Generating Diagnostic Multiple Choice Comprehension Cloze Questions

Jack Mostow and Hyeju Jang
Project LISTEN (www.cs.cmu.edu/~listen)
Carnegie Mellon University
RI-NSH 4103, 5000 Forbes Avenue
Pittsburgh, PA 15213-3890, USA
mostow@cs.cmu.edu, hyejuj@cs.cmu.edu

Abstract

This paper describes and evaluates DQGen, which automatically generates multiple choice cloze questions to test a child's comprehension while reading a given text. Unlike previous methods, it generates different types of distracters designed to diagnose different types of comprehension failure, and tests comprehension not only of an individual sentence but of the context that precedes it. We evaluate the quality of the overall questions and the individual distracters, according to 8 human judges blind to the correct answers and intended distracter types. The results, errors, and judges' comments reveal limitations and suggest how to address some of them.

1 Introduction

This paper presents an automated method to check a reader's comprehension of a given text while reading it, and to diagnose comprehension failures. In contrast to testing reading comprehension skill, for which there are published tests with well-established psychometric properties (e.g., Wiederholt & Bryant, 1992; Woodcock, 1998), testing comprehension during reading of a given text requires generating a test for that specific text.

A widely used solution is to replace some of the words with blanks for the student to fill, typically by selecting from multiple candidates. Such multiple choice fill-in-the-blank questions are called cloze questions. They are trivial to score because the correct answer is simply the original text word.

Cloze questions test the ability to decide which word is consistent with the surrounding context. Thus it taps the comprehension processes that judge various types of consistency, such as syntactic, semantic, and inter-sentential.

In a nutshell, these processes successively encode sentences, integrate them into an overall representation of meaning, notice gaps and inconsistencies, and repair them (see, e.g., Kintsch, 1993, 2005; van den Broek, Everson, Virtue, Sung, & Tzeng, 2002). The reader's resulting *situation model* represents "the content or microworld that the text is about" (Graesser & Bertus, 1998).

In this paper, we introduce DQGen (Diagnostic Question Generator), a system that uses natural language processing to generate diagnostic cloze questions that check the comprehension of someone reading a given text. DQGen differs from previous methods for generating cloze questions in that it is designed to minimize disruption to the reading process, and to diagnose different types of comprehension failure.

The intended application context that motivated the development of DQGen is an automated reading tutor that listens to children read aloud and helps them build their oral reading fluency (Mostow, 2008). Periodic comprehension checks should deter children from reading as fast as they can and ignoring what the text means. When the child answers incorrectly, the wrong answers should provide clues to why they are wrong.

The rest of this article is organized as follows. Section 2 describes the generated questions. Section 3 describes how DQGen generates distracters. Section 4 reports a pilot evaluation of it. Section 5 analyzes errors. Section 6 relates DQGen to prior work. Section 7 concludes.

2 Form of Generated Cloze Questions

Generating cloze questions requires deciding:

1. Which sentences to make cloze questions?
2. Which words to delete from them?
3. How many distracters to provide for them?
4. What types of distracters?

To illustrate the results of DQGen's decisions, Figure 1 shows one of the better questions it generated:

> *Some of those cells patrol your body. They are hungry, and they eat germs! Some stop the trouble germs make. Others make antibodies. They stick to germs. That helps your body find and kill _____ .*
>
> *a) are*
> *b) intestines*
> *c) terrorists*
> *d) germs*

Figure 1. An example of a generated question

The four decisions enumerated above involve tradeoffs among preserving the flow of reading, encouraging comprehension, and assessing it accurately. As this example illustrates, DQGen inserts cloze questions as comprehension checks at the end of paragraphs, where there are natural breaks, in order to minimize disruption to the flow of reading. If the last sentence is shorter than four words or DQGen fails to find an acceptable distracter of each type, it simply leaves the last sentence unchanged rather than turn it into a bad cloze question.

DQGen deletes the last word of the sentence, in order to allow normal reading up till that point and thereby minimize disruption to the flow of reading. Deleting a word earlier in the sentence would force the reader to skip the deleted word and read ahead to answer the cloze question. Indeed, a review of of comprehension assessments (Pearson & Hamm, 2005) indicates that end-of-sentence multiple choice cloze questions are widely used: "Delete words at the end of sentences and provide a set of choices from which examinees are to pick the best answer (this tack is employed in several standardized tests, including the Stanford Diagnostic Reading Test and the Degrees of Reading Power)."

The number of distracters involves a tradeoff. On the one hand, the more distracters, the less chance of lucky guesses, and the more types of distracters possible. On the other hand, offering more distracters lengthens the disruption to the flow of reading and raises the cognitive load on the reader to remember the paragraph when reading the distracters. As a compromise, DQGen adds three distracters, for a total of four choices to present in randomized order – typical for multiple choice questions on educational tests for children.

DQGen uses three types of distracters. Each type of distracter indicates a different type of comprehension failure when chosen incorrectly by the reader as the answer. By aggregating children's performance over questions with these same three types of distracters, we hope not only to test their comprehension, but to profile the difficulties encountered by a given child or posed by a given text.

2.1 Ungrammatical distracters

The first and presumably easiest type of distracter renders the completed sentence ungrammatical. Syntactic processing is part of comprehension but not necessarily well-developed in children. Analysis of children's responses to 69,000 multiple cloze questions automatically generated, presented, and scored by the Reading Tutor (Mostow et al., 2004) found that children's performance decreased as the number of distracters with the same part of speech as the correct answer increased. However, this effect was weaker for lower-level readers, indicating less sensitivity to syntax (Hensler & Beck, 2006). Choosing an ungrammatical distracter indicates failure to detect a syntactic inconsistency. The ungrammatical distracter, e.g., *are* in Figure 1, has a different part of speech (POS) than the correct answer *germs*.

2.2 Nonsensical distracters

The second type of distracter makes the completed sentence grammatical but nonsensical. Choosing a nonsensical distracter indicates failure to detect a local semantic inconsistency with the rest of the sentence. The nonsensical distracter has the same

	Ungrammatical	Nonsensical	Plausible
Source of candidates	Other words in paragraph	List of words at grade level up to 4	Matching Google N-grams
Same as correct answer?	No	No	No (94.96%)
Related to words earlier in paragraph?	–	–	No (lowest score)
Related to words earlier in sentence?	–	–	Yes (55.77%)
Contains a space?	–No	No (100%)	–No
Frequent enough for children to know?	–Yes	–Yes	Yes (96.15%)
Passes grammar checker?	No (65.48%)	Yes (52.62%)	Yes (92.31%)*
Same POS as answer?	–No	Yes (26.67%)	–
Matches a Google N-gram?	No (95.83%)	No (91.67%)	–Yes

Table 1. Sources and constraints for each distracter type, in order tested (with % satisfied in pilot data). Constraints guaranteed to be satisfied or violated without explicit testing are marked –Yes or –No. * We added this test after the pilot evaluation because Google N-grams aren't always grammatical.

POS as the correct answer, but plugging it into the sentence forms a context not found in the Google N-grams corpus. For example, the nonsensical distracter in Figure 1 is *intestines*.

2.3 Plausible distracters

The third and hardest type of distracter makes the completed sentence meaningful in isolation but inconsistent with the preceding global context. This type of distracter is essential in testing inter-sentential processing, i.e. "understanding that reaches across sentences in a passage," because otherwise "an individual's ability to fill in cloze blanks does not depend on passage context" – a frequent criticism of cloze questions (Pearson & Hamm, 2005). A plausible distracter has the same POS as the correct answer, like a nonsensical distracter, but the sentence it forms when plugged into the blank sounds reasonable – in isolation. That is, it ends with an N-gram that occurs in the Google N-grams corpus. However, it doesn't make sense in the context of the preceding sentences, because the distracter is unrelated to the words in the preceding sentences. For example, *terrorists* in Figure 1 is a plausible distracter.

3 Generating and Filtering Distracters

DQGen uses generate-and-test to construct each type of distracter: it chooses randomly from a source of candidates and backtracks if the chosen candidate violates a constraint on that type of distracter. If none of the candidates that satisfy a constraint survive subsequent tests, DQGen drops the constraint and considers candidates that violate it. The source and constraints vary by distracter type (ungrammatical, nonsensical, plausible). Table 1 summarizes the tests and the order they are applied. Sections 3.1-3.3 discuss them in further detail.

3.1 Lexical constraints on distracters

Three constraints apply at the word level.

No spaces: We constrain all three types of distracters to be words rather than phrases. This constraint is guaranteed for paragraph words and Google N-grams, DQGen's respective sources of ungrammatical and plausible distracters. However, our source of nonsensical distracters is a table (Biemiller, 2009) that specifies the grade level not only of words but also of some phrases, such as *barbeque sauce*, which DQGen therefore filters out. Table 2 shows an excerpt from the table used.

Word	Meaning	Level	…
barbecue sauce	flavored sauce for meat	2	
intestines	guts	4	
intimate	close, friendly	10	
intimate	a close friend	10	

Table 2. Excerpt from Biemiller's (2009) table

Distinct: DQGen explicitly excludes the correct answer as a distracter. Other constraints on different types of distracters are mutually exclusive with

each other. Consequently, no answer choice can appear twice.

Familiar: Distracters must be familiar to children. DQGen satisfies this constraint for ungrammatical and nonsensical distracters by choosing them from the paragraph and a grade-leveled word list (Biemiller, 2009), respectively. These sources suffice to provide candidates, but they are not comprehensive enough to test candidates from another source, such as the Google N-grams used to generate plausible distracters. To exclude words likely to be unfamiliar to children, DQGen filters out candidates whose unigram frequency falls below 5,000,000. We tuned this threshold by informal trial and error; higher thresholds proved too stringent to allow any distracters from the limited source of candidate plausible distracters.

3.2 Constraints on completed sentences

Three constraints pertain to making completed sentences sensible or not.

Grammatical: As Table 1 shows, all three types of distracters involve grammaticality constraints. Ungrammatical distracters must make the completed sentence ungrammatical, e.g., *That helps your body find and kill are*. In contrast, nonsensical and plausible distracters must make the completed sentence grammatical, e.g., *That helps your body find and kill terrorists*.

To check grammaticality of a completed sentence, we use the Link Grammar Parser (Sleator & Temperley, 1993), a syntactic dependency parser, as a grammar checker. As a grammar checker, the Link Grammar Parser usually accepts grammatical sentences and rejects ungrammatical ones, perhaps because sentences in children's text tend to be short. However, it sometimes fails to accept a grammatical sentence, as the last row of Table 3 illustrates.

sentence	grammatically	parser
The germs hide in food or people	correct	accepted
The germs hide in food or world	incorrect	rejected
So keep dirty hands away from cuts and your face.	correct	rejected

Table 3. Examples of grammar checking by parser

Part of speech: More than one POS may make a distracter grammatical. DQGen uses the Stanford POS Tagger (Toutanova, Klein, Manning, & Singer, 2003) to tag the correct answer and a candidate nonsensical distracter when used to complete the sentence, and requires them to have the same POS. This test is superfluous for ungrammatical distracters and unnecessary for plausible distracters.

Google N-gram: As a heuristic test of whether a completed sentence is plausible, we check whether its ending occurs in the Google N-grams corpus (Brants & Franz, 2006), which means that it appears at least 40 times on the Web. For ungrammatical and nonsensical distracters, the last 4 words of the completed sentence must not occur in this corpus. For plausible distracters, the last 4 words followed by "." must occur. To enforce this constraint, DQGen's source of candidate plausible distracters consists of Google 5-grams of the form *W X Y* __ . Here *W*, *X*, and *Y* are the words preceding the correct answer in the original sentence, e.g., *find and kill*. If there are fewer than 5 such 5-grams, DQGen allows 4-grams of the form *X Y* __ ., e.g. *and kill* __.

3.3 Relevance to context

Two constraints on distracters concern context.

Irrelevant to words earlier in paragraph: A plausible distracter should not be *too* plausible, so DQGen tries to ensure that it is unrelated to the earlier sentences and hence unlikely to make sense in context. We measure the relatedness of a distracter to words in the earlier sentences by how often it co-occurs with them *when used as in the last sentence*. DQGen therefore first pairs the candidate distracter, e.g. *terrorists*, with the last content word preceding the blank, e.g., *kill* in *That helps your body find and kill* ____. It then estimates the probability of these two words (*kill* and *terrorists*) co-occurring with the words in the earlier sentences of the paragraph, using a Naïve Bayes formula to score their relevance to that context:

$$\Pr(c,k \mid \vec{w}) \propto \Pr(c,k)\prod_{i=1}^{n}\Pr(w_i \mid c,k)$$

The formula omits $\Pr(\vec{w})$ because it's the same for all candidate plausible distracters for a given question. Here c is a candidate distracter (e.g., *terrorists*), k is the last content word before the blank (e.g., *kill*), $\vec{w}$ is a vector of the n content words earlier in the paragraph, and w_i is the i^{th} such word.

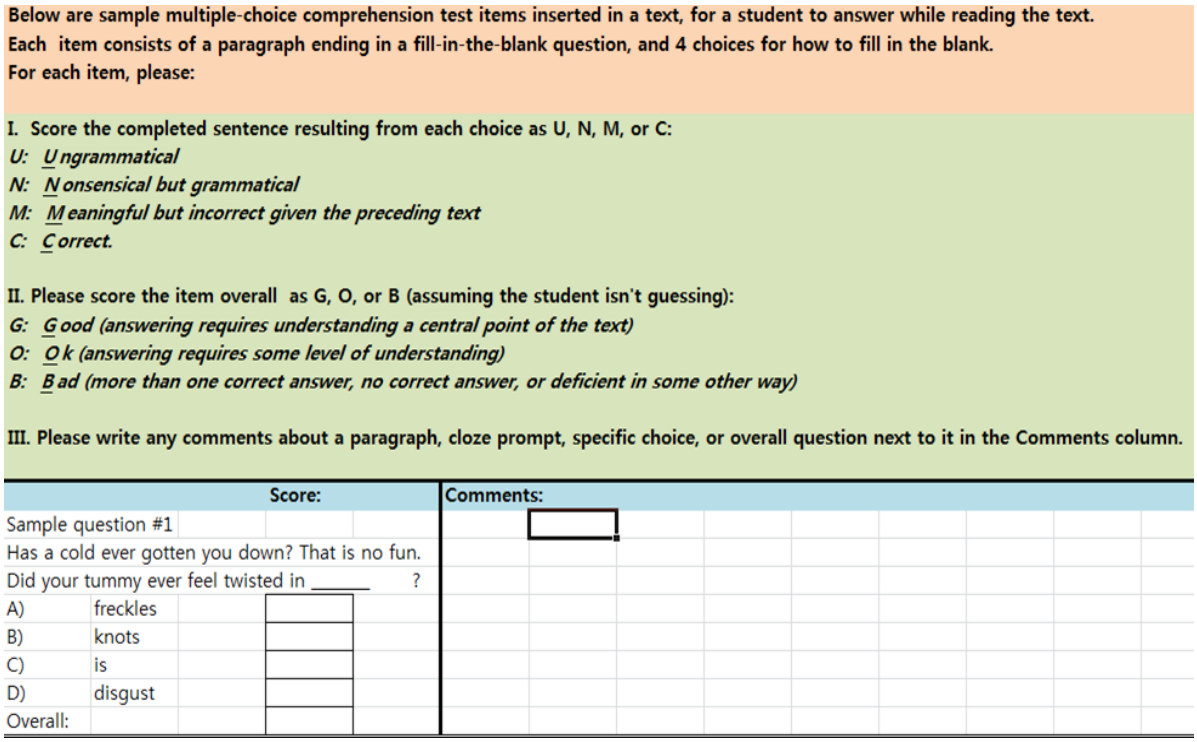

Figure 2. Prompt for the pilot user test

DQGen scores $\Pr(w_i \mid c,k)$ based on how often word w_i co-occurs with words c and k in the same 30-word window in the British National Corpus (BNC).

The purpose of a plausible distracter is to detect failures of intersentential comprehension processes that monitor global consistency. As a heuristic to violate global consistency, DQGen picks distracters with the *lowest* relevance scores.

Relevant to words earlier in sentence: A plausible distracter should be relevant to the words earlier in the sentence. To score local relevance, DQGen uses a Naïve Bayes formula similar to its formula for global relevance:

$$\Pr(c \mid \vec{w}) \propto \Pr(c)\prod_{i=1}^{n}\Pr(w_i \mid c)$$

Here, c is a candidate distracter, $\vec{w}$ is a vector of the n content words earlier in the sentence, and w_i is the i^{th} such word. DQGen estimates $\Pr(w_i \mid c)$ in the same way as before, but omits k because n is so much smaller for the sentence than for the paragraph context preceding it. DQGen averages these local coherence scores over the candidates, and allows only candidates whose local coherence scores are above the mean.

4 Pilot Study

How good are the generated questions? To evaluate DQGen, we asked human judges to score them. Section 4.1 explains how we evaluated questions, Section 4.2 reports inter-rater reliability, and Section 4.3 presents results.

4.1 Methodology

For the evaluation, we used DQGen to insert sample questions in an informational text for children, *The Germs*, which explains the concept of germs and their danger. Of the 18 paragraphs in this text, we rejected one because it was only two sentences long, and DQGen rejected another because the last sentence failed the grammar checker. For each of the other 16 paragraphs, DQGen generated a cloze question with ungrammatical and nonsensical distracters, but it found plausible distracters for only 13 of the questions, which we evaluated as follows.

We recruited eight human judges, members of our research team but unfamiliar with DQGen. We asked them to evaluate each question at two levels, using the form illustrated in Figure 2.

At the high level, we evaluated the overall quality of each question by asking judges to rate it as

Good, *OK*, or *Bad*. We report the percentage of generated questions rated by human judges as acceptable, defined as *Good* or *OK*. We used a 3-point scale rather than a finer-grained scale both to get higher inter-rater reliability, and because we were interested more in how many of the questions were acceptable than in precise ratings of quality.

At the low level, we evaluated how often DQGen generated the intended type of distracter. We asked the judges to categorize each of the multiple choices (correct answer plus 3 distracters) as *Ungrammatical*, *Nonsensical but grammatical*, *Meaningful but incorrect given the preceding text*, or *Correct*. To avoid biasing their responses, we did not tell them that each question was supposed to have one choice in each category.

To elicit additional feedback, the form invited judges to comment on the questions and distracters.

4.2 Inter-rater reliability

It is important to measure inter-rater reliability among human judges, especially on experimenter-designed measures such as the form we used.

The overall quality ratings involved ranked data from more than two judges, so to measure their inter-rater reliability we used Kendall's Coefficient of Concordance (Kendall & Smith, 1939). KCC for overall quality was .40 on a scale from 0 to 1. This low value reflects the considerable variation between the judges, whose average ratings of overall quality ranged from 1.3 to 2.6.

Categorization of each answer choice involved unranked data from more than two judges, so we used Fleiss' Kappa (Shrout & Fleiss, 1979) to measure its inter-rater reliability. Kappa was .58; a value of .4-.6 is considered moderate, .6-.8 substantial, and .8-1 outstanding (Landis & Koch, 1977). Figure 3 shows the Kappa values for each label by the judges.

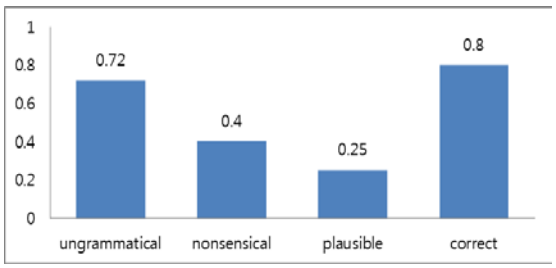

Figure 3. Fleiss's Kappa for inter-rater reliability of each type of choice

The low values of inter-rater reliability measures revealed the raters' lack of consensus, presumably due to differing interpretations of the instructions. For instance, one judge commented that instruction for rating the overall quality did not indicate whether a good question requires reading the preceding text. Another issue was missing and multiple categorical responses.

Evidently we need to specify our rating criteria more clearly, both for overall quality and for individual components, especially nonsensical and plausible distracters. A worked-out example might help judges understand each type better, but must avoid phrasing biased toward how DQGen works.

4.3 Results

We computed average ratings of overall quality and agreement with the intended category of each answer choice.

We averaged all the ratings of overall quality after converting *Bad*, *OK*, and *Good* ratings into 1, 2, and 3, respectively. Overall quality ratings averaged 2.04, which corresponds to *OK*. For agreement of judges with the intended category of each answer choice, Cohen's Kappa was .60. Note that in contrast to Section 4.2, where we used Kappa to measure inter-rater reliability, i.e., how well the judges agreed with each other on overall question quality, here we use Kappa to measure distracter quality, i.e., how well the judges agreed with DQGen on the intended type of answer choices.

Individual judges ranged from 63% to 79% agreement with the intended answer (Cohen's Kappa .51 to .72). As Figure 4 shows, agreement was stronger for correct answers and ungrammatical distracters than for nonsensical and plausible distracters. On average, judges rated 94% of the correct answers as correct and agreed with DQGen's intended distracter type for 91% of the ungrammatical distracters, 63% of the nonsensical distracters, and only 32% of the plausible distracters. Apparently correct answers are obviously right and ungrammatical answers are obviously wrong, but nonsensical and plausible distracters are harder to classify.

5 Analysis of errors

We now discuss issues revealed by errors and judges' comments, and how to address them.

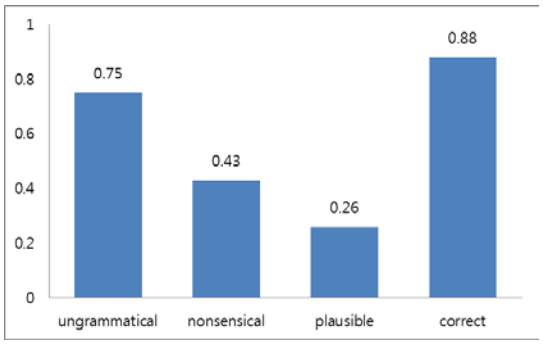

Figure 4. Cohen's Kappa for agreement with the intended type of each choice

5.1 Dependence on preceding text

The judges' most frequent comment about the quality of a question was that answering it did not require reading the preceding text. The judges rated only 32% of the intended plausible distracters as plausible. Evidently we need to identify further constraints on plausible distracters. We may also need to identify constraints on sentences where plausible distracters exist for the correct answer.

5.2 Idioms

Answer choices, whether correct answers or distracters, are problematic when they form idioms such as *twisted in knots* or *make do.* For instance, one pilot cloze question ended with *twisted in* ____, where the correct answer was *knots*. Another question ended with *get your body to make* ____, with *do* as a supposedly ungrammatical distracter.

Idioms pose multiple problems, although we found only two cases in our small pilot study. First, we want to test comprehension of the paragraph, not just knowledge of specific idioms. Second, the word that completes an idiom can be far likelier than any other choice, making it too easy to guess based solely on local context, whether correct or not. Third, because idioms have non-componential semantics, the missing word is liable to be semantically unrelated to other sentence words, causing DQGen to badly underestimate its local relevance.

Detecting idioms automatically is a research problem in its own right (Li, Roth, & Sporleder, 2010; Li & Sporleder, 2009). We might be able to recognize idioms by using the fact that its N-gram frequency is much higher than expected based on the frequency of its individual words. A simpler approach is to consult a dictionary of common phrases. Either approach would require extension to handle parameterized idioms such as *a chip on [someone's] shoulder*, or non-contiguous forms such as *Actions do in fact speak louder than words.*

5.3 Lexical issues for distracters

The pilot study exposed a number of issues affecting the suitability of words as distracters.

Same-root words

DQGen ensures that answer choices are distinct. However, one question included two forms of the same word as choices, namely *throats* as the correct answer and *throat* as a plausible distracter. We need to ensure that answer choices are not only distinct but dissimilar, unless we want questions that focus on minor differences between them.

Common verbs and modal verbs

One judge commented that we might want to avoid common verbs as distracters, such as any form of *be*, *do*, *have*, and *get*, and modal verbs, such as *can*, *cannot*, and *will*, lest children notice that they are seldom the correct answer, and therefore eliminate them without considering them. Accordingly, we plan to filter out common verbs and modal verbs.

Word difficulty

The same judge considered some words too difficult for children, such as *gauge* and *roast*. Actually, Biemiller (2009) rates noun senses of these words at grade 2, but the verb sense of *gauge* as *estimate* at grade 10. These examples illustrate a limitation of DQGen's methods to pick familiar words as distracters. It picks ungrammatical distracters from the words in the paragraph, nonsensical distracters from Biemiller's word list, and plausible distracters from Google N-grams, filtered by unigram frequency to avoid rare words. In all three cases, DQGen constrains words rather than word senses.

A more sophisticated approach would determine a distracter's word sense, or at least POS, when used to complete the sentence, and rate the familiarity of its specific sense or POS. Tagging the distracter POS is easier than determining its word sense(s) when inserted in the sentence. Rating the familiarity of different word senses would require either a grade-leveled list of them like Biemiller's (2009), or a resource with information about the frequency of different word senses or POS.

6 Relation to Prior Work

How does this research relate to previous work? There has been considerable research on automatic generation of multiple choice cloze questions to test vocabulary, grammar, and comprehension. Although these types of questions differ in purpose, they have much in common when it comes to generating them automatically.

6.1 Vocabulary and grammar cloze questions

A multiple choice cloze question to test vocabulary and grammar is constructed from a sentence selected from a corpus by deleting part of it (typically the target vocabulary word) and selecting distracters for it.

Selecting distracters with the same POS and approximate frequency as the answer word is a common strategy (Brown, Frishkoff, & Eskenazi, 2005; Coniam, 1997; Liu, Wang, & Gao, 2005).

Besides matching the correct answer's POS and frequency, Liu et al. (2005) added a culture-dependent strategy for generating distracters: choose English words with semantically similar translations in the learner's native language to the translation of the answer word.

Correia et al. (2010) generated vocabulary questions for Portuguese with three types of distracters. One type of distracter had the same POS and word level as the target word, based on its unigram frequency in Portuguese textbooks used in different grades. A second type had the lowest Levenshtein distance to the target out of all words with its POS. A third type was misspellings of the target word using a table of common spelling mistakes. Aldabe et al. (2007) also included students' common mistakes as candidate distracters.

Some work also used semantic similarity between a distracter and the answer word to choose distracters. Pino et al. (2008) selected distracters that made the completed sentence grammatical and tended to co-occur with the words in the sentence, but were semantically distant from the target word as measured by WordNet. In constrast, Smith et al. (2008) looked for distracters semantically similar to the answer word based on distributional similarity. In addition, Sumita et al. (2005) used a thesaurus for the same purpose, and then consulted the web to filter out plausible distracters.

Aldabe et al. (2009) considered context in a question sentence when choosing distracters. They used an n-gram language model to predict the probability of occurrence of a distracter with its preceding words.

Gates et al. (2011) generated phrase-type distracters, unlike other work. They generated questions from a dictionary definition of the target vocabulary word. Rather than delete the target word, they parsed the definition, deleted a phrase from it, and chose distracters with the same syntactic phrase type from definitions of other words, filtered to exclude synonyms of the target word.

6.2 Comprehension cloze questions

In contrast to vocabulary and grammar questions constructed from isolated sentences, DQGen's comprehension questions are for (and inserted into) connected text.

The most closely related work was by Mostow et al. (2004). Their Reading Tutor dynamically generated multiple choice cloze questions to test children's comprehension of randomly chosen sentences while reading a story. It randomly chose an approximate level of difficulty ('sight', 'easy', 'hard', and 'defined') for which word to delete from the sentence, and which words to choose randomly from the same story as distracters.

Goto et al. (2010) also generated questions from texts. They used a training corpus of existing cloze questions to learn how to select sentences to turn into cloze questions, words to delete, and types of distracters distinguished by their relation to the answer word: inflectional (e.g., *ask* → *asked*); derivational (e.g., *work* → *worker*); orthographic (e.g., *circulation* → *circumcision*); and semantic (e.g., synonyms and antonyms).

Aldabe et al. (2010) generated questions for learners' assessment in the science domain. To generate distracters, they measured semantic similarity by using Latent Semantic Analysis (LSA) and additional information such as semantic relationships between words. Experts discarded distracters that could form a correct answer.

DQGen differs from prior work on generating cloze questions for vocabulary and comprehension in two key respects. First, each question it generates has multiple types of distracters designed to detect different types of comprehension failure. Second, to generate plausible distracters it considers their relation not only to the clozed sentence but to the entire paragraph that contains it.

7 Conclusion

We conclude by summarizing contributions, limitations, and future work.

7.1 Contributions

This paper describes a method for generating multiple choice cloze questions to test students' comprehension while reading. Unlike previous methods, some of which also generate multiple types of distracters, DQGen's distracter types are diagnostic. It generates ungrammatical, nonsensical, and plausible distracters in order to detect failures of syntactic, semantic, and intersentential processing, respectively. Unlike prior methods, which test comprehension only of individual sentences, DQGen's plausible distracters take their preceding context into account.

We observed that candidate plausible distracters with high relevance scores tend to be surprisingly sensible answers – even though the formula doesn't "know" the correct answer or even the ungrammatical and nonsensical distracters. That is, grammaticality, N-grams, and a simple relevance measure often suffice to produce intelligent answers to a cloze question despite their shallow representation of the meaning of the paragraph – that is, without really understanding it. This finding is surprising insofar as one would expect good performance on such questions to require a deep representation such as the situation model constructed by human readers.

7.2 Limitations

Besides describing DQGen's design and implementation, we report on an evaluation of 13 generated questions by eight human judges blind to correct answer and intended distracter type. On average they rated overall question quality OK, but with a wide range from the least to most favorable judge. They agreed well with DGQen in classifying answers as ungrammatical or correct, but not as nonsensical or plausible. They criticized many questions as answerable without reading the text.

7.3 Future work

Our analysis of errors and judges' comments revealed several limitations and suggested ways to address some of them. In addition to identifying further constraints on plausible distracters, we need to identify constraints on good sentences to turn into end-of-paragraph cloze questions, beyond just the ability to generate a distracter of each type. One criterion is reliability: how well does performance on a question correlate with performance on other questions about the same text? Another criterion is informativeness: what do wrong answers reveal about comprehension?

Besides improving DQGen, we need to test it on more stories (both narrative fiction and informational text) and readers (especially children, our target population) to expose additional problems and avoid overfitting their solutions.

One possible use of DQGen is machine-assisted generation of comprehension questions, or more precisely, human-assisted machine generation, for example with the human vetting or selecting among candidate questions generated automatically, thereby reducing the amount of human effort currently required to compose comprehension questions, and producing them more systematically.

Success in getting DQGen to produce cloze questions on a large scale would have useful applications. Periodic comprehension checks should deter children from reading as fast as they can and ignoring what the text means. Diagnostic feedback based on incorrect answers should shed light on the nature of their comprehension failures and may be valuable as feedback to teachers or as guidance to the reading tutor.

Another use for large numbers of automatically generated cloze questions is to develop methods to monitor reading comprehension unobtrusively. Student responses to cloze questions could provide automated labels for data collected while they read the preceding text. Such data could include oral reading (Zhang, Mostow, & Beck, 2007) or even EEG (Mostow, Chang, & Nelson, 2011). Models trained and tested on the labeled data could estimate reading comprehension based on unlabeled data – that is, without interrupting to ask questions.

Acknowledgements

The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A080157. The opinions expressed are those of the authors and do not necessarily represent the views of the Institute or the U.S. Department of Education. We thank our colleagues who judged the generated questions, and the reviewers for their helpful comments.

References

Aldabe, I., & Maritxalar, M. (2010). *Automatic Distractor Generation for Domain Specific Texts Advances in Natural Language Processing.* Paper presented at the The 7th International Conference on NLP, Reykjavk, Iceland.

Aldabe, I., Maritxalar, M., & Martinez, E. (2007). *Evaluating and Improving Distractor-Generating Heuristics.* Paper presented at the The Workshop on NLP for Educational Resources. In conjuction with RANLP07.

Aldabe, I., Maritxalar, M., & Mitkov, R. (2009). *A Study on the Automatic Selection of Candidate Sentences and Distractors.* Paper presented at the Proceedings of the 14th International Conference on Artificial Intelligence in Education (AIED2009), Brighton, UK.

Biemiller, A. (2009). *Words Worth Teaching: Closing the Vocabulary Gap*. Columbus, OH: SRA/McGraw-Hill.

Brants, T., & Franz, A. (2006). Web IT 5-gram Version 1. Philadelpha: Linguistic Data Consortium.

Brown, J. C., Frishkoff, G. A., & Eskenazi, M. (2005). *Automatic Question Generation for Vocabulary Assessment*. Paper presented at the Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), Vancouver.

Coniam, D. (1997). A preliminary inquiry into using corpus word frequency data in the automatic generation of English language cloze tests. *CALICO Journal, 14*(2-4), 15-33.

Correia, R., Baptista, J., Mamede, N., Trancoso, I., & Eskenazi, M. (2010, September 22-24). *Automatic generation of cloze question distractors.* Paper presented at the Proceedings of the Interspeech 2010 Satellite Workshop on Second Language Studies: Acquisition, Learning, Education and Technology, Waseda University, Tokyo, Japan.

Gates, D., Aist, G., Mostow, J., Mckeown, M., & Bey, J. (2011, November 4-6). *How to Generate Cloze Questions from Definitions: a Syntactic Approach.* Paper presented at the Proceedings of the AAAI Symposium on Question Generation, Arlington, VA.

Goto, T., Kojiri, T., Watanabe, T., Iwata, T., & Yamada, T. (2010). Automatic Generation System of Multiple-Choice Cloze Questions and its Evaluation. *Knowledge Management & E-Learning: An International Journal (KM&EL), 2*(3).

Graesser, A. C., & Bertus, E. L. (1998). The Construction of Causal Inferences While Reading Expository Texts on Science and Technology. *Scientific Studies of Reading, 2*(3), 247-269.

Hensler, B. S., & Beck, J. (2006, June 26-30). *Better student assessing by finding difficulty factors in a fully automated comprehension measure [Best Paper nominee]*. Paper presented at the Proceedings of the 8th International Conference on Intelligent Tutoring Systems, Jhongli, Taiwan.

Kendall, M. G., & Smith, B. B. (1939). The Problem of m Rankings. *The Annals of Mathematical Statistics, 10*(3), 275-287.

Kintsch, W. (1993). Information Accretion and Reduction in Text Processing: Inferences. *Discourse Processes, 16*(1-2), 193-202.

Kintsch, W. (2005). An Overview of Top-Down and Bottom-Up Effects in Comprehension: The CI Perspective. *Discourse Processes A Multidisciplinary Journal, 39*(2&3), 125-128.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics, 33*(1), 159-174.

Li, L., Roth, B., & Sporleder, C. (2010). *Topic models for word sense disambiguation and token-based idiom detection*. Paper presented at the Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Uppsala, Sweden.

Li, L., & Sporleder, C. (2009). *Classifier combination for contextual idiom detection without labelled data*. Paper presented at the Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore.

Liu, C.-L., Wang, C.-H., & Gao, Z.-M. (2005). Using Lexical Constraints to Enhance the Quality of Computer-Generated Multiple-Choice Cloze Items. *Computational Linguistics and Chinese Language Processing, 10*(3), 303-328.

Liu, C.-L., Wang, C.-H., Gao, Z.-M., & Huang, S.-M. (2005). *Applications of lexical information for algorithmically composing multiple-choice cloze items*. Paper presented at the Proceedings of the second workshop on Building Educational Applications Using NLP, Ann Arbor, Michigan.

Mostow, J. (2008). Experience from a Reading Tutor that listens: Evaluation purposes, excuses, and methods. In C. K. Kinzer & L. Verhoeven (Eds.), *Interactive literacy education: facilitating literacy environments through technology* (pp. 117-148). New York: Lawrence Erlbaum Associates, Taylor & Francis Group.

Mostow, J., Beck, J. E., Bey, J., Cuneo, A., Sison, J., Tobin, B., & Valeri, J. (2004). Using automated questions to assess reading comprehension, vocabulary, and effects of tutorial interventions. *Technology, Instruction, Cognition and Learning, 2*(1-2), 97-134.

Mostow, J., Chang, K.-m., & Nelson, J. (2011, June 28 - July 2). *Toward Exploiting EEG Input in a Reading Tutor [Best Paper Nominee]*. Paper presented at the Proceedings of the 15th International Conference on Artificial Intelligence in Education, Auckland, NZ.

Pearson, P. D., & Hamm, D. N. (2005). The history of reading comprehension assessment. *S. G. Paris & S. A. Stahl (Eds.), Children's reading comprehension and assessment*, 13-69.

Pino, J., Heilman, M., & Eskenazi, M. (2008). *A selection strategy to improve cloze question quality*. Paper presented at the Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains. 9th International Conference on Intelligent Tutoring Systems, Montreal, Canada.

Shrout, P. E., & Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, *86*(2), 420-428.

Sleator, D. D. K., & Temperley, D. (1993, August 10-13). *Parsing English with a link grammar*. Paper presented at the Third International Workshop on Parsing Technologies, Tilburg, NL, and Durbuy, Belgium.

Smith, S., Sommers, S., & Kilgarriff, A. (2008). *Learning words right with the Sketch Engine and WebBootCat: Automatic cloze generation from corpora and the web*. Paper presented at the Proceedings of the 25th International Conference of English Teaching and Learning & 2008 International Conference on English Instruction and Assessment, Lisbon, Portugal.

Sumita, E., Sugaya, F., & Yamamoto, S. (2005). *Measuring non-native speakers' proficiency of English by using a test with automatically-generated fill-in-the-blank questions*. Paper presented at the Proceedings of the second workshop on Building Educational Applications Using NLP, Ann Arbor, Michigan.

Toutanova, K., Klein, D., Manning, C., & Singer, Y. (2003). *Feature-rich part-ofspeech tagging with a cyclic dependency network*. Paper presented at the HLT-NAACL, Edmonton, Canada.

van den Broek, P., Everson, M., Virtue, S., Sung, Y., & Tzeng, Y. (2002). Comprehension and memory of science texts: Inferential processes and the construction of a mental representation. In J. L. J. Otero, & A. C. Graesser (Ed.), *The psychology of science text comprehension*. Mahwah, NJ: Erlbaum.

Wiederholt, J. L., & Bryant, B. R. (1992). *Gray Oral Reading Tests* (3rd ed.). Austin, TX: Pro-Ed.

Woodcock, R. W. (1998). *Woodcock Reading Mastery Tests - Revised (WRMT-R/NU)*. Circle Pines, Minnesota: American Guidance Service.

Zhang, X., Mostow, J., & Beck, J. E. (2007, July 9-13). *Can a computer listen for fluctuations in reading comprehension?* Paper presented at the Proceedings of the 13th International Conference on Artificial Intelligence in Education, Marina del Rey, CA.