Automatic Generation of Leveled Visual Assessments for Young Learners

Anjali Singh, Ruhi Sharma Mittal, Shubham Atreja
{ansingh8, ruhi.sharma, satreja1}@in.ibm.com
IBM Research AI

Mourvi Sharma[*]
mourvi.sharma@byjus.com
Think & Learn Pvt. Ltd.

Seema Nagar, Prasenjit Dey, Mohit Jain
{senagar3, prasenjit.dey, mohitjain}@in.ibm.com
IBM Research AI

Abstract

Images are an essential tool for communicating with children, particularly at younger ages when they are still developing their emergent literacy skills. Hence, assessments that use images to assess their conceptual knowledge and visual literacy, are an important component of their learning process. Creating assessments at scale is a challenging task, which has led to several techniques being proposed for automatic generation of textual assessments. However, none of them focuses on generating image-based assessments. To understand the manual process of creating visual assessments, we interviewed primary school teachers. Based on the findings from the preliminary study, we present a novel approach which uses image semantics to generate visual multiple choice questions (*VMCQs*) for young learners, wherein options are presented in the form of images. We propose a metric to measure the semantic similarity between two images, which we use to identify the four options – one answer and three distractor images – for a given question. We also use this metric for generating *VMCQs* at two difficulty levels – *easy* and *hard*. Through a quantitative evaluation, we show that the system-generated *VMCQs* are comparable to *VMCQs* created by experts, hence establishing the effectiveness of our approach.

1 Introduction

One of the very first ways children engage with the world is through images and visual cues (Ausburn and Ausburn 1978)). Developmentally, a young child's visual literacy, *i.e.*, the ability to make meaning from and communicate using images, forms the foundation of their later manipulations with language. It is imperative to consider this factor while designing assessments for children, so that they are engaging and can be used to appropriately assess their learning abilities. Visual Multiple Choice Questions (*VMCQs*) that ask children to choose the correct answer from a given set of images, are an ideal way of assessing children, and have indeed become common practice amongst assessments in the early childhood age group (Dunn et al. 2015). While designing such assessments, it is important to include questions at different difficulty levels, in order to measure children's learning progress as well as to cater to children with different learning proficiencies (Alsubait, Parsia, and Sattler 2013).

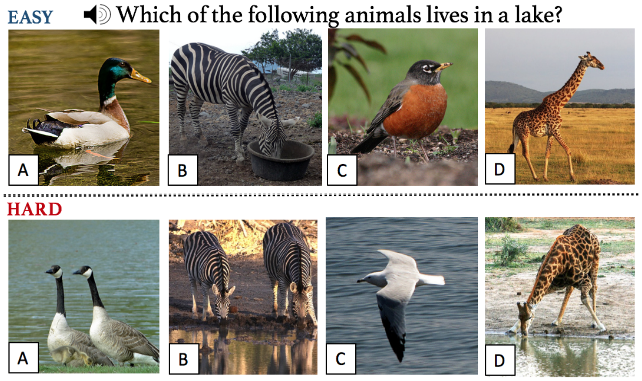


Figure 1: Example *VMCQ* at two different difficulty levels – *easy* and *hard* with correct answer A: Duck

The use of distractors that appear as plausible answers to a student who does not know the correct answer, helps in making an MCQ difficult. Hence, the level of similarity between answer and distractor options allows test-makers to control the difficulty level of MCQs to a great extent (Alsubait, Parsia, and Sattler 2013). For *VMCQs*, there are several factors that determine the level of similarity between the different options. Consider the example shown in Figure 1. The question is assessing a child's conceptual knowledge of animals and their habitat. Two variants of the question have been shown - *easy* and *hard*. In the *easy* variant, the correct option (A: Duck) shows a duck in a lake, while none of the animals in the distractor images are near a water body. In contrast, in the *hard* variant, all options show animals near a water body, making it difficult for a child to select the correct answer solely on the basis of the details in the images. It further requires the child to know unique features of the answer object (ducks in this case).

To further understand the process of creating *VMCQs* and the associated challenges, we conducted semi-structured interviews with five primary school teachers. From this study, we found that the difficulty level of *VMCQs* is highly dependent on the choice of option images, and also how the answer and distractor images compare to each other semantically. Furthermore, manually creating *VMCQs* requires considerable effort in selecting images, considering factors such as the images' clarity, quality and appropriateness for children.

[*] Work done while at IBM Research

While several methods have been proposed for automatically generating textual MCQs, none of them focuses on image-based assessments. Choosing an appropriate set of images as options for an MCQ is challenging, as it requires understanding and semantically comparing the candidate images. This can be addressed using image understanding techniques, such as image caption generation (Vinyals et al. 2017) and scene graph generation (Xu et al. 2017). The task of generating scene graphs, *i.e.,* structured representations of images showing the objects in them and the relationships between them, requires a large dataset of appropriate images annotated with scene graphs, thus incurring a huge cost. For real world applications, curating a dataset of images annotated with natural language captions is easier and has relatively less associated cost. Therefore, in our work, we generate image captions, and use them to determine the set of objects and relationships present in the images.

To summarize, in this work, we present a system that can automatically generate *VMCQs* suitable for young learners. Given a textual MCQ, our system selects the right set of images as options for the corresponding *VMCQ*. Based on our findings from the initial user study with teachers, we propose a metric for measuring the semantic similarity between two images using the objects in each image and the relations between them. We further use this metric to identify an appropriate set of answer and distractor images for a *VMCQ* at two difficulty levels - *easy* and *hard*. To evaluate our approach, we asked experts to rate system-generated and expert-created *VMCQs* on a 4-point Likert scale. We found that the ratings were comparable for both 78.3% times (with a difference of only one). Moreover, we found the difference in difficulty levels of the *easy* and *hard VMCQs* to be statistically significant ($p<0.0001$). Finally, through a qualitative study with teachers, we highlight the appropriateness of *VMCQs* generated by our system for young learners.

Overall, the two major contributions of this work are: (1) Given a textual MCQ, selecting candidate images from an image database as options to generate the corresponding *VMCQ*, and (2) Generating *VMCQs* at various difficulty levels. To the best of our knowledge, this is the first attempt to automatically generate visual assessments at varied difficulty levels suitable for young learners.

2 Related Work

Images and picture books support literacy in the young learners' classroom in multiple ways (Strasser and Seplocha 2007). Hence image-based questions are essential for assessing the learning rate of young children. The Peabody Picture Vocabulary Test (Dunn et al. 2015) is an established method for testing vocabulary knowledge in the early years, where an examiner reads out the question, and the learner points to one out of four possible pictures to select the correct answer option. Despite the popularity of visual assessments amongst young learners and the difficulty involved in manually creating such assessments, no attempt has been made yet to automate the process of generating them at scale.

A great deal of research has been done on generating MCQs with textual multiple choice options. Several works (Mitkov, LE AN, and Karamanis 2006; Lin, Sung, and Chen 2007) discuss methods to generate different types of MCQs using WordNet (Miller 1995). Other works (Al-Yahya 2014; Vinu and Kumar 2015) discuss ways to utilise domain ontologies for generating such questions. In our work, we use the method proposed by (Sharma Mittal et al. 2018) to generate the textual questions and options, where a subset of ConceptNet (Liu and Singh 2004), curated to be suitable for young learners, is used for generating MCQs.

Another important aspect while generating questions automatically is assessing their difficulty level. There have been some attempts such as in (Seyler, Yahya, and Berberich 2016; E V and Kumar 2017), which utilize machine learning based approaches to determine the difficulty level of a question. However, their notion of difficulty is associated with words and not images. In our work, we propose a method to assess difficulty levels on the basis of semantic similarities between the answer and distractor images in a *VMCQ*.

Several computer vision techniques have been proposed for describing images such as image caption generation (Vinyals et al. 2017; You et al. 2016), scene-graph generation (Li et al. 2017; Xu et al. 2017), *etc.* As explained before, we first generate image captions, and use them to determine the set of objects present in an image as well as the relationships that exist between those objects. These relations are further used for finding the level of similarity between images and for a assigning difficulty level to a given *VMCQ*.

3 User Study

We conducted a user study with primary school teachers to understand the process of creating visual assessments for young children and the associated challenges. We interviewed five teachers (referred to as P_1, P_2, P_3, P_4, P_5) from different schools in India, who were experienced with teaching and assessing children of the age group 4-8. All teachers belonged to the age group of 25-46 and had a Bachelors or a Masters degree in Education. P_1, P_2 and P_4 had 2-5 years experience while P_3 and P_5 had 15-20 years of experience. As part of the user study, we asked questions along three dimensions, 1) process of creating assessments and effort involved, 2) importance of visual assessments and 3) difficulty levels in assessments and especially *VMCQs*. Our observations from the study are as follows:

Process of Creating Assessments & Effort Involved: All the teachers were regularly involved in creating assessments for children, on a weekly or monthly basis. The process of creating an assessment varied for different teachers, depending on the format which their school adhered to. Four teachers were required to create a digital document, while one teacher used a computer application provided by the school for the same. Two of the teachers regularly read out instructions while assessing children. According to P_5, *"Comprehending written instructions at that age is very difficult, hence we prefer reading them out"*. All teachers mentioned that images were regularly used as part of their assessments, which were mostly objective. Four teachers mentioned that 20-25% of the questions in their assessments were MCQs or *VMCQs*. All teachers took between 1-3 hours for generating one assessment. They mentioned that considerable amount of time was spent in finding the right set of images to be used in the assessments, especially since the images have to be appropriate for children: "*The amount of time it takes to generate an assessment is usually proportional to the number of images I have to put*" – P$_3$. All teachers used web search to find the relevant images and P$_3$ also used digital clip arts. The teachers also mentioned that creating *MCQs* can take even longer time as they have to be very careful with the choice of distractors.

Importance of Visual Assessments: All teachers mentioned that children learn easily from images and visual cues like flash cards and smart boards as they find images relatable and enjoyable. *"I always begin my lessons with a picture story on which I ask children to dwell upon"* – P$_5$. They concluded that since the learning process is largely visual in nature, image based assessments are important as they help children recall what they had learnt. Teachers also mentioned that despite their preference for visual assessments for young children, their assessments have more textual MCQs than *VMCQs* as creating is very time consuming.

Difficulty Levels in Visual MCQs: All teachers highlighted that they create assessment questions at 2-3 difficulty levels, since different children have different learning proficiencies: *"We have to ensure that every child can answer some questions but not all of them should be able to answer all questions."* – P$_1$. Most of them followed a standard distribution (60-70% easy and 30-40% difficult). When asked about their approach for varying the difficulty level of a *VMCQ*, all teachers mentioned that the choice of distractors is important. They highlighted two different ways to control the difficulty level: (i) changing the textual distractors (ii) changing the images used for the answer and the distractors. Some examples that were provided by P$_2$ and P$_1$, respectively: (i) Q: Identify the fruit that grows on trees – *Easy*: For answer image, show the fruit hanging from a tree & *Difficult*: Show image of the fruit without a tree, (ii) Q: Identify the mammal that can fly – *Easy*: Show images of land animals, and the answer image as a 'flying bat' & *Difficult*: Show images of flying birds as distractors. They felt that such variations in the options ensure that children understand the concept clearly and cannot answer a question simply by eliminating options. P4 also mentioned other ways to make a *VMCQ* difficult, such as choosing images with high visual similarity based on colors and other visual features.

Through these findings we conclude that:

- Images and visual cues play an important role in young learners' education. Hence, it is imperative that their assessments have a major visual component. As children are still learning to read and write at that age, objective assessments are most suitable for testing their conceptual knowledge.
- Including questions at various difficulty levels is a major requirement while designing assessments. But coming up with *VMCQs*, especially for 'hard' difficulty level, is a very time consuming task. Hence automating the process can be very helpful for teachers.
- The level of semantic similarity between answer and distractor images helps vary the difficulty level of *VMCQs*.

Hence, we propose a metric for measuring this semantic similarity and use the same for selecting the most appropriate image set as options for a *VMCQ*. Furthermore, we use this metric to quantify the difficulty level of a given *VMCQ*.

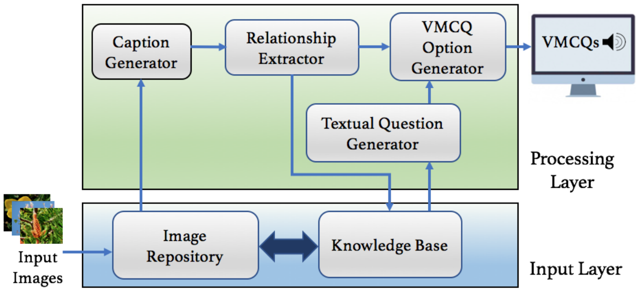


Figure 2: System Architecture and Flow Diagram

4 VMCQ Generation System

We propose a system capable of automatically generating *VMCQs* for assessing conceptual knowledge and visual literacy skills of young learners. Figure 2 shows the various components of our system and the flow of processes between them. The system consists of two components in the input layer and four components in the processing layer, which can be summarized as follows: 1) Knowledge Base – We use a subset of ConceptNet (Liu and Singh 2004), curated to be suitable for young learners, which has been proposed in (Sharma Mittal et al. 2018). In this knowledge graph called *YL-KB* (Young Learners' Knowledge Base) a node constitutes a word/phrase representing either an object or an attribute, an edge represents a relationship between the two nodes it is connecting, and the edge label represents the type of relationship, *e.g., isA, capableOf, atLocation, etc.*; 2) Textual Question Generator – The textual question generation component uses the Knowledge Base to generate different types of textual MCQs with textual answer and distractors using the approach proposed by Sharma Mittal et al.. We further associate a question-query with each generated MCQ which is used for finding the correct answer image; 3) Image Repository – It consists of all the images that can be used as possible options for the *VMCQs*. Any image dataset suitable for young learners can be used; 4) Caption Generator – This component generates short and descriptive captions for all images in the Image Repository using a neural image caption generator (Vinyals et al. 2017). Any image captioning model, that generates appropriate captions for images in the Repository, can be used; 5) Relationship Extractor – This component breaks down the generated captions into triples of the form ['subject', 'predicate', 'object'] to help in identifying the different objects and relationships present in each image, which are further used for semantically comparing different images; 6) *VMCQ* Option Generator – Given a difficulty level and using the question-query for a question as input, the *VMCQ* Option Generator outputs the best images to be used as options for the question. Finally, when the output of our system, is displayed, the question is rendered in audio format, using a standard text-to-speech tool[1] and images are shown as options to the *VMCQ* . In the following sub-sections, we present the methodology in more detail.

[1] https://www.ibm.com/watson/services/text-to-speech/

4.1 Relationship Extraction

To extract triples representing the objects and relationships between them from all image captions, we use a combination of two tools – Stanford Scene Graph Parser (Schuster et al. 2015) and Stanford-OpenIE API from Stanford CoreNLP (Manning et al. 2014). Both of these tools take as input single-sentence image descriptions and parse it into triples of the form [subject, predicate, object] depicting the object-relationships in the image. We analyze the triples outputted by the Scene Graph Parser and Stanford OpenIE and conclude that individually their recall is low – approximately 0.79 and 0.71 respectively. But when the triples are combined together, we get a much better recall – approximately 0.89. We leave out the detailed analysis as it is out of scope for discussion here. Hence, we use a combination of triples extracted by both the tools after removing all duplicates, and refer to them as 'primitive triples'.

The final set of triples should have the following properties: i) The subject and object entities should represent a single object (noun); ii) The predicate entity should be a single word which is either a verb, or a preposition. Keeping this in mind, we create a set of rules for extracting cleaner and more precise triples from the set of primitive triples. These rules are based on the Part-of-Speech (POS) tags of the words present in each caption. We use the Log-linear Part-Of-Speech Tagger from Stanford CoreNLP (Manning et al. 2014) to get the POS tags for all captions, following which we use these rules to extract the final set of triples:

- For subjects and objects in primitive triples: The noun words performing action or on which action is being performed are identified as the new subject/object, and other words are removed. *E.g.*, 'two donuts covered' ⇒ 'donuts'
- For predicates in primitive triples: Verbs and prepositions are extracted. In case of multiple verbs and/or prepositions, multiple triples are created from that primitive triple. When a noun is also present with the verb, another triple is created using that noun as the new object and the verb as the new predicate. *E.g.*, ['man', 'holding tennis racket on', 'court'] ⇒ ['man', 'holding', 'tennis racket'] and ['man', 'on', 'court']
- Dealing with quantificational modifiers: Given a set of triples, if one of the extracted triples has predicate 'of', subject belonging to the set: 'bunch', 'couple', 'group', 'herd', 'flock', 'pair' and object *Obj*, then for all those triples which have subject belonging to the same set but a predicate other than 'of', the subject is replaced with *Obj*.
- After all the subjects, predicates and objects have been processed using the above rules, they are lemmatized, such that all tokens are singular and the verbs are in the root form. *E.g.*, 'men' ⇒ 'man'; 'holding' ⇒ 'hold'.

4.2 Connecting Image Dataset to YL-KB

The *VMCQs* are generated by using the image dataset in combination with the knowledge base *YL-KB*. Hence, the vocabulary consisting of all subjects, objects and predicates in the triples needs to be mapped to the entities (nodes) and relations (edges) in *YL-KB*. To identify the list of all unique object categories in the image captions, we perform K-means clustering (Pedregosa et al. 2011) on the list of all subjects and objects occurring in the triples. Before the clustering, all words are converted to singular noun form. Out of the 331 object categories found, we picked 94 most frequently occurring object categories, which constitute the list of objects. Further, we divide all predicates into two categories – verbs and prepositions. A total of 18 prepositional predicates are identified which we group together into three categories based on how similarly they represent the relative location of an object with respect to another object. *E.g.*, ['up', 'over', 'around', 'down', 'outside'] are grouped together, hence represent a single prepositional predicate category. The predicate words which are verbs are grouped together using K-means clustering, following which 57 different verb predicate categories are identified. Finally, we have the Image Dataset Vocabulary (IDV) consisting of 94 object categories, 57 verb predicates and 3 prepositional predicates.

To map the objects and relationships in IDV to *YL-KB* vocabulary, we extract the subset of *YL-KB* corresponding to those nodes which consist of entities in IDV. Similarly, we extract those edges which have labels that belong to IDV. The, we identify words/phrases from IDV which have no corresponding node/edge in *YL-KB*. For such words/phrases, we find the *YL-KB* node entity which has closest word2vec[2] distance from the IDV term. We only keep the *YL-KB* entity if its word2vec distance from the IDV term is greater than a parameter α (whose value we determine experimentally), else we remove it from IDV. Images containing the removed term are also removed. This also ensures that any images containing inappropriate objects/relationships are removed. We finally use this subset of Image Repository, and a dictionary mapping IDV terms to *YL-KB* entities and relationships for generating *VMCQs*.

4.3 Question-query Formation

The next step is to generate textual MCQs suitable for young learners, and further use them for generating the *VMCQs*. We use the method presented by Sharma Mittal et al. for generating textual questions and their options. As proposed, we use *YL-KB* to find seed words which are either semantic categorical words such as 'animal', 'fruit', *etc.* or their child nodes, by employing graph techniques like high in-degree and low out-degree. Given a seed word and one or two functional properties, the natural language question is generated using a template based approach. Graph traversal techniques are used to find the correct answer and distractors such as finding children nodes of the seed word for correct answer and children nodes of siblings for distractors.

Once an MCQ is generated, to find a set of images to be used as options for the corresponding *VMCQ*, we associate structured queries called 'question-queries' with the MCQ. Question-queries are used for finding the best answer image by matching against the subjects, objects, predicates or a combination of them, associated with each image. The structure of question-queries depends on the type of textual MCQ that is generated. Overall there are three categories of textual questions, which are are as follows:

[2] https://code.google.com/archive/p/word2vec/

- **Type 1**: These questions test categorical knowledge related to a word. For Type 1, question-query is simply [‘answer’], where ‘answer’ is the correct textual answer to the question.
- **Type 2**: These are the first type of questions which test functional knowledge related to a word. They are further divided into two types. The functional relation for which Type 2(a) questions are created is ‘capableOf’, while for Type 2(b) it is ‘usedFor’. To generate the textual MCQ, if ‘action’ is the node connected to the seed word via the functional relation edge (in *YL-KB*), then question-query for Type 2(a) is [‘answer’, ‘action’] while for 2(b) it is [‘action’, ‘answer’].
- **Type 3**: These are the second type of questions which test functional knowledge related to a word. They test a learner’s functional knowledge for the ‘atLocation’ relation. Knowledge of ‘atLocation’ coupled with another different attribute, such as ‘travel’, ‘fly’, *etc.* can also be tested. For these questions, the node connected to the seed word via the ‘atLocation’ edge is a noun. The query associated with these questions is of the form [‘answer’, ‘relation’, ‘noun’], where ‘relation’ refers to the other attribute, if it exists, such as ‘fly’, ‘travel’, *etc.* In cases where only the ‘atLocation’ edge is being used, the relation term is set to ‘at’.

In Table 1, we show examples of question-query formation.

Type	Question	Answer	Question-query
1	*Which of the following is a fruit?*	banana	[banana]
2(a)	*Which of the following animals can fly?*	parrot	[parrot, fly]
2(b)	*Which of the following objects is used for playing?*	ball	[play, ball]
3	*Which of the following objects can fly in air?*	kite	[kite, fly, air]

Table 1: Question-query Formation Examples

4.4 VMCQ Option Generation

The approach for selecting an image from the repository, corresponding to a textual option, depends upon the type of MCQ generated and the difficulty level required. Selecting distractor images which are semantically similar to the answer image makes a question more difficult for children. On the other hand, if in the answer image, the context of the answer object is the same as the context being talked about in the question, then the question becomes easier to answer and even more easy if the distractor images show different actions/location from the answer image. Hence, we propose that the difficulty level of a question increases with a decrease in the semantic similarity between question-query and answer image. We also propose that the difficulty level increases with an increase in the semantic similarity between answer images and distractor images. We now define the following terms (related to a Visual MCQ V):
q: Question query for V
a: Correct answer image of V
d_i: Distractor images of V, where $i \in \{1, 2, 3\}$
$S_1(q, a)$: Semantic similarity between q & a
$S_2(I, J)$: Semantic similarity between images I & J
$S_2(a, d) = \text{avg}(S(a, d_1), S(a, d_2), S(a, d_3))$
Hence, we propose a measure of the difficulty level D for question V as follows:

$$D = f_1(1 - S_1(q, a)) * f_2(S_2(a, d))$$

where f_1 and f_2 are monotonically increasing functions of $S_1(q, a)$ and $S_2(a, d)$ respectively. Hence, by varying the values of f_1 and f_2, *VMCQs* at different difficulty levels may be generated. For the sake of simplicity, we only focus on two difficulty levels – *easy* and *hard*. We propose to use the Jaccard index for defining S_1 and S_2 and further explain this in detail later on in this section. Given two sets A and B, the Jaccard similarity J is given by $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$. Using Jaccard index as a measure ensures that when we get two similar images, they not only have multiple objects and relationships in common, but also the number of uncommon objects and relationships is low. It is to be noted that this difficulty measure is independent of the difficulty level of the textual MCQ using which the *VMCQ* is generated.

Answer Image Selection: Given an MCQ with four options and based on the question type, we use the question-query for finding the correct answer image. As the first step, we create tuples from all images in the repository such that their structure is the same as the question-query. Since question-query for type 3 are triples only, no change needs to be done to the image triples for those questions. For type 2(a) and type 2(b) questions, the object and the subject respectively need to be removed from the triples. For type 1 questions, both the predicate and object need to be removed from the triples so that the output tuples contain only the subject. *E.g.*, For the question: “Which of the following modes of transport can fly?” with correct answer ‘airplane’, the associated question-query is: [‘airplane’, ‘fly’]. Hence for an image in the repository which has triples: {[‘airplane’, ‘on’, ‘ground’], [‘airplane’, ‘next to’, ‘hangar’]}, the tuples created are: {[‘airplane’, ‘on’], [‘airplane’, ‘next to’]}. If π_I is the set of all tuples created for image I and R is the set of all images in the repository containing the correct answer object (such as ‘airplane’ in the above example), then the answer images a_e and a_h for difficulty levels *easy* and *hard* respectively, are found by:

$$a_e = \underset{I \in R}{\text{argmax}} \quad J(q, \pi_I); \quad a_h = \underset{I \in R}{\text{argmin}} \quad J(q, \pi_I)$$

Distractor Image Selection: For finding a distractor image for a given textual distractor, we define *Weighted Jaccard Index* which is a weighted mean of Jaccard indices measured between two images using different types of tuples. We first define these tuple sets for an image I which are used in the measurement of weighted Jaccard index:
T_I: Set of all unique triples for image I
S_I: Set of subjects & objects contained in the triples for I
D_I: Set of [subject, predicate] tuples contained in I’s triples
E.g., For an image with caption: “a man holding a tennis racket on a court” and triples: [‘person’, ‘at’, ‘court’], [‘person’, ‘hold’, ‘racket’] we have: S_I = {‘person’, ‘court’,

'racket'}, D_I = {['person', 'at'], ['person', 'hold']} and T_I = {['person', 'at', 'court'], ['person', 'hold', 'racket']}
Using these set of tuples, we define three different types of Jaccard indices between two images *I* and *M*:

$$J^1_{IM}= J(S_I, S_M);\ J^2_{IM}= J(D_I, D_M);\ J^3_{IM}= J(T_I, T_M)$$

The weighted Jaccard index J^W_{IM} between images *I* & *M* is:

$$J^W(I, M) = \lambda_1 J^1_{IM} + \lambda_2 J^2_{IM} + \lambda_3 J^3_{IM}$$

where values of λ_1, λ_2 and λ_3 depend on the image dataset and are determined experimentally.

Hence, for a *VMCQ* with answer image a, if K is the set of all images in the repository which do not contain the answer object and contain the distractor object, the distractor images d_e and d_h (corresponding to a textual distractor) for difficulty levels *easy* and *hard* respectively, are found by:

$$d_e = \operatorname*{argmin}_{I \in K} \quad J^W(a, I); \quad d_h = \operatorname*{argmax}_{I \in K} \quad J^W(a, I)$$

It can be observed that the Jaccard similarity can also be used for quantifying the difficulty level of *VMCQs*. This implies that $S_1(q, a) = J(q, \pi_a)$ and $S_2(a, d) = J^W(a, d)$. Hence, the difficulty level 'D' can be expressed as follows:

$$D = f_1(1 - J(q, \pi_a)) * f_2(J^W(a, d))$$

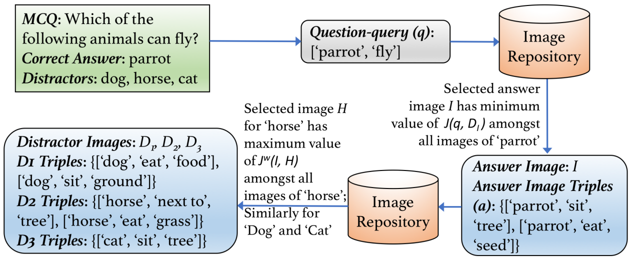

MCQ: Which of the following animals can fly?
Correct Answer: parrot
Distractors: dog, horse, cat
Question-query (q): ['parrot', 'fly']
Image Repository
Selected answer image *I* has minimum value of *J(q, D$_I$)* amongst all images of 'parrot'
Answer Image: *I*
Answer Image Triples (a): {['parrot', 'sit', 'tree'], ['parrot', 'eat', 'seed']}
Image Repository
Selected image *H* for 'horse' has maximum value of *J^w(I, H)* amongst all images of 'horse'; Similarly for 'Dog' and 'Cat'
Distractor Images: *D$_1$, D$_2$, D$_3$*
D1 Triples: {['dog', 'eat', 'food'], ['dog', 'sit', 'ground']}
D2 Triples: {['horse', 'next to', 'tree'], ['horse', 'eat', 'grass']}
D3 Triples: {['cat', 'sit', 'tree']}

Figure 3: Example of VMCQ generation process for a Type 2 textual MCQ at difficulty level *hard*

Figure 3 shows the method used for selecting the answer and distractor images for a Type 2 *VMCQ* at difficulty level *hard*. A similar approach is used for Type 1 & Type 2 questions using different question-queries and tuples as required. Figure 1 in Section 1 presents another example of the *easy* and *hard* variants of a *VMCQ* generated by our system.

5 Evaluation

We first carried out a quantitative evaluation by comparing the *VMCQs* generated by our system with those created by an expert. We also conducted a difficulty level validation test by comparing the difficulty levels of the *easy* and *hard VMCQs* generated by our system. For these experiments, we asked three primary school teachers (other than the participants of the user study) to evaluate. Furthermore, for qualitative evaluation, we collected feedback for our system from the five participants of the user study.

Experimental Setting: We used the MS COCO Image Dataset (Lin et al. 2014) as our Image Repository, which consists of images showing everyday objects across a diverse set of categories such as animals, modes of transportation, food, *etc.* For image captioning, we used the Show & Tell image captioning model (Vinyals et al. 2017) pretrained on the MS COCO Image Dataset. For mapping IDV words to *YL-KB* entities (Section 4.2), the value of α was set to 0.6. For the distractor image selection step (Section 4.4), we identified the best distractor images using the following values of λ_1, λ_2 and λ_3 respectively: 0.5, 1 and 0. On increasing the value of λ_1, distractor image selection gets more dependent on the set of objects contained in the images and less weightage is given to the relationships. Also, the value of J^3_{IM} was 0 for majority of the cases as probabilty of finding the same triples in two candidate options for a *VMCQ* is very low. Since all values of λ_3 were giving similar results, its value was set to 0. With this setting, we sampled 45 *VMCQ* sets uniformly distributed across the 3 types, where each set consisted of the *easy* and *hard* variants of the same textual MCQ. These samples were then utilized for carrying out all the evaluations.

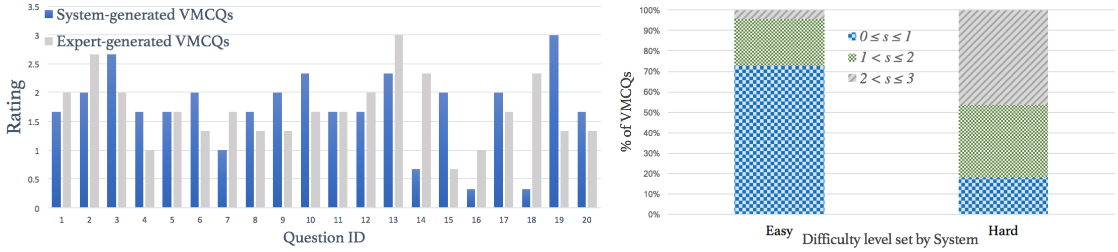
5.1 Quantitative Evaluation

For each of the following experiments, we asked three teachers to evaluate. As final scores, we report the average of the ratings given by the three evaluators.

Comparison with Human Generated VMCQs: For this experiment, we randomly sampled a set of 20 questions from the *VMCQs* generated by our system at *hard* difficulty level. We provided the corresponding set of 20 textual MCQs (along with the options) to an experienced teacher (expert), along with the following guidelines: 1) Create *VMCQs* for these questions by identifying the set of answer and distractor images based on the options; 2) Choose images such that the resulting *VMCQ* is difficult to answer for an average child. The teacher was provided with the same image dataset that we use as our image repository, consisting of separate folders corresponding to different object categories.

Evaluators were asked to rate these questions on a four-point likert scale $(0-3)$ on the basis of the quality and choice of answer and distractor images used for the question, given that the *VMCQ* should have difficulty level *hard* (0 = invalid *VMCQ* i.e. one or more images are inappropriate or there are multiple correct answer images, 1 = valid *VMCQ* which is easy to answer for an average child, 2 = valid *VMCQ* having moderate level of difficulty and 3 = valid *VMCQ* which is difficult to answer for an average child).

In Figure 4(a), we report the scores for each of the human-generated & system-generated *VMCQs*. The average rating for all human-generated *VMCQs* was 1.70 ($\pm$0.58), while for system-generated *VMCQs* it was 1.72 ($\pm$0.69). On conducting a paired t-test ($\alpha = 0.05$) on the scores received by human and system-generated *VMCQs*, we obtained a p-value of 0.93. This shows that there is a lack of evidence to show that there is significant difference between the two set of scores. Out of the 60 data points of comparison between human and system generated *VMCQs* (20 data points for each evaluator), 38.3% times both ratings were the same, and 78.3% times the difference in both ratings was in the range 0-1. This indicates that the quality and difficulty of the



(a)

(b)

Figure 4: (a) Scores for human-generated vs system-generated *VMCQs*; (b) Difficulty level validation statistics

system-generated *VMCQs* is comparable to that of *VMCQs* created by experts, given the same image dataset.

Difficulty Level Validation: For the 45 *VMCQ* sets of *easy* and *hard* variants of the same question, evaluators were asked to rate both *VMCQs* in a set, on a four-point likert scale $(0-3)$ on the basis of the difficulty level of the *VMCQ*, where difficulty (for an average child) increases as we go from 0 to 3 (0 = very easy *VMCQ*, 1 = easy *VMCQ*, 2 = moderately difficult *VMCQ*, and 3 = very difficult *VMCQ*).

Figure 4(b) shows the percentage of ratings received by the *easy* and *hard* *VMCQs* in the ranges $0 \leq s \leq 1$, $1 < s \leq 2$ and $2 < s \leq 3$, where s is the average of ratings given by all 3 evaluators for a *VMCQ*. As we can see, a significant majority ($\sim$72%) from the *easy* *VMCQs* received ratings in the range $0 \leq s \leq 1$. Similarly, majority of *hard* *VMCQs* got rated more than 1, where 35.5% had average score in the range $1 < s \leq 2$ and 46.6% got rated more than 2. The average scores received by *easy* and *hard* *VMCQs* were 1.07 (±0.56) and 1.98 (±0.80) respectively. On conducting a paired t-test ($\alpha = 0.05$), the results showed that the difference in the difficulty levels for the two sets is statistically very significant (p <0.0001). This indicates that there is significant difference in the difficulty levels of the *easy* and *hard* *VMCQs* generated by our system.

5.2 Qualitative Evaluation

We conducted interviews with the same primary school teachers who had participated in the user study. We showed them a random sample of 10 *VMCQs* generated by our system and asked questions regarding their quality and usefulness. Particularly, we asked them if they understood the concept being tested through the *VMCQs*, and whether they were appropriately tested the knowledge of young learners.

All teachers were able to understand the concept being tested through the *VMCQs*, and mentioned that they were appropriate and correctly tested the knowledge of children. On being asked about the quality of images used in the *VMCQs*, the teachers found most of them to be appropriate but mentioned that some images could be clearer for children to understand them properly, *e.g.*, P_1 & P_2 found an image of a boat to be unclear due to its irregular shape. P_3 also mentioned that cultural/geographical variations should be considered while choosing images of animals, modes of transport, *etc.*, so that children can find them relatable. As positive feedback, P_1 and P_4 mentioned that the choice of the answer and distractor images very good, especially for the *hard* *VMCQs*. But P_1 and P_2 also added that, to make the images more appealing, they should be brighter and more colorful. P_3 was surprised on finding out that the *VMCQs* were generated by our system automatically and found the system to be *"very promising for generating real assessments"*.

5.3 Discussion

Due to the complex, multi-dimensional nature of images, rating *VMCQs* is a highly subjective exercise. An individual's perspective on whether a *VMCQ* is easy/difficult or good/bad may be influenced by factors such as familiarity with the objects in an image, and in the case of educators, the unique and shifting perspectives gained from the children they engage with every day. As a result, there were few cases of disagreement between the evaluators regarding the quality/difficulty of a *VMCQ* while comparing with *VMCQs* created by experts . Furthermore, the difficulty of a *VMCQ* can not be evaluated independent of the level of difficulty of the corresponding textual MCQ. Hence, there were cases where the system (and experts also), were unable to generate a *VMCQ* deemed as difficult by the evaluators, since the textual MCQ itself was not hard enough. While our system is capable of generating suitable questions for young learners, curating a dataset of images that are very clear and apt for children can bring substantial improvement to our system.

6 Conclusion

We presented a system which can automatically generate *VMCQs*, i.e. MCQs with options given in the form of images. We used image semantics as the basis for generating *VMCQs* at two difficultly levels – *easy* and *hard*, using different techniques for selecting answer and distractor images. The results of the quantitative evaluation demonstrate that our system is capable of generating *VMCQs* that are comparable with those created by an experienced test-maker. Furthermore, the qualitative evaluation shows that our system can be very helpful for teachers who spend considerable amount of time in designing such assessments manually. As part of our future work, we intend to use sub-image analysis techniques to find which objects occupy how much visual area of an image to understand the saliency of different components for selecting better *VMCQ* options.

7 Acknowledgments

We would like to thank all the teachers for their contribution in the user study and the evaluation of our system.

References

Al-Yahya, M. 2014. Ontology-based multiple choice question generation. *The Scientific World Journal* 2014.

Alsubait, T.; Parsia, B.; and Sattler, U. 2013. A similarity-based theory of controlling mcq difficulty. In *e-Learning and e-Technologies in Education (ICEEE), 2013 Second International Conference on*, 283–288. IEEE.

Ausburn, L. J., and Ausburn, F. B. 1978. Visual literacy: Background, theory and practice. *Programmed Learning and Educational Technology* 15(4):291–297.

Dunn, L. M.; Dunn, D. M.; Lenhard, A.; Lenhard, W.; and Suggate, S. 2015. *PPVT-4: Peabody picture vocabulary test;[manual]*. Pearson.

E V, V., and Kumar, P. S. 2017. Difficulty-level modeling of ontology-based factual questions. *arXiv preprint arXiv:1709.00670*.

Li, Y.; Ouyang, W.; Zhou, B.; Wang, K.; and Wang, X. 2017. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1261–1270.

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

Lin, Y.-C.; Sung, L.-C.; and Chen, M. C. 2007. An automatic multiple-choice question generation scheme for english adjective understanding. In *Workshop on Modeling, Management and Generation of Problems/Questions in eLearning, the 15th International Conference on Computers in Education (ICCE 2007)*, 137–142.

Liu, H., and Singh, P. 2004. Conceptnet- a practical commonsense reasoning tool-kit. *BT technology journal* 22(4):211–226.

Manning, C.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S.; and McClosky, D. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 55–60.

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

Mitkov, R.; LE AN, H.; and Karamanis, N. 2006. A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering* 12(2):177.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.* 12:2825–2830.

Schuster, S.; Krishna, R.; Chang, A.; Fei-Fei, L.; and Manning, C. D. 2015. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the fourth workshop on vision and language*, 70–80.

Seyler, D.; Yahya, M.; and Berberich, K. 2016. Knowledge questions from knowledge graphs. *arXiv preprint arXiv:1610.09935*.

Sharma Mittal, R.; Nagar, S.; Sharma, M.; Dwivedi, U.; Dey, P.; and Kokku, R. 2018. Using a common sense knowledge base to auto generate multi-dimensional vocabulary assessments. In *EDM*.

Strasser, J., and Seplocha, H. 2007. Using picture books to support young children's literacy. *Childhood Education* 83(4):219–224.

Vinu, E., and Kumar, P. S. 2015. Improving large-scale assessment tests by ontology based approach. In *FLAIRS Conference*, 457.

Vinyals, O.; Toshev, A.; Bengio, S.; and Erhan, D. 2017. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE transactions on pattern analysis and machine intelligence* 39(4):652–663.

Xu, D.; Zhu, Y.; Choy, C. B.; and Fei-Fei, L. 2017. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2.

You, Q.; Jin, H.; Wang, Z.; Fang, C.; and Luo, J. 2016. Image captioning with semantic attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4651–4659.