Using a Common Sense Knowledge Base to Auto Generate Multi-Dimensional Vocabulary Assessments

Ruhi Sharma Mittal
IBM Research
Bangalore, India
ruhi.sharma@in.ibm.com

Seema Nagar
IBM Research
Bangalore, India
senagar3@in.ibm.com

Mourvi Sharma
IBM Research
Bangalore, India
mourshar@in.ibm.com

Utkarsh Dwivedi*
StudyPad
Bangalore, India
dwivediu@acm.org

Prasenjit Dey
IBM Research
Bangalore, India
prasenjit.dey@in.ibm.com

Ravi Kokku
IBM Research
Yorktown, US
rkokku@us.ibm.com

ABSTRACT
As education gets increasingly digitized, and intelligent tutoring systems gain commercial prominence, scalable assessment generation mechanisms become a critical requirement for enabling increased learning outcomes. Assessments provide a way to measure learners' level of understanding and difficulty, and personalize their learning. There have been separate efforts in different areas to solve this by looking at different parts of the problem. This paper is a first effort to bring together techniques from diverse areas such as knowledge representation and reasoning, machine learning, inference on graphs, and pedagogy to generate automated assessments at scale. In this paper, we specifically address the problem of Multiple Choice Question (MCQ) generation for vocabulary learning assessments, specially catered to young learners (YL). We evaluate the efficacy of our approach by asking human annotators to annotate the questions generated by the system based on relevance. We also compare our approach with one baseline model and report high usability of MCQs generated by our system compared to the baseline.

Keywords
Knowledge base, Vocabulary learning, Vocabulary Assessment, Assessment Generation, MCQ Generation, Personalized Vocabulary learning

1. INTRODUCTION
Personalized automated tutoring provides a scalable solution for augmenting in-class learning, and hence helps teachers effectively achieve increased learning outcomes in multi-student classrooms.

In automated tutoring, assessments play an important role, since they provide a way to continuously measure learners' level of understanding for a given concept. For young children, automatic vocabulary assessment is an interesting research problem and several efforts have been devoted to it [9, 17, 18, 23, 29]. It is a complex problem since word knowledge acquisition for first language learners is an incremental, continuous process, in part determined by the richness of a word's connection to other related words [5, 8]. This is important because the more associations a word has, the easier it is for learners to identify the meaning of the word when it is encountered again in a new context [7]. Hence, automatic assessment generation should strive to assess the multiple facets of a word, in the context of its relationships with other words.

Among the different assessment types, an MCQ test is a simple and highly scalable assessment mechanism, and is easily gamifiable for increased engagement by young learners. In this paper, we mainly focus on MCQ generation with a single correct answer and multiple distractors, although the solution is equally and trivially extensible to MCQs with multiple correct answers. There are three important parts of an MCQ, a) a Question Stem, b) a Correct Answer and c) one or more Distractors. For a young language learner, the scope of varying the question stem and the correct answer is limited, but distractors play an important role in determining the level and relevance of an automatically generated MCQ. Generating the right set of distractors for an MCQ is a difficult and tedious task even for humans. Hence, our main attention in this paper is on automatic generation of good distractors for MCQs.

We use ConceptNet5.4 [19] as a common sense knowledge base (KB) and generate a diverse set of MCQs for assessing conceptual understanding of a word. Using ConceptNet, however, leads to several challenges: 1) some of the links may not be appropriate to vocabulary learning for young learners, 2) there may be missing or sparse links for some words, and 3) there is no explicit information about word sense. To address these challenges, we first employ a supervised learning approach to filter out inappropriate links before generating MCQs. Second, we employ word embeddings [28] to overcome missing and sparse links. Third, even
though information on multiple meanings of the same word is not directly available in ConceptNet, we detect the presence of multiple meanings of words through varying independent relationships in ConceptNet based graph (e.g. seal–the animal and seal–the stamp have independent hierarchies of word relationships in the graph), and hence are able to generate questions which aim to assess knowledge of multiple meanings of a word. Hereafter, we refer to this curated ConceptNet for young learners as YL-KB (Young Learners Knowledge Base).

We evaluate the efficacy of our approach by asking human annotators to annotate the questions generated by the system based on the relevance and the automatic difficulty level assigned. We also compare our approach with two baseline models. We perform extensive evaluation on a set of 600 automatically generated questions. For relevance of the generated MCQs we report Fleiss Kappa [14] moderate (0.44) inter-annotator agreement.

The paper is organized as follows. We review the related work on question generation as it applies to MCQs in Section 2. We describe our design considerations, and approach along with the system architecture in Sections 3 and 4 respectively. We report the results of our evaluation in Section 5 and conclude in Section 6.

2. RELATED WORK

Prior research has mainly addressed MCQ generation from two dimensions, namely 1) utilizing text corpora and lexical resources such as WordNet [13] to generate question stem, correct answers and distractors, and 2) utilizing domain ontologies to generate domain specific MCQs. Some notable work utilizing WordNet [13] lexical resource for generating MCQs are [9, 17, 20, 18]. Brown et al. [9] generate different types of questions for a word, aiming to assess different aspects such as synonyms, antonyms, definition etc. The approach for choosing distractors is to pick words which have the same part of speech as the word in the question stem. Hoshino et al. [17] present different methods for generating distractors, such as mutual information and edit distance, using WordNet. Mitkov et al. [20] find keywords based on frequency of occurrences and create a question for a word based on the phrase it is occurring in. They use WordNet’s hypernym relationship to find distractors. Generation of MCQ distractors using WordNet for English language adjective understanding is discussed in [18]. Gates et al. [15] use definitions for words to generate a cloze type question for vocabulary building. They remove verb phrase to create cloze type question. For distractor generation, they employ a simple technique where phrases which have same structure as the answer phase are the potential distractors. Mostow et al. [21] propose automatic generation of multiple choice cloze questions to test a child’s comprehension while reading a given text. Unlike previous methods, it generates different types of distractors designed to diagnose different types of comprehension failure, and tests comprehension not only of an individual sentence but of the context that precedes it. More recent work aims to generate MCQs for any Wikipedia topic [16] and using DBpedia [27] fills the gap of generating MCQs for quiz-style knowledge questions from a knowledge graph such as DBpedia[6].

A number of papers utilize domain ontology for automatic question generation. Some notable works in this domain are [24, 3, 1, 4, 12, 2, 30], which address different aspects of automatic question generation from domain based ontology: 1) how to generate distractors; 2) how to control the difficulty of a question; 3) how to control the number of questions to be generated, since in a practical setting only a specific number of questions would make sense; 4) how to generate domain relevant questions and 5) limitations of using domain ontology for automatic question generation. Our paper advances the state-of-the-art in significant ways. It cuts across all different dimensions of generating MCQs for assessing vocabulary learning in young children, by using a common sense knowledge base with wider coverage but high noise. To the best of our knowledge, this is the first of its kind work that addresses the sophisticated task of automatically generating varying word knowledge assessments using techniques from diverse areas of knowledge representation and reasoning, machine learning, inference on graphs, and pedagogy. Further, using ConceptNet instead of WordNet provides significant advantages in terms of the number and diversity of word relationships available.
3. DESIGN CONSIDERATIONS

Our approach to MCQ generation builds on ConceptNet, a semantic network containing common sense knowledge (often stored as networks of related ideas) created to help computers understand the world. When a learner is assimilating information about words in a language, in a way they are trying to make a mental semantic network of words [22]. Since a common sense knowledge base mirrors this semantic network, it can potentially serve as a resource for generating robust, multi-dimensional assessments for vocabulary learning.

4. OUR APPROACH

In this section, we first present the definitions of terms we use throughout the paper. Then, we present the high level overview of our system for automatic MCQ generation from a common sense KB. A detailed explanation of each component of the system follows in further subsections.

4.1 Terminology

A common sense knowledge base (KB) is a directed semantic network of common sense entities such as words and phrases as shown in Figure 1. The entities in the network are connected with a diverse set of semantic relations such as isA, hasA, atLocation etc. We represent the semantic network as graph $G$, consisting of $V$ nodes and $E$ edges, where edge labels come from the conceptual relationship connecting the two nodes in a directed manner. We call relations representing functional characteristics such as hasA, usedFor, and capableOf functional relations, and isA as hierarchical relation. We represent the words and relations for which we create MCQs as seed words and seed relations, respectively. If there is a directed edge from node $c$ to node $p$ with relation $r$, we call $p$ as a parent of node $c$ with relation $r$ and $c$ as a child of node $p$ with relation $r$. The siblings of a node, with respect to a specific relation, are defined as all the children of its parent node, except for the node itself.

4.2 System Architecture

Our goal is to enable a holistic solution for automatic MCQ generation from a KB. The high level overview of our solution is depicted in Figure 2. The KB is curated for themes that are relevant for young learners and then filtered using the Children's Book Test [26] corpus, which is a dataset curated from an extensive selection of children's books. Further filtering is done to remove noisy and irrelevant edges. The curated KB, referred to as YL-KB is now free from inappropriate and noisy data, which makes it suitable to use for vocabulary assessments. The YL-KB is used to select seed words and generate all six types of questions. We now discuss each of these stages in detail.

4.3 KB Curation

The goal of this paper is to generate age-appropriate MCQs for vocabulary assessments catering to young language learners. Therefore, it is essential to remove the semantic relationships which are 1) inappropriate, 2) rare to observe and 3) inherently noisy from a KB. We handle this problem in a systematic way.

Theme Specific Retrieval and Filtering Based on Children's Book Test: First, we retrieve the part of the KB based on themes relevant to young learners such as fruits, animals, vegetables, transport, etc. Next, from this theme specific KB, we filter the edges where either the source node word or the target node word is part of the Children's Book Test.

Supervised Learning for Filtering Noisy and Irrelevant Links: We use a supervised learning approach to filter out noisy or irrelevant edges. This process begins with crowd-sourcing annotators to manually label the edges as relevant or not. After the manual annotation, we train a binary classifier on the annotated links. The features we pick are, 1) Edge relation, 2) Cosine similarity between source word and target word from word embedding vectors and 3) Weight or confidence score on the edges, if present. After this step, the curated KB is relatively free from noisy and inappropriate data and can be used for MCQ generation for YL.

In this section, we first present the method used for automatic seed word selection. Next, we present the strategy used for hard and easy MCQ generation.

4.3.1 Seed Words Selection for Each MCQ Type:
This process involves two steps. 1) Selecting words which are representative of semantic categories such as ‘mammal’, 'fruit', and 'bird', and 2) Selecting the child nodes of these semantic categories based on a criterion. We employ graph based heuristics to select words corresponding to semantic categories. Words that have a relatively high degree for incoming isA relations, and a relatively low degree for outgoing isA relations qualify as semantic categorical words. Next, we pick the child nodes of these semantic category words which have a relatively high number of edges for usedFor, capableOf and hasA relationships. Thus, we generate a list of seed words which we use to create MCQs.

### 4.3.2 Method for Handling Missing Edges:
As described earlier, the curated KB is processed to remove noisy and irrelevant data before MCQ generation. However, YL-KB still contains missing edges. Because of this, some nodes which are correct answers show up as distractors instead. For example, as shown in Figure 1, for question “Which of the following has claws?” correct options are {bear, dog, ....} and distractors are {cow, seal, elephant, bat, ....}. Due to the missing edge hasA(seal, claw), seal becomes a distractor even though it is a correct answer. Our hypothesis for adding missing edges is that if there is a missing edge from word w1 to word w2 of relation r, then the cosine similarity score between w1 and w2 must be approximately similar to the cosine similarity score between other words connected to word w2 with the same relation r. For adding missing edges, we performed several simulations for the cosine similarity scores (ρ), their means (μ), and their standard deviation (σ) and obtained the following: if (ρ ≥ μ) then we assume a valid link; if (μ − σ ≤ ρ < μ) we are not sure about the quality of the link; and if (ρ < μ − σ), we characterize it as an invalid link.

### 4.3.3 MCQ Generation Method
Our hypothesis for MCQ is that it should have distractors that do not share any common properties with the correct answer. To ensure some confidence in discontinuity between an answer and distractors we leverage the idea of finding non-overlapping graph communities within words in YL-KB. We take the YL-KB graph as a directed graph, ignoring the relationship labels on the edges and use CNM [10] to find communities. For each community, we do a one-hop expansion of each node in that community and remove repeated nodes in this set of expanded and original nodes. Thus, we get new nodes that belong to other communities. We call them leading nodes, as they form a bridge between the communities. To generate MCQs, we find the community for each seed word, and its leading nodes. In this way, we can move from a seed word to a related community, if a path between a chosen leading node and a seed word exists. To generate distractors for the seed word and for a seed relation, we pick words from the related communities which are related to other words in their community using the same seed relation.

### 5. EXPERIMENTS AND EVALUATION
In this section, we present the experiments we conducted for evaluating our proposed approach.

#### 5.1 Experimental Setup
As mentioned earlier, we curated the ConceptNet to create YL-KB. It has age-appropriate themes relevant for young language learners as specified by [11] such as bird, fruit, vegetable, color, insect and animals. We then picked edges labeled with isA, hasA, atLocation, synonym, antonym, usedFor, and capableOf relationships. The edges were filtered where either the source node word or target node word was not part of the children’s book test [26] for the purpose of filtering inappropriate words.

#### 5.2 Experiment Design
From YL-KB, using the methods described in Section 4, we generated correct answers and distractors. For each seed word, we could generate questions in the range of thousands.

<table>
<thead>
<tr>
<th>Question</th>
<th>Correct Options Before Adding Missing Edges</th>
<th>Distractors Before Adding Missing Edges</th>
<th>Nodes Added to Correct Answers</th>
<th>Nodes removed from distractors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fruit hasA peel?</td>
<td>{lemon, orange, banana, apple}</td>
<td>{melon, lime, pineapple, pumpkin, pear, pomegranate, avocado, plum, ......}</td>
<td>(avocado, melon, lime)</td>
<td>{pumpkin, pomegranate, pear, pineapple}</td>
</tr>
<tr>
<td>Tools usedFor cut?</td>
<td>{knife, saw}</td>
<td>{chisel, screw, axe, hoe, ......}</td>
<td>(chisel, axe)</td>
<td></td>
</tr>
<tr>
<td>Food hasA crust?</td>
<td>{bread, pie}</td>
<td>{fruit, mushroom, snack, candy, los[...]}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insect capableOf fly?</td>
<td>{butterfly, bee}</td>
<td>{grasshopper, wasp, bumblebee, tick, worm, ......}</td>
<td>(wasp)</td>
<td>{grasshopper, bumblebee}</td>
</tr>
</tbody>
</table>

Table 1: Examples of Missing Edges Removed
In order to establish the efficacy of our approach, we conducted a question usability validation. We also conducted validations which compared our approach with a baseline. For all the validations, we had three volunteers manually annotate the questions. All volunteers were in the age group of 25 – 35 years and had a higher education degree where English was the medium of instruction.

5.2.1 Baseline:
We used vanilla ConceptNet, without applying any filtering to handle noisy or missing edges, to generate MCQs using our logic to create correct answer and distractors. We generated 600 questions using our approach (filtered KB i.e., YL-KB) and this baseline approach (without filtered KB), keeping the same number of questions per word. We asked each annotator to manually annotate all the 600 questions based on usability of the questions on a rating scale of 0 to 3, where 0 corresponds to “no problem with correct answer and distractors”, 1 corresponds to “no problem with correct answer and there is a problem with only one distractor”, 2 corresponds to “no problem with correct answer and there is a problem with two distractors”, and 3 corresponds to “either there is a problem with correct answer or all the distractors”.

5.2.2 Question Usability Validation:
The experimental setup and rating score criteria in this validation was the same as described in Baseline. This validation set had 300 questions each from baseline and our approach, i.e. 600 questions in total.

5.3 Results & Discussion
In this section, we report the results of validations we conducted. Figure 3 compares the average annotator percentage for each rating between Baseline (vanilla ConceptNet) and our approach. The difference of 21% in rating 0 and 17% in rating 3 signifies that the MCQs generated using vanilla ConceptNet require more revision than MCQs generated using our approach due to noisy and missing links. We observe an inter-annotator Fleiss Kappa agreement of 0.56 i.e. a moderate inter-annotator agreement. Although this validation was done to compare the usability of generated MCQs, however, all the annotators reported that the relatedness of distractors with the correct answer was low in Baseline compared to our approach. Based on annotation data and interviews conducted with annotators, we infer that some of the ambiguity and less than perfect annotation results arise because of each annotator’s individual perspective on word meanings. The observation reiterates why vocabulary assessment, especially for young learners, is a hard problem space, since words are not fixed units of meaning, and can be interpreted differently based on the context they occur in, or on individual perceptions.

6. CONCLUSION
In this paper we presented a system that uses a curated common sense knowledge base for young learners in combination with graph based inferencing to automatically generate MCQs for vocabulary assessments. We tested our system extensively by comparing human inter-annotator agreements on a large set of system generated MCQs, and observed moderate agreement on the MCQs. These initial results are very encouraging to conduct further investigations into how we can build such systems which can generate more complex questions, generate more personalized vocabulary assessments etc. We would also like to look at how this kind of a framework affects the generation of assessments in different modalities (image, audio, video etc.) which are so prevalent in early childhood learning curricula.

7. REFERENCES


