing
W2V	umpire	man, balding/bearded, suspender wearing

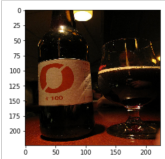
(a)



Question: what are they doing
Answer: skateboarding

	Prediction	Activating Word(s)
BOW	skateboarding	doing
W2V	skateboarding	doing, sport

(b)



Question: what does the label on the bottle mean
Answer: brand

	Prediction	Activating Word(s)
BOW	wine	drinking
W2V	beer	drinking, drink, beverage

(c)

Figure 10: More examples of textual activation mapping.

7. Conclusion

Our evaluations on the BOW-based and W2V-based classifiers confirm that semantically meaningful embedding vectors lead to higher accuracy. Our key insights, however, lie not in the marginal improvement in top-1 accuracy, but in an understanding of how the model behaves when using different text encoding methods. The W2V model learns to utilize both textual and visual inputs, recognizing most of the image features and then using textual information to inform its predictions. The BOW model, on the other hand, depends primarily on the question.

While the encoding method is a key difference between the W2V and BOW models, there are other factors to consider. The W2V embeddings are pre-trained on a much larger dataset, giving the W2V method an advantage due to transfer learning. Furthermore, the VQA dataset contains significant bias and is not the most robust dataset for visual question answering.

Though our analysis is based on a simple model, we believe that our insights provide ideas for the development and evaluation of future state-of-the-art VQA models.

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] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko. Learning to reason: End-to-end module networks for visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[3] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[4] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[5] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[6] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[7] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *CVPR*, 2014.

[8] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 487–495. Curran Associates, Inc., 2014.

[9] B. Zhou, Y. Tian, S. Sukhbaatar, A. Szlam, and R. Fergus. Simple baseline for visual question answering. *arXiv:1512.02167*, 2015.