Using Wikipedia and Conceptual Graph Structures to Generate Questions for Academic Writing Support

Ming Liu, Rafael A. Calvo, *Senior Member*, *IEEE*, Anindito Aditomo, and Luiz Augusto Pizzato

Abstract—In this paper, we present a novel approach for semiautomatic question generation to support academic writing. Our system first extracts key phrases from students' literature review papers. Each key phrase is matched with a Wikipedia article and classified into one of five abstract concept categories: Research Field, Technology, System, Term, and Other. Using the content of the matched Wikipedia article, the system then constructs a conceptual graph structure representation for each key phrase and the questions are then generated based the structure. To evaluate the quality of the computer generated questions, we conducted a version of the Bystander Turing test, which involved 20 research students who had written literature reviews for an IT methods course. The pedagogical values of generated questions were evaluated using a semiautomated process. The results indicate that the students had difficulty distinguishing between computer-generated and supervisor-generated questions. Computer-generated questions were also rated as being as pedagogically useful as supervisor-generated questions, and more useful than generic questions. The findings also suggest that the computer-generated questions were more useful for the first-year students than for second or third-year students.

Index Terms—Automatic question generation, writing support, natural language processing

1 Introduction

Reviewing the literature to summarize and build upon current knowledge about a topic is a key part of academic writing [1]. According to Steward [2] a good review should present a critical discussion of relevant materials, with the goal of developing new ideas from available evidence and knowledge. Unfortunately, determining the relevance of materials, engaging in critical discussion, and synthesizing evidence are precisely what students often find difficult in writing literature reviews [3]. This paper discusses the provision of reflective questions as a way to help students overcome some of these problems.

Generic trigger questions have been widely used for literature review support [4]. These are questions such as "*Have you critically analyzed the literature you use?*" and "*Have you discussed how your project will contribute to that discipline or field?*" While generic questions may be useful, students may benefit more from questions which are specific to the content of their document. However, creating such specific questions is typically difficult and time consuming. The goal of our research is to develop a fully automated method to generate specific questions to support academic writing.

- *M. Liu and R.A. Calvo are with the School of Electrical and Information Engineering, The University of Sydney, NSW 2006, Australia. E-mail: {Ming.Liu, Rafael.Calvo}@sydney.edu.au.*
- *A. Aditomo is with the University of Sydney, Room 237, A35 Education Building, NSW 2006, Australia. E-mail: aadi4954@uni.sydney.edu.au.*
- *L.A. Pizzato is with the University of Sydney, Room 328, J12-School of Information Technologies Building, NSW 2006, Australia. E-mail: luiz.pizzato@sydney.edu.au.*

Manuscript received 5 Sept. 2011; revised 28 Nov. 2011; accepted 27 Feb. 2012; published online 5 Mar. 2012.
For information on obtaining reprints of this article, please send e-mail to: lt@computer.org, and reference IEEECS Log Number TLT-2011-09-0095.
Digital Object Identifier no. 10.1109/TLT.2012.5.

Automatic Question Generation (AQG) is a challenging task which involves natural language understanding and generation [5]. Three major aspects of AQG have been addressed in the literature: selection of the target content (what to ask about), selection of the question types (e.g., Who, Why, Yes/No), and construction of the actual questions. An increasing number of automatic question generation techniques have been explored [6], [7], [8], [9], [10]. Most of this research has focused on generating questions for supporting reading comprehension or vocabulary assessment using factual or nonfactual questions. These questions can be asked to any student in the cohort and assessed by anyone qualified in the topic. In contrast, our work focuses on generating trigger questions to support academic writing, In this context the questions are based on the individual students' work and therefore can only be assessed by them.

In a previous study [11], we described and evaluated a QG approach that focuses on citation sentences in a literature review. In that approach, we classified citation sentences into these categories: Opinion, Aim, Result, Method, System, and Other. For example, if a student describes an opinion found in a citation: "*Cannon (1927) challenged this view mentioning that physiological changes were not sufficient to discriminate emotions*," the system would generate trigger questions about, for instance, the evidence regarding the opinion: *Why did Cannon challenge this view mentioning that physiological changes were not sufficient to discriminate emotions? (What evidence is provided by Cannon to prove the opinion?) Does any other scholar agree or disagree with Cannon?*

Our evaluation showed that the automatically generated questions outperformed generic questions; furthermore, they were judged as being equally useful compared to questions generated by human supervisors. In another

study [12], we analyzed 125 trigger questions generated by 25 human supervisors for supporting their research students' literature review. We identified important categories of concepts which the questions were about. These categories included *Research Field*, *System*, *Technical Term*, *Technologies*, *Opinion*, *Result and Application*. We also identified three frequent question types: *Concept*, *Verification*, *and Judgment*, based on a question taxonomy proposed by Grasser and Person [13].

The present study builds upon our previous studies. In this study, we propose a novel approach to address three key challenges of automatic trigger question generation for supporting writing. The first challenge concerns the identification of key/central concepts from the potentially many concepts that are contained in an academic paper. The second is related to the system's lack of knowledge about the domain discussed in an academic paper. And the third is how to evaluate whether the questions generated by the system are considered useful by authors/students.

To address the first challenge of identifying key concepts, the system uses an unsupervised extraction algorithm to extract key phrases from an academic paper. The system then classifies each key phrase based on a Wikipedia article matched with the key phrase by using a rule-based approach. The key phrases can belong to one of the following five concepts adapted from a conceptual taxonomy proposed by Lehnert et al. [14].

1. Research Field: The key phrase is about a research field. For example, "***Social sciences*** *are the fields of academic scholarship that study society.*"
2. Technology: the key phrase is related to a technology/method/model/algorithm/protocol, e.g., "***SOAP***, *originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation of web services in computer networks.*"
3. System: the key phrase refers to a software system or hardware device. For example, "*An* ***image retrieval system*** *is a computer system for browsing, searching and retrieving images from a large database of digital images.*"
4. Term: the key phrase describes a technical term. For example, "*The term* ***cognitive load*** *is used in cognitive psychology to illustrate the load related to the executive control of WM.*"
5. Other.

To address the second challenge, Wikipedia was used as a domain knowledge base. Knowledge from a single article is used to build conceptual graphs used to generate questions. Table 1 shows some examples of computer-generated questions from students' literature review papers. For example, the question "One limitation of Principal Component Analysis is that the results of PCA depend on the scaling of the variables; how do you address these issues in your project?" is used to encourage the student to think about the limitation of this technology and how it is related to his/her own project.

The third challenge of evaluating the quality of generated questions is addressed by conducting a Bystander Turing Test [15], where participants judge whether a question was generated by a computer system or by human tutors. The test also asks participants to rate the quality of the questions. In this study, students writing a literature review were asked to evaluate the quality of questions generated by a human tutor, automatically generated questions, and also generic questions.

TABLE 1
An Example of Question Generated by the System

Key phrase	Question
Social Science is *a Research Field*	*Social sciences are the fields of academic scholarship that study society. What impact would the proposed project have on the field of Social sciences?*
LSI is a *Technology*	*Early challenges to LSI focused on scalability and performance. How do you address these issues in your project?*
EEG is *A Technology*	*Some advantages of EEG over FMRI include Hardware costs are significantly lower. Have you considered this strength in relations to your project?*
Cognitive Load is *a Term*	*The term cognitive load is used in cognitive psychology to illustrate the load related to the executive control of WM. How do you see cognitive load being applied in your project?*

The remainder of the paper is organized as follows: Section 2 provides a review of the relevant literature. Section 3 provides details of the process of generating questions and the architecture and components of the system evaluated. Section 4 describes the evaluation and Section 5 the results and discussion. Section 6 concludes and suggests lines of future work.

2 Related Research

2.1 Writing Support Systems

Haswell [16] reviewed systems for automated feedback tracing back to the 1950s. These systems have focused on assessment of end products, and less on providing formative feedback [17], [18]. The Writer Workshop [19] and Editor [20] both focus on grammar and style. Sourcer's Apprentice Intelligent Feedback system (SAIF) [21] is a computer assisted essay writing tool used to detect plagiarism, uncited quotations, lack of citations, and limited content integration problems. It uses a rule-based approach and Latent Semantic Analysis (LSA), a technique used to measure the semantic similarity of texts [22]. SaK, a writing tutoring system [23] developed at the University of Memphis, assesses student compositions. It uses multiple animated characters to provide different aspects of feedback, such as coherence, purpose, topic, and overall quality. Like SaK, a number of automated essay assessment tools or scoring systems [23], [24], [25], [26] has been built based LSA. For example, Apex [26] uses LSA to assess student essays on topic coverage, discourse structure, and coherence.

The Glosser system [27], [28] aims to support reflection in writing through trigger questions, such as the ones discussed here. It uses text mining algorithms to help learners think about issues such as coherence, topics, and concept visualization. However, Glosser only provides generic trigger questions.

2.2 Automatic Question Generation Systems

One of the first automatic QG systems proposed for supporting novices to learn English was AUTOQUEST [29]. This approach is based on simple pattern matching rules to transform the declarative sentence into a question. For example, the pattern S1 (cause) + so that (conjunction) + S2 (effect) can be used to generate why question. For example, sentence: Jamie had an afternoon nap **so that** he wouldn't fall asleep at the concert later. Question: Why did Jamie have an afternoon nap? Other systems that support reading and language learning include Kunichika et al. [7] who proposed a question generation method based on both syntactic and semantic information (Space, Time, and Agent) so that it can generate more question types (Where, When, and Who). More recently, Mostow and Chen [9] proposed an approach to generate deep questions based on a situation model. It can generate what, how and why questions. For example, what did <character> <verb>? why/how did <character> <verb> <complement>?

Several approaches have been proposed for automatic multiple-choice QG [6], [30] from reading materials. Coniam [30] removed every nth-word in the text to be a test item, and distractors were identified by choosing the same part of speech (e.g., noun, verb, or adjective) and similar word frequency to a tagged corpus. Mitkov and Ha [6] removed Key Terms, which are noun phrases with a frequency over a certain threshold. The distractors (e.g., hypernyms and hyponyms of the term) were identified by consulting WordNet. Mitkov and Ha demonstrated that automatic generation and manual correction of questions could be more time efficient than manual question creation alone. However, these automatic QG systems focused on generating assessment items for learning language or reading comprehension, and not in providing feedback as in the current project.

2.3 Key Phrase Extraction Techniques

Key phrases provide important information about the content of a document. Two approaches for the automatic extraction of key phrases have been studied. Supervised techniques require labeled data to train the system and tend to be more accurate but also more restricted. Unsupervised techniques do not require training sets and tend to be applicable to wider knowledge domains, but they are also less accurate.

Turney [31] introduced a system for key phrase extraction called GenEx, which is based on a set of parameterized heuristic rules tuned by a genetic algorithm. Frank et al. [32] applied a Naïve Bayes classifier for key phrase extraction on the same data used by Turney, which improved the results. Both GenEx and the Naïve Bayes classifier are examples of supervised approaches to key phrase extraction. In general, supervised approaches require an annotated training set, which is often not practical.

To eliminate the need for training data, several authors have developed unsupervised approaches to key phrase extraction. Barker and Cornacchia [33] ranked noun phrases extracted from a document by using simple heuristics based on the length and the frequency of their head noun. Bracewell et al. [34] clustered terms which share the same noun term from a list of extracted noun phrases. Another widely adopted unsupervised approach for key phrase extraction is to use graph-based ranking methods. Mihalcea and Tarau [35] represented a document as a term graph based on term relatedness; a graph based ranking algorithm is then used to assign importance scores to each term.

The Lingo algorithm [36], another unsupervised approach, is generally used for clustering web search results. It is based on singular value decomposition (SVD). The cluster-label induction phrase in Lingo involves following steps. First, a term-document matrix A is built from the input documents. Second, the term-document matrix is broken into three matrix (U, S, and V) by performing SVD, such that $A = USV^T$. Third, k column vectors of U are extracted. Each column vector refers to a cluster or latent concept. Fourth, the semantic similarities between latent concepts and single words\phrases are calculated by using classic cosine distances, $M = U_k^T * P$, where each column vector of matrix P represents a single word or phrase. Last, we choose the most similar single word or phrase as the concept label by finding the largest value in each row of matrix M. Rows of the matrix M represent latent concepts, its columns represent phrases or single words, and individual values are the cosine similarities.

2.4 Tregex in Natural Language Processing (NLP)

Tregex [37] is a powerful syntactic tree search language for identifying syntactic elements (e.g., main verbs of sentences). It has been widely used for natural language processing, including for sentence simplification [38] and question generation [8]. Tregex can be used to specify the various relations between the tree nodes. For example, "Node A is the parent of Node B" is denoted as A < B while "B is the rightmost descendent of A" is denoted as A <<- B. It also supports regular expressions. For example, NP(Noun Phrase) </area|discipline?/ matches NP is the parent of areas or areas, discipline or disciplines.

In a previous study [15], we used Tregex with cue phrases to classify citation sentences. For example, the citation (*Yi applied natural language processing techniques to sentiment analysis)* was extracted and parsed by using the Standford parser [39]. The parsed syntactic tree for this citation sentence is shown in Fig. 1.

The following Tregex rules (which belong to the Method category) were used to match the cue phrases in the tree: *apply* is a main verb and *technique* is the head of leftmost noun phrase dominated by the same Verb Phrase (VP) as the main verb

$$VP > (S > ROOT) <<, (/use|apply?/) \\ <<(NP << -(/method|technique?/)).$$

Tregex is more powerful than regular sentence classification expressions because it can capture a greater number of syntactic ways with which people write a certain type of sentence with cue phrases. In this study, we use Tregex to identify the syntactic structure of sentences and to extract them as graph nodes.

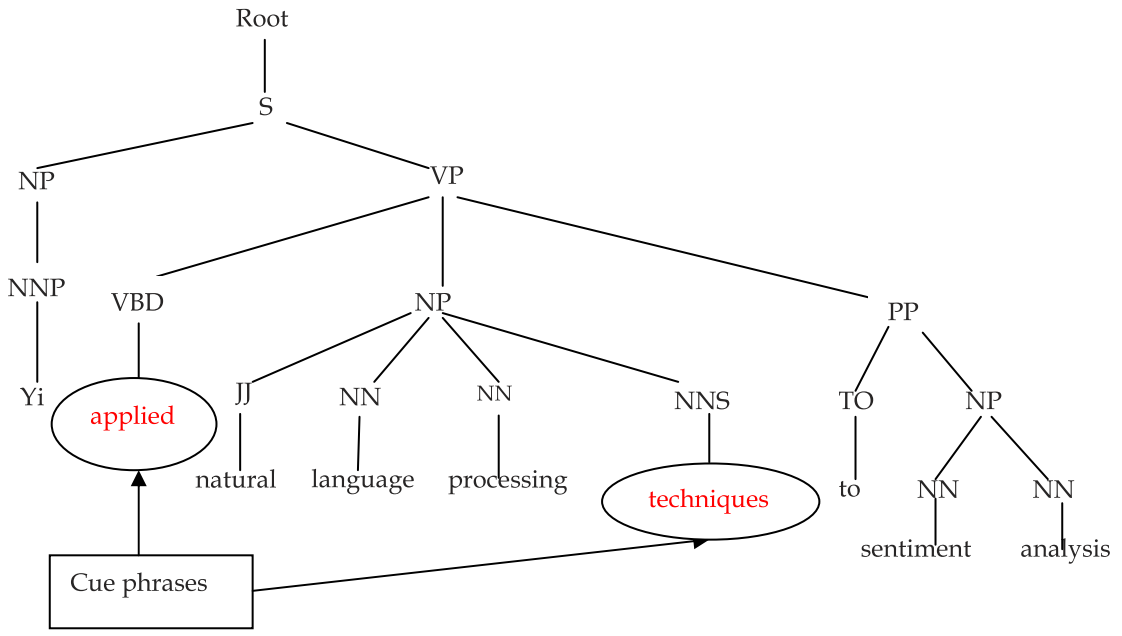

Fig. 1. A phrase structure parsed tree.

2.5 Using Wikipedia as Background Knowledge Source

Wikipedia can be seen as a lexical semantic resource that includes knowledge about named entity and domain specific terms. It has been successfully applied in many natural language processing tasks, such as named entity recognition [40], information extraction [41], named entity disambiguation [42] semantic relatedness [43] between two concepts and Wikipedia mining toolkit [44]. Wikipedia is used here for the following reasons. First, with more than three million pages, it is a very large knowledge base that covers multiple domains. The text is freely available and it can be used to filter out irrelevant key phrases. Furthermore, Wikipedia articles can be used to construct conceptual graph structures, a knowledge representation of a concept/key phrase. The process is made easier because all articles share a similar structure: section headings, paragraphs, section list, and linked lists.

Researchers can access Wikipedia in XML dumps, but they are not programmatically accessible. To overcome this problem, we used Java Wikipedia Library (JWPL) [45], an open-source, Java-based APIs. It provides programmatic access to Wikipedia during large-scale NLP tasks; it can also parse Wikipedia articles with Wikipedia markup language and convert them into a relational database.

3 System Architecture

In this section, we provide an overview of the system's pipeline architecture shown in Fig. 2.

3.1 Key Phrase Extraction

In the preprocessing stage, all input documents (literature review papers) are split into sentences. A term-sentence Vector Space Model (VSM) is then built. In stage 1, the key phrase extraction based on the VSM was performed using the Lingo algorithm. The key phrases extracted can be used in different ways. A fully automatic system could use the key phrases unfiltered or use a blacklist prepared once (not in runtime) by a domain expert, and reused as needed. If the system is not used regularly, a blacklist can be made ad hoc (as is the case in this study).

If the extracted key phrase is in abbreviated form (acronym), its full name was searched by using regular expression pattern matching techniques to increase the chances of finding matching Wikipedia articles.

3.2 Named Entity Linking and Key Phrase Classification

The extracted key phrases are linked to Wikipedia articles using JWPL. If a key phrase matches the title of a Wikipedia article, that article will be retrieved. Key phrases that cannot

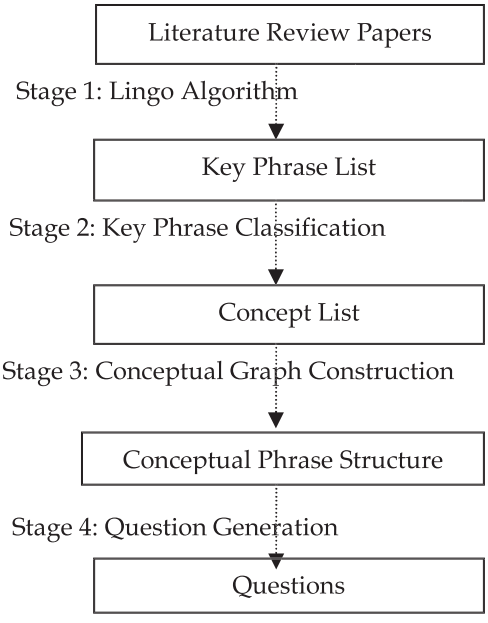

Fig. 2. Question generation in pipeline processing.

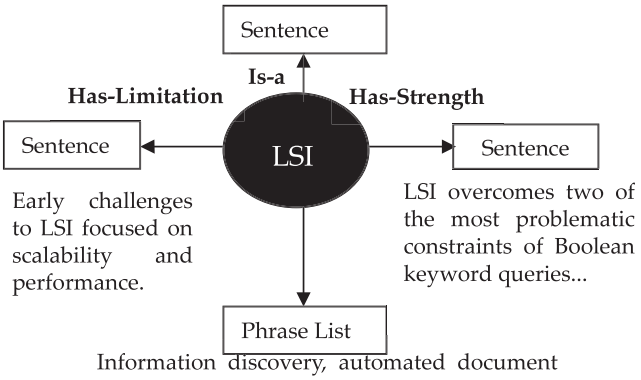

Fig. 3. A conceptual graphical structure with LSI as the key concept.

be matched to any Wikipedia article are discarded. The definition sentence refers to the first sentence that describes the key phrase in the first section of a Wikipedia page. (Examples of *definition sentence* are shown in the conceptual taxonomy in the introduction section.) From each retrieved Wikipedia page, we identify the definition sentence as one type of the conceptual taxonomy by using Tregex expression rules. We use this definition to classify the associated key phrase.

As a training set, this study used 23 literature review papers also written by postgraduate engineering students, but different from the ones in the evaluation. From this corpus 280 key phrases were generated by using the Lingo algorithm, and 180 definition sentences were extracted manually from these key phrases. Twelve common syntactical rules for classifying the definition sentence were developed from these definition sentences. The most frequent linguistic pattern for a definition sentence is that the subject complement is in the form of a possessive case. The following pattern indicates necessary linguistic units. {BE} means some form of *be*, such as *is, am, and belongs to*. The processed is the cue phrase (in bold). According to the cue phrase, we can classify the definition sentence into different category.

$$\text{The concept name} + \{\text{BE}\} + \text{the} + \textbf{study}|\textbf{technology}|\textbf{term}\ \text{of} \ldots (\text{object}).$$

The following Tregex rule is used to capture a definition sentence, which belongs to research field type:

$$S < (NP\ \$+\ (VP << (NP << (NP << -\ \textbf{/study|subarea|area|field|branch|discipline?/})) << /is|remain|are|refers?/)).$$

Another frequent pattern is that the appositive indicates the type of a definition sentence. For example, the following definition sentence refers to the key phrase *data mining*:

Data mining, a ***branch*** *of computer science and artificial intelligence, is the process of extracting patterns from data.*

We use following syntactical rule to capture this type of sentences:

$$/,/\ \$+\ (NP << (NP <<- study|subarea|area|areas|field|fields|branch|branches|disciplines|discipline)\ \$+ /,/.$$

TABLE 2
An Example of Cue Phrases in Each Edge Relation Category

Edge Relation	Cue Phrase
Has-Limitation	*Limitation, issue, challenge, problem, effect, concern, disadvantage criticism, drawback*
Has-Strength	*benefit, advantage,*
Apply-to	*application,use,apply,usage*
Include-Technology	*technologies, approaches, techniques, algorithm, methods,*

Each *definition sentence* is classified into one of five conceptual categories: *Research Field, Technology, System, Term, and Other*. If the *definition sentence* is classified as *other*, the key phrase will be discarded.

3.3 Conceptual Graph Construction

After the key phrases are classified, conceptual graph structures are then created based on information such as the section headings and the section content in a Wikipedia article. A conceptual graph structure [46] contains white nodes that represent phrase lists or sentences, and a black node that represents the key concept (key phrase). In our case, a five edge relations between the black node and the white nodes are defined. These include Is-a (Definition of the Concept) Has-Limitation (Drawback of the Concept), Has-Strength (Advantage of the Concept), Apply-to (Application of the Concept) or Include-Technology (Methods used in the Concept).

Fig. 3 shows a conceptual graph structure constructed from a Wikipedia article called *Latent Semantic Indexing*, abbreviated as LSI. The conceptual graph structure can be considered as a list of triples. A triple contains a black node, a white node, and their relation. The basic idea of this graph construction algorithm is to create each triple by setting the key phrase as the black node and a target sentence (or phrase list extracted from the target section) as the white node. A target section in a page is identified by using the cue phrases matched with the section title. Each cue phrase belongs to a relation type. Table 2 shows the cue phrase list for each edge relation. The key of this algorithm is to extract the target sentence or phrase list from the target section.

We use Tregex expression rules and the structure of a Wikipedia page (lists or sections) to extract the sentence and phrases. A target sentence containing the page title or its abbreviation is retrieved and classified as one of three relations (Apply-to, Has-Limitation, and Has-Strength) by using Tregex expression rules. We found that the most common syntactical structure for the Apply-to type is that the target sentence is normally in a passive voice, like $concept\ name + is|are + \textbf{usedfor/in} + Object$. The common syntactical structure for Has-Limitation and Has-Strength is *that the* ***advantage/disadvantage*** *of the* ***key concept*** *is that.* In addition Has-Strength type sentences have another common structure: *concept name* + **overcome/address**/+ a **problem/limitation** of something. We obtained 15 rules from the same training set as described in Section 3.2.

The structure of Wikipedia pages can be used while extracting phrase lists. The table of contents often implies a relation between the key concept and related concepts. For

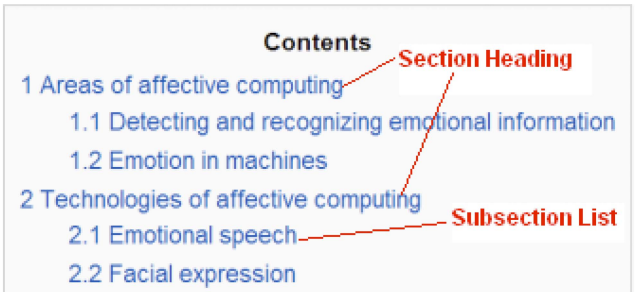

Fig. 4. A screen shot of a Wikipedia article called Affective computing.

example, in Fig. 4, the target section *Technologies of affective computing* is identified by matching the cue phrase *technologies* in the Include-Technology category. The target phrase list *Emotional speech, Facial expression, and Body gesture* can then be extracted and represented as a white node, which is connected to the black node *Affective computing* with Include-Technology relation.

The following example illustrates the construction of a graph like Fig. 3. In stage 2, the system has used the Tregex expression rule to classify *Latent Semantic Indexing* as a *Technology* with a definition sentence "*Latent Semantic Indexing is an indexing and retrieval method*." In stage 3 (conceptual graphical construction), the first triple is defined by assigning *Latent Semantic Indexing* as the black node (key concept), the *definition sentence* as the white node, and *is-a* as the edge relation between the black and white nodes. This becomes the first triple. Each section title is checked against the cue phrases in Table 2. In this case, the target section in the associated Wikipedia page called *Challenges to LSI* is chosen. The section title implies a Has-Limitation relation. The target sentence "*Early challenges to LSI focused on scalability and performance*" is extracted by matching the *LSI* with the abbreviation of the page title. The Tregex rule is used to assign the Has-Limitation as its edge relation. The second triple is thus defined by representing the extracted sentence as the white node and setting up relation as Has-Limitation.

Similarly, another target section called Additional Uses of LSI is chosen. This section title implies the Apply-to relation with LSI. Because this target section contains a list (such as information discovery, automated document classification), the title of each item is extracted. The maximum number of list items we use is three. Thus, the third triple can be defined by setting the extracted phrase list as the white node, and setting the relation as Apply-to. In our current implementation, if the phrase list is extracted for an edge relation, we don't extract the sentence for that edge relation.

3.4 Question Generation

In this section, we describe how to generate questions from a single conceptual graph structure. Our previous study [12] showed that *Judgment* questions are one of the frequent question types used by human supervisors as feedback for literature review writing. Judgment questions ask students to judge how important concepts relate to their own work/project, and hence can trigger higher level cognitive processes in Bloom's Taxonomy [47]. Thus, we adopted two principles to guide the design of our question templates. First, the questions should be specific. We place the description of a key concept in the beginning. Second, the questions should be linked to the author's research. We place the judgmental questions after or combined with the description of the key concept. Question templates are shown in Table 3

The questions are generated based on the triples in the conceptual graph. Table 3 shows the nine question generation rules defined in our repository. Each rule contains a

TABLE 3
Question Generation Rules

	Triple	Question Templates
1	Is-A(Research field,Sentence)	Sentence + What impact would the proposed project have on the field of +ConceptName+?
2	Is-A (Technology/System/Term,Sentence)	Sentence + How do you see +ConceptName+being applied in your project?
3	Has-Limitation (Concept, Phrase1/Phrase2)	ConceptName + has been known to have several limitations, in relations to issues such as Phrase1, Phrase2 and Phrase3. How do you address these issues in your project?
4	Has-Limitation (Concept,Sentence)	Sentence + How do you address these issues in your project?
5	Has-Strength (Concept,Sentence)	Sentence + Have you considered this strength, in relations to your project?
6	Has-strength (Concept,Phrase1/Phrase2)	ConceptName + has been known to have several strengths, in relations to issues such as Phrase1, Phrase2 How are these issues relevant to your project?
7	Apply-to(Concept, Phrase1/Phras2)	Do you know that + ConceptName +has been applied in Phrase1, Phrase2? How are these applications of + ConceptName +relevant to your project?
8	Apply-to (Concept, Sentence)	Do you know that + Sentence+? How is this application of relevant to your project?
9	Include-Technology (Concept, Phrase1/Phrase2)	Do you need to include other technologies used in the field of + ConceptName +, like Phrase1 andPhrase2 in your project?

TABLE 4
Five Generic Questions Chosen from Educational Learning Materials for Writing Review by Taylor [4]

1 What is the specific thesis, problem, or research question that your literature review helps to define?
2 Have you reviewed the appropriate types of publications (e.g., journals, books, government documents, popular media) for your purpose?
3 What discipline or field of research are you working in? Have you discussed how your project will contribute to that discipline or field?
4 How good was your information seeking? Has your search been wide enough to ensure you've found all the relevant material? Has it been narrow enough to exclude irrelevant material?
5 Have you critically analyzed the literature you use? Instead of just listing and summarizing items, have you assessed them, discussing strengths and weaknesses?

triple and a question template. For example, in Rule 1, the triple Is-A(Research Field, Sentence) means that the relation between the white node, which is the *definition sentence* in this case, and the black node, which is a Research field concept, is Is-A. Similarly, Rule 9 Include-Technology (Concept, Phrase1/Phrase2) means that the relations between the white node, which is a phrase list, and the black node, which belongs to one of the concepts, is Include-Technology.

The question generation process is simpler than our previous question generation approach [11]. Like other question generation approach [8], our previous approach requires complex NLP transformations in order to fill out the question template, such as sentence parsing, complex sentence splitting, and subject_auxiliary_inversion. Errors often occur during these processes. However, our current approach only needs to fill out the extracted information in the predefined question templates. Thus, it is less error prone. For example, assume that in Stage 3 a conceptual graph structure has been built based on the Wikipedia article called *Latent Semantic Indexing*. In Stage 4, we match each triple in the graph with predefined question generation rules and generate template-based questions. If the first triple with Is-a relation matches Rule 2, then the question template is filled with the concept name and sentence (white node). The following question is generated: *Latent Semantic Indexing is an indexing and retrieval method that uses a math... How do you see Latent semantic indexing being applied in your project?* Similarly, if the second triple with an Apply-to relation and a phrase list as white node matches Rule 7, the following question is then generated: *Do you know that Latent Semantic Indexing has been applied in Information Discovery, automated document Classification, and Text summarization? How are these applications of + Latent Semantic Indexing + relevant to your project?*

4 Evaluation

4.1 Participants

The participants in this study consisted of 20 engineering students doing a postgraduate or undergraduate research project and two domain experts who supervise many of the students (i.e., academics supervisors). All participants had completed a Research Methods course that required them to write a literature review for their research project. All participants signed an informed consent form approved by an ethics committee. Each of them was given a movie voucher as a reward.

TABLE 5
Six Quality Measures Are Used in This Study

QM1	This question is correctly written.
QM2	This question is clear and not ambiguous
QM3	This question helped me develop a deeper understanding of important concepts or topics related to my project.
QM4	This question helped me think or learn about topics or concepts that I did not consider before.
QM5	This question prompted me to reconsider the structure of my literature review.
QM6	This question helped me think about how I can revise and improve my literature review.

4.2 Materials

The digital versions of the 20 literature review papers written by each student were used for the study. Based on these literature review papers, 82 questions were generated by the two academic supervisors and 154 by the computer system. Papers were allocated to the actual supervisor or the one with the closest expertise, so that they could generate questions more effectively. Each paper was assigned for review to one supervisor. Besides human and computer generated questions, we obtained five common generic trigger questions from a literature review writing tutorial [4] (see Table 4).

4.3 Procedure

To evaluate our automatic QG system, we conducted a Bystander Turing Test [48]. Student participants were asked to guess whether questions were generic, written by a human supervisor, or by the computer system. They also rated the quality of these questions in terms of six quality measures (QM) shown in Table 5. QM 1 and 2 are about whether a question is understandable. QM 3 and 4 assess a question's usefulness for learning, while QM 5 and 6 assess whether the question's usefulness for improving the literature review document itself.

5 Result and Discussion

5.1 Key Phrase Classification

In the key phrase extraction stage, the computer system extracted 289 unique key phrases from the 20 literature review papers. This is excluding 67 duplicate key phrases (which occurs when, for example, the same key phrase was extracted from two papers). As we mentioned in Section 1, each key phrase belongs to one of five predefined conceptual categories: Research Field, Technology, Term, System, and Other. Table 6 shows the classification result on

TABLE 6
Key Phrase Classification Confusion Matrix

True / Predicted	Research Field	Tech	Term	System	Other
Research Field	19				9
Technology		35			7
Term	1		7		
System			1	14	7
Other		4	4		181
Total num. of True Category	20	39	12	14	204

the 289 key phrases. Apart from the category Other, we can see that there are more Technology (39) and Research Field (20) key phrases than Term (12) and System (14).

Two people independently annotated the 289 key phrases. Cohen's Kappa coefficient was 0.58 ($n = 5$; $N = 289$; $k = 2$). To measure the classifier's performance we calculated a balanced F-score, taking into account precision and recall.

Technology (F-score: 0.86), System (F-score: 0.8), and Research Field (F-score: 0.79) were easier to identify than Term (F-score: 0.7). This may be because Term contains relatively more implicit definition sentences. For example, the following sentence was extracted for the key phrase "*denial-of-service attack*": *A denial-of-service attack (DoS attack) or distributed denial-of-service attack (DDoS attack) is an attempt to make a computer resource unavailable to its intended users.* This sentence is difficult to identify as a Term because it contains no cue phrases. Another example, for the key phrase "feature extraction technology," is *In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction.* This sentence is also difficult to classify as technology if the system does not know that *dimensionality reduction* is a technology. Implicit definition sentences affect the recall index.

Another typical problem is when one definition sentence is matched with two or more rules which belong to different categories. For example, take the definition sentence for Data Mining: "*Data mining, a **branch** of computer science and artificial intelligence, is the **process** of extracting patterns from data.*" This sentence was matched with a rule in the Research Field category because the appositive contains ***branch*** as a cue phrase. It was also matched with a rule in the Technology category because the head of the leftmost noun phrase in the sentence contains the cue phrase ***process***. Hence, this type of sentence is difficult to classify. Misclassifications in this case affect the precision index. Currently, to overcome this problem we simply randomly select one category.

We have discarded the "Other" questions since this type of question are useless for question generation. Then, we manually filtered out 10 key phrases and used the remaining 75 to construct conceptual graphical structures based on the content of the corresponding Wikipedia articles. It is easy for a human to identify these key phrases (algorithm, collaboration, compiler, gray, measurement, ownership, research, goal, lingua franca, bought, and review) as not likely to produce valuable questions in an engineering research project. They are too general or not specific enough. In a real scenario, where different cohorts work on the same topic, a blacklist could have been used for a fully automated process. We point out that the pedagogical value is evaluated using this semiautomated process.

As a result, the computer system generated 154 questions. From this pool of questions, we randomly sampled five questions at most for each paper; 96 questions were used across the 20 literature reviews. In total, 278 questions were evaluated, including 96 questions generated by the computer system, 82 questions from the human supervisors, and 100 generic questions.

5.2 Question Authorship Evaluation

For each of the questions, participants were asked to guess whether it was a computer generated question, a question from a human supervisor, or a generic one. Table 7 shows that the F-scores were 0.43 for supervisor-generated

TABLE 7
Participants' Classification of Question Authorship

Real / Prediction	Human Supervisor (F-score:0.43)	Computer System (F-score:0.53)	Generic Question (F-score:0.79)	Total (Predication)
Supervisor	38 (46%)	42 (44%)	16 (16%)	96
Computer System	34 (41%)	50 (52%)	9 (9%)	93
Generic Question	10 (13%)	4 (4%)	75 (75%)	89
Total (Real)	82	96	100	

TABLE 8
Evaluation of Computer Generated, Human-Generated, and Generic Questions

Question Producer / Quality Measure	Computer System Mean (SD)	Human Supervisor Mean (SD)	Generic Mean (SD)
QM1: Correctness	4.14 (.85)	4.26 (.83)	4.40 (.72)
QM2: Clarity	4.05 (.89)	4.18 (.85)	4.30 (.85)
QM3: Useful for learning important concepts	3.68 (.81)	3.56 (.83)	3.21 (.84)
QM4: Useful for learning new concepts	3.30 (.77)	3.32 (.78)	2.97 (.83)
QM5: Useful to restructure document	2.94 (.86)	3.02 (.87)	3.18 (.87)
QM6: Useful to improve document	3.39 (.75)	3.43 (.81)	3.21 (.85)

questions, 0.53 for computer-generated questions, and 0.79 for generic questions. This shows that, as expected, generic questions were the easiest to identify.

Participants were sometimes still able to distinguish between human and computer generated questions. This was perhaps because the human questions were often more concise, and sometimes the computer system questions (especially the lengthier ones) contained grammatical errors. Nonetheless, the F-scores show that distinguishing human generated from computer generated questions was not easy: 44 percent (42 out of 96) of the computer generated questions were wrongly identified as supervisor generated, while 41 percent (34 out of 82) of the supervisor questions were wrongly identified as being from the computer system. Moreover, the Cohen's Kappa coefficient ($k = 0.38$) shows a low agreement on the classification result between the students' perception and the real category, which indicates that the students experienced moderate difficulties in distinguishing the questions.

There are two major possible reasons for this classification result. First, the computer-generated questions were specific and related to the major topics of the participants' literature reviews.

Second, both human and computer generated questions used abstract concepts (especially for questions in the categories of Application, and Limitation, and Strength). For example, the following question was generated by a human supervisor: *What are the advantages and disadvantages of implementing recommender technologies on a mobile device?* This type of question often requires student-writers to critically identify the strength or limitations of a method/theory/system.

5.3 Comparative Evaluation of Three Question Producers

The student participants rated the quality of each question on a five-point Likert point scale along six quality measures. Higher scores reflect stronger agreement with the quality measure statements; the midpoint, 3, reflects a neutral stance. The results are displayed in Table 8. Generic questions were perceived to be slightly more grammatically correct (QM1) and also clearer (QM2) than the other question types. However, analyses of variance (ANOVA) showed that this trend was only marginally significant for QM1, $F(2,275) = 2.03$, $p = .07$, and not significant for QM2, $F(2,275) = 2.03$, $p = .13$.

QM3 and QM4 were designed to assess the perceived pedagogical usefulness of the questions. In both of these quality measures, the generic questions (QM3: $M = 3.21$, $SD = 0.84$; QM4: $M = 2.97$, $SD = 0.83$) were rated to be least useful, while the computer- (QM3: $M = 3.68$, $SD = 0.81$; QM4: $M = 3.30$, $SD = 0.77$) and supervisor-generated (QM3: $M = 3.56$, $SD = 0.83$; QM4: $M = 3.32$, $SD = 0.78$) questions were given similar higher scores. ANOVA results indicated that some of these differences were statistically significant for QM3, $F(2,275) = 8.37$, $p < .001$, as well as for QM4, $F(2,275) = 5.79$, $p = .003$.

To test the difference between pairs of question types, we used Fisher's least significant difference (LSD). For all LSD tests in this paper, we adopted an alpha level of .05. This test indicated that computer-generated questions significantly outperformed generic questions, on both QM3 (MD(0.47) > LSD(0.23)) and QM4 (MD(0.33) > LSD (0.22)). Meanwhile, supervisor-generated questions also significantly outscored generic questions on both QM3 (MD(0.35) > LSD(0.24)) and QM4 (MD(0.35) > LSD(0.23). The difference between computer-generated and supervisor-generated questions was not significant for QM3 (MD(0.12) < LSD(0.25), nor for QM4 (MD(0.01) < LSD(0.24)).

QM5 is about the usefulness of the questions to improve the structure of the literature review. In this case, generic questions slightly outscored both supervisor- and computer-generated questions. However, ANOVA indicated that these differences were not significant, $F(2,275) = 1.97$, $p = .14$. QM6 assesses the usefulness of the question for revising and improving the literature review. Supervisor questions slightly outscored computer-generated ones, which in turn outscored generic questions. However, once again ANOVA indicated that these differences were not significant, $F(2,275) = 1.91$, $p = .15$

Taken together, these results show that while our computer system questions were judged as being relatively less clear, they were perceived to be pedagogically as useful as supervisor-generated questions, and more pedagogically useful than generic questions. This indicates that our

TABLE 9
Pearson Correlation Coefficient of Six Quality Measures

	QM1	QM2	QM3	QM4	QM5	QM6
QM1	1					
QM2	0.769	1				
QM3	0.276	0.213	1			
QM4	0.097	0.101	0.611	1		
QM5	0.0837	0.129	0.273	0.572	1	
QM6	0.132	0.081	0.468	0.584	0.533	1

system was successful in identifying important key concepts in the students' literature reviews, and providing meaningful questions about those key concepts.

It is possible that some questions produced by the human supervisors had relatively low pedagogical utility because they were unfamiliar with the topics in some of students' literature reviews. This is reflected in the fact that the human supervisors sometimes used generic questions (as shown in Table 7, 13 percent of the supervisor questions were wrongly identified as generic). However, we argue that this reflects a realistic situation: instructors who teach academic writing courses are not always familiar with the topics in their students' writing. Hence, we can say that, in terms of pedagogical utility, the performance of our computer system is comparable to the human instructors in typical academic writing courses.

The computer-generated questions were rated as relatively less clear (QM1 and QM2). Part of the reason might be that some of the phrase lists extracted were not informative enough. For example, the following question was generated based on a triple which included the key phrase "*Latent semantic analysis,*" the phrase list (white node) "*polysemy, bag of words model, and probabilistic model,* and the relation "Has-Limitation": *"Latent semantic analysis has been known to have several limitations, in relation to issues such as polysemy, bag of words model, and probabilistic model. How do you address these issues in your project?"*. This question may be relatively unclear because it does not convey adequate information about *bag of words model* and *proabilistic model*. Nonetheless, it is important to note that the computer-generated questions still obtained average scores above 4, which indicates that on average, the participants perceived the questions to be clear and understandable.

5.4 Impact of Manually Filtering Misclassified Key Phrases

The blacklist filtering of the 10 key phrases (out of 85) in stage 1 can be considered to affect the evaluation results, particularly the human classification results in Table 7 and the pedagogical value of computer-generated questions in Table 8.

Here, we performed a worse case scenario analysis, where these key phrases generated the 20 least valuable questions (and therefore correctly classified as computer generated). In this scenario each question got the lowest score (1) in QM 3, 4, 5, and 6 (the pedagogical value). As a result, the question classification result on system category was increased from an F-score of 0.53 (see Table 7) to 0.611 (precision: 0.619 and recall: 0.603). On the other hand, the overall quality measure score for the system was decreased. After performing a series of ANOVA and Fishers' LSD tests, we found that human supervisors significantly outperformed the system in QM 3, 4, 5, and 6. In addition, ANOVA indicated that generic questions significantly outperformed system-generated questions in QM5, $F(2,295) = 10.6,\ p = .004$. These results indicate that the system's performance would have decreased if we had not used a man-made blacklist.

5.5 Correlation Analysis of Quality Measures

We used the scores for the computer system given by the students to perform a correlation analysis of the six quality measures. The results are displayed in Table 9. As we expected QM1 and QM2—which are both about the correctness—are strongly correlated ($r = 0.769$). QM3 and QM4, both about the usefulness for learning, are moderately correlated ($r = 0.611$). We also observed that QM4, QM5, and QM6 are moderately correlated to each other. This indicates that learning new concepts (QM4) and reconstructing new concepts (QM5) are perceived to be related to revising and improving the literature review (QM6).

5.6 Investigation on the Impact of Computer Generated Questions by Groups

To examine whether participants' research experience influenced their evaluation of the questions, we divided the 20 participants into two groups: Group 1 was composed of first year students (see Table 10), while Group 2 was composed of second or third year students (see Table 11).

TABLE 10
Comparisons of Mean Scores among Six Quality Measures for the First Year Students

Question Producer / Quality Measure	Computer System		Human Supervisor		Generic	
	Mean (SD)	N	Mean (SD)	N	Mean (SD)	N
QM1	4.08 (.95)	49	4.23 (.87)	39	4.54 (.61)	50
QM2	4.00 (.94)	49	4.08 (.93)	39	4.42 (.76)	50
QM3	3.94 (.59)	49	3.87 (.52)	39	3.26 (.85)	50
QM4	3.51 (.58)	49	3.64 (.45)	39	2.88 (.82)	50
QM5	2.90 (.82)	49	2.97 (.87)	39	3.12 (.92)	50
QM6	3.53 (.62)	49	3.64 (.78)	39	3.00 (.88)	50

TABLE 11
Comparisons of Mean Scores among Six Quality Measures for Higher Level Students

Question Producer / Quality Measure	Computer System		Human Supervisor		Generic	
	Mean (SD)	N	Mean (SD)	N	Mean (SD)	N
QM1	4.19 (.74)	47	4.28 (.80)	43	4.26 (.80)	50
QM2	4.11 (.84)	47	4.28 (.77)	43	4.18 (.92)	50
QM3	3.40 (.92)	47	3.28 (.96)	43	3.16 (.84)	50
QM4	3.04 (.93)	47	2.90 (.87)	43	3.06 (.84)	50
QM5	2.98 (.90)	47	3.07 (.88)	43	3.24 (.82)	50
QM6	3.23 (.84)	47	3.13 (.83)	43	3.42 (.78)	50

For each group, ANOVA and Fishers' LSD tests were performed to examine differences between the three question types.

For second and third year students (Table 11), there was no significant difference between the three question types in any of the quality measures. For first year-students, generic questions were perceived to be more grammatically correct (QM1) and clearer (QM2) than computer generated questions, but there was no significant difference between computer-generated questions and human supervisor questions, $F(2,137) = 0.16$, $p = .85$ for QM1 and $F(2,137) = 0.47$, $p = .62$ for QM2. For QM3 and QM4, computer generated questions significantly outperformed generic questions, while there was no significant difference between computer-generated questions and human supervisor questions, $F(2,137) = 0.89$, $p = .42$ for QM3 and $F(2,137) = 0.89$, $p = .42$ for QM4. For QM5, there were no significant differences between any of the question types, $F(2,137) = 0.42$, $p = .66$ For QM6, computer-generated questions significantly outscored generic questions; but there was no significant difference between computer generated and supervisor questions.

Surprisingly, compared to second/third-year students, first-year students gave higher scores for questions (QM3 and QM4). ANOVA results indicated that first-year students gave significantly higher scores for computer generated question in QM3 ($F(1,94) = 11.477, p = .02$) and QM4 ($F(1,94) = 7.851, p = .01$). This indicates that the computer-generated questions were more useful for first-year students in terms of developing deeper understanding of concepts and learning new concepts. This might be because first-year research students are less familiar with the important concepts in their area of study. This also indicates that the Wikipedia knowledge base is more useful for the students who are new in their research area.

5.7 Relation Type Evaluation

In order to examine the usefulness of each relation type for question generation, we evaluated the five relation types: Definition, Apply-to, Has-Limitation, Has-Strength, and Include-Technology. We used the 96 computer generated questions rated by the students described before. There were 50 questions generated from Definition relation type, 26 from Apply-to, nine from Has-Limitation, five from Has-Strength, and six from Include-Technology. Table 12 shows the average score for each relation type according to the quality measure. We observed that all five relation types were well regarded in QM1 and QM2 because these two quality measure requires only surface level correctness without considering pedagogical aspects. Has-Strength and Has-Limitation relation types obtained better scores than other relation types in QM3, QM4, and QM6 (above 3.67). Writers generally "agreed" or "almost agreed" that questions had good pedagogical value in terms of helping to develop deep understanding of important concepts, learn about new concepts, and revise the literature review paper. The Definition type obtained relative poor performance: the lowest rank in QM3 ($M = 3.56$) and the second lowest rank in QM4 ($M = 3.16$) and in QM6 ($M = 3.32$). After performing a series of ANOVA and LSD tests, we found that Has-Strength significantly outperformed Apply-to and Include-Technology in terms of QM4 ($F(4,91) = 3.15, p < .001$). However, all relation types were perceived to have pedagogical value in terms of deep understanding of important concepts (QM3).

These results suggest that the five relation types were perceived to be useful for learning important concepts. The Has-Strength and Has-Limitation questions were particularly valuable because these question types addressed critical analysis issues on literature review writing. However, the number of questions generated from these two types is limited due to the limitation of our current conceptual graph construction approach, which uses the section headings to indicate the target sentence of the concept. Sometimes the section content could still include

TABLE 12
Comparisons of Scores for Each Relation

Type / Quality Measure	Definition	Apply-to	Has-Limit	Has-Strength	Include-Tech
QM1	4.14	4.12	4.11	4.60	3.83
QM2	4.08	3.96	4.00	4.60	3.83
QM3	3.56	3.77	3.67	4.40	3.67
QM4	3.16	3.35	3.67	4.20	3.00
QM5	2.88	2.81	3.33	3.60	2.83
QM6	3.32	3.31	3.67	3.80	3.50

the target sentence, even when the section heading does not imply the strength or limitation of a concept.

6 Conclusions and Future work

In this paper, we presented an intelligent question generation tool for supporting literature review writing. The tool is based on conceptual graph structures constructed from Wikipedia articles. The questions are intended to prompt students to reflect on key concepts in their area of study. In order to evaluate the quality of the computer generated questions, we compared them to questions generated by two human supervisors and also to a set of generic questions. The computer-generated questions were perceived to be as pedagogically useful as human supervisor questions, and more useful than generic questions. In addition, the computer-generated questions were considered to be more useful by first-year students, compared to second and third year students. Findings of this study also suggested that Wikipedia can play an important role in identifying the substantive concepts/key phrases in a literature review, and filtering out less relevant concepts.

One problem we encountered was that some of the definition sentences extracted from Wikipedia were implicit, or matched with multiple rules from different key phrase categories. One possible solution is to use a supervised machine learning approach to classify the sentences; this, however, would require more human resources for labeling and defining useful features.

The headings of Wikipedia sections were found to be useful for identifying the target sentence or phrase list in the content. Using information from section headings in Wikipedia reduces the computational cost needed to scan and classify each sentence in a Wikipedia article. The drawback of using section headings is that some target sentences or phrase lists cannot be extracted because they appeared in sections whose heading does not contain the cue phrases.

One of limitation of our current AQG system is that it is domain dependent, because we only defined a limited number of concepts for generating questions (e.g., Research Field, Technology, Term, and System). Although these concepts are common in the science disciplines, they may not be directly suitable for other humanity or social science disciplines (e.g., English literature). Another limitation is that we may not apply this approach to other applications, such as reading comprehension. For such applications, the knowledge representation is commonly constructed from the reading material rather than other resources, such as Wikipedia.

In this study, questions were only evaluated by the student-writers. Sometimes, the student's perception and the true value might mismatch. In the future work, we will evaluate the questions based on human experts' perception.

Acknowledgments

This project was partially supported by the University of Sydney TIES grant, an Australian Research Council Discovery Project grant (DP0986873), and a Google Research Award for measuring the impact of feedback on the writing process. The authors thank the anonymous reviewers for their helpful feedback.

References

[1] G. Graswell, *Writing for Academic Success: A Postgraduate Guide.* SAGE, 2008.
[2] B. Steward, "Writing a Literature Review," *The British J. Occupational Therapy,* vol. 67, pp. 495-500, 2004.
[3] M. Afolabi, "The Review of Related Literature in Research," *Int'l J. Information and Library Research,* vol. 4, pp. 59-66, 1992.
[4] D. Taylor, "The Literature Review: A Few Tips on Conducting It," http://www.writing.utoronto.ca/advice/specific-types-of-writing/literature-review, July 2011.
[5] V. Rus and A.C. Graesser, "The Question Generation Shared Task and Evaluation Challenge," *Proc. Sixth Int'l Natural Language Generation Conf.,* pp. 251-257, 2009.
[6] R. Mitkov and L.A. Ha, "Computer-Aided Generation of Multiple-Choice Tests," *Proc. HLT-NAACL Workshop Building Educational Applications Using Natural Language Processing,* pp. 17-22, 2003.
[7] H. Kunichika, T. Katayama, T. Hirashima, and A. Takeuchi, "Automated Question Generation Methods for Intelligent English Learning Systems and Its Evaluation," *Proc. Int'l Conf. Computers in Education,* pp. 1117-1124, 2002.
[8] M. Heilman and N.A. Smith, "Good Question! Statistical Ranking for Question Generation," *Proc. Ann. Conf. North Am. Chapter of the Assoc. for Computational Linguistics - Human Language Technologies,* pp. 609-617, 2010.
[9] J. Mostow and W. Chen, "Generating Instruction Automatically for the Reading Strategy of Self-Questioning," *Proc. Int'l Conf. Artificial Intelligence in Education,* pp. 465-472, 2009.
[10] P. Mannem, R. Prasad, and A. Joshi, "Question Generation from Paragraphs at UPenn: QGSTEC System Description," *Proc. Third Workshop Question Generation,* pp. 84-91, 2010.
[11] M. Liu, R.A. Calvo, and V. Rus, "Automatic Question Generation for Literature Review Writing Support," *Proc. 10th Int'l Conf. Intelligent Tutorial Systems,* pp. 45-54, 2010.
[12] M. Liu and R.A. Calvo, "Question Taxonomy and Implications for Automatic Question Generation," *Proc. Int'l Conf. Artificial Intelligence in Education,* pp. 504-506, 2011.
[13] A.C. Graesser and N.K. Person, "Question Asking during Tutoring," *Am. Education Research J.,* vol. 31, pp. 104-137, 1994.
[14] W. Lehnert, C. Cardie, and E. Riloff, "Analyzing Research Papers Using Citation Sentences," *Proc. 12th Ann. Conf. Cognitive Science Soc.,* pp. 511-518, 1990.
[15] N.K. Person and A.C. Graesser, "Human or Computer? AutoTutor in a Bystander Turing Test," *Proc. Sixth Int'l Conf. Intelligent Tutoring Systems (ITS '02),* pp. 821-830, 2002.
[16] R. Haswell, "The complexities of Responding to Student Writing; or, Looking for Shortcuts via the Road of Excess," *Across the Disciplines,* vol. 3, 2006.
[17] M.D. Shermis and J.C. Burstein, *Automated Essay Scoring: A Cross-Disciplinary Perspective,* vol. 16. MIT, 2003.
[18] R. Williams and H. Dreher, "Automatically Grading Essays with MarkIT©," *Proc. Informing Science Conf.,* 2004.
[19] J. Anderson, *Mechanically Inclined: Building Grammar, Usage, and Style into Writer's Workshop.* Stenhouse, 2005.
[20] E.C. Thiesmeyer and J.E. Theismeyer, *Editor: A System for Checking Usage, Mechanics, Vocabulary, and Structure.* Modern Language Assoc., 1990.
[21] M.A. Britt, P. Wiemer-Hastings, A.A. Larson, and C.A. Perfetti, "Using Intelligent Feedback to Improve Sourcing and Integration in Students' Essays," *Int'l J. Artificial Intelligence,* vol. 14, pp. 359-374, 2004.
[22] T.K. Landauer, D.S. McNamara, S. Dennis, and W. Kintsch, *Handbook of Latent Semantic Analysis.* Lawrence Erlbaum, 2007.
[23] P. Wiemer-Hastings and A.C. Graesser, "Select-a-Kibitzer: A Computer Tool that Gives Meaningful Feedback on Student Compositions," *Interactive Learning Environments,* vol. 8, pp. 149-169, 2000.
[24] D. Wade-Stein and E. Kintsch, "Summary Street: Interactive Computer Support for Writing," *Cognition and Instruction,* vol. 22, pp. 333-362, 2004.
[25] T.K. Landauer, D. Laham, and P.W. Foltz, "The Intelligent Essay Assessor," *IEEE Intelligent Systems,* vol. 15, no. 5, pp. 27-31, Sept./Oct. 2000.
[26] B. Lemaire and P. Dessus, "A System to Assess the Semantic Content of Student Essays," *J. Educational Computing Research,* vol. 24, pp. 305-320, 2001.

[27] J. Villalon, P. Kearney, R.A. Calvo, and P. Reimann, "Glosser: Enhanced Feedback for Student Writing Tasks," *Proc. IEEE Eighth Int'l Conf. Advanced Learning Technologies (ICALT '08)*, pp. 454-458, 2008.
[28] R.A. Calvo and R.A. Ellis, "Students' Conceptions of Tutor and Automated Feedback in Professional Writing," *J. Eng. Education*, pp. 427-438, 2010.
[29] J.H. Wolfe, "Automatic Question Generation from Text - An Aid to Independent Study," *SIGCUE Outlook*, vol. 10, pp. 104-112, 1976.
[30] D. Coniam, "A Preliminary Inquiry into Using Corpus Word Frequency Data in the Automatic Generation of English Language Cloze Tests," *CALICO J.*, vol. 14, no. 2, pp. 15-33, 1997.
[31] P.D. Turney, "Learning Algorithms for Keyphrase Extraction," *Information Retrieval*, vol. 2, pp. 303-336, 2000.
[32] E. Frank, G.W. Paynter, I.H. Witten, C. Gutwin, and C.G. Nevill-Manning, "Domain-Specific Keyphrase Extraction," *Proc. 16th Int'l Joint Conf. Artificial Intelligence*, 1999.
[33] K. Barker and N. Cornacchia, "Using Noun Phrase Heads to Extract Document Keyphrases," *Proc. 13th Biennial Conf. Canadian Soc. Computational Studies of Intelligence*, 2000.
[34] D.B. Bracewell, F. Ren, and S. Kuriowa, "Multilingual Single Document Keyword Extraction for Information Retrieval," *Proc. Int'l Conf. Natural Language Processing and Knowledge Eng.*, pp. 517-522, 2005.
[35] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," *Proc. Conf. Empirical Methods in Natural Language Processing*, 2004.
[36] S. Osinski, J. Stefanowski, and D. Weiss, "Lingo: Search Results Clustering Algorithm Based on Singular Value Decomposition," *Proc. Int'l Conf. Intelligent Information Systems*, 2004.
[37] R. Levy and G. Andrew, "Tregex and Tsurgeon: Tools for Querying and Manipulating Tree Data Structures," *Proc. Fifth Int'l Conf. Language Resources and Evaluation*, 2006.
[38] M. Heilman and N.A. Smith, "Extracting Simplified Statements for Factual Question Generation," *Proc. Third Workshop Question Generation*, 2010.
[39] D. Klein and C.D. Manning, "Fast Exact Inference with a Factored Model for Natural Language Parsing," *Proc. Advances in Neural Information Processing Systems 15 (NIPS)*, pp. 3-10, 2003.
[40] R. Bunescu and M. Pasca, "Using Encyclopedic Knowledge for Named Entity Disambiguation," *Proc. 11th Conf. European Chapter of Assoc. for Computational Linguistics*, pp. 9-16, 2006.
[41] M. Ruiz-Casado, E. Alfonseca, and P. Castells, "Automatic Assignment of Wikipedia Encyclopedic Entries to WordNet Synsets," *Proc. Atlantic Web Intelligence Conf. (AWIC)*, 2005.
[42] S. Cucerzan, "Large-Scale Named Entity Disambiguation Based on Wikipedia Data," *Proc. Joint Conf. Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 2007.
[43] I. Gurevych, C. Müller, and T. Zesch, "What to Be? - Electronic Career Guidance Based on Semantic Relatedness," *Proc. 45th Ann. Meeting of the Assoc. of Computational Linguistics*, 2007.
[44] D. Milne and I.H. Witten, "An Open-Source Toolkit for Mining Wikipedia," *Proc. New Zealand Computer Science Research Student Conf.*, 2009.
[45] T. Zesch, C. Müller, and I. Gurevych, "Extracting Lexical Semantic Knowledge from Wikipedia and Wiktionary," *Proc. Sixth Int'l Conf. Language Resources and Evaluation*, 2008.
[46] A.M. Olney, W.L. Cade, and C. Williams, "Generating Concept Map Exercises from Textbook," *Proc. ACL HLT Workshop Innovative Use of NLP for Building Educational Applications*, 2011.
[47] B. Bloom, *Taxonomy of Educational Objectives: The Classification of Educational Goals: Handbook I, Cognitive Domain.* Longman, 1956.
[48] N.K. Person and A.C. Graesser, "Human or Computer? AutoTutor in a Bystander Turing Test," *Proc. Sixth Int'l Conf. Intelligent Tutoring Systems*, 2002.

Ming Liu received the master's degree of information technology from the University of Tasmania in 2006. Currently, he is working toward the PhD degree in the Learning and Affecting Technologies Engineering Group in the School of Electrical and Information Engineering, University of Sydney. He is a current recipient of the Australia Research Council Discovery Project Grant and a Google Research Award. His main research interests include question generation, natural language processing, and learning technologies.

Rafael A. Calvo received the PhD degree in 2000 in artificial intelligence applied to automatic document classification. He is an associate professor in the School of Electrical and Information Engineering, University of Sydney, and the director of the Learning and Affect Technologies Engineering (Latte) research group. He has also worked at Carnegie Mellon University and the Universidad Nacional de Rosario, and as a consultant for projects worldwide. He is the author of numerous publications in the areas of affective computing, learning systems, and web engineering, and the recipient of four teaching awards. He is an associate editor of *IEEE Transactions on Affective Computing*. He is a senior member of the IEEE, the IEEE Computer Society, and the IEEE Education Sociey.

Anindito Aditomo received the undergraduate degree in psychology and the master's of philosophy degree in education. He is working toward the PhD degree at the Centre for Research on Computer Supported Learning and Cognition, Faculty of Education and Social Work, Sydney University. He is a current recipient of the University of Sydney International Scholarship. Prior to the PhD degree, he was a lecturer with the Faculty of Psychology, University of Surabaya.

Luiz Augusto Pizzato received the PhD degree in computer science from Macquarie University, Australia, in 2009. He is a postdoctoral researcher in the Computer Human Adaptive Interaction (CHAI) research lab at the University of Sydney. His research interests span the fields of natural language processing, information retrieval, and recommender systems.