International Journal of Advance Research in Science and Engineering
Vol. No. 5, Issue No. 07 , July 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

TEXT ELABORATION USING AUTOMATIC GAP-FILL QUESTION AND WH-TYPE QUESTION GENERATION: A PRELIMINARY APPROACH

Surekha Baban Gite[1], Prof.Manjushree D. Laddha[2]

[1]*Computer Engineering, Dr. Babasaheb Ambedkar Technological University- Lonere, (India)*

[2]*Computer Engineering, Dr. Babasaheb Ambedkar Technological University- Lonere, (India)*

ABSTRACT

One always seeks easier and understandable way for teaching and learning. This paper will present an automatic question generation system that can generate gap-fill questions and WH-type question from a document. Gap-fill questions are fill-in-the-blank questions with multiple choices, one is answer and three are distractors, provided. The system finds the informative sentences from the document and generates gap-fill questions from them by first blanking keys from the sentences and then determining the distractors for these keys. Syntactic and lexical features are used in this process without relying on any external resource apart from the information in the document.

Text adaptation is an activity performed by teachers for reading comprehension. There are two general approaches for it, first is Text Simplification (TS) and second is Text Elaboration (TE). TE aims at clarifying, explaining information and making connections explicit in texts. An automatic question categorization system which assigns wh-question labels to verbal arguments in a sentence is a new approach for TE.

This annotation is similar to semantic role labeling, approached successfully via statistical language processing techniques, and addresses two key research questions: (1) which machine-learning algorithm presents the best results? (2) Which problems this task presents and how to overcome them?

Keywords: AQG, Distractor Selection, Key Selection, Natural Language Processing, Sentence Selection, Text Elaboration.

I INTRODUCTION

Assessment and evaluation are fundamental part of teaching, to measure a student's mastery of the material and to identify areas where they may need improvement. Gap-fill questions and WH-type questions are effective way for that. Gap-fill questions are the Fill-in-the-blank questions with multiple options, where one or more words are removed from a sentence and potential answers are listed. Preparing these questions manually will

take a lot of time and effort. This is where automatic gap-fill question generation (GFQG) and WH-type question generation from a given text is useful.

In AQG, system generates GFQ or question sentence (QS) from gap fill sentence (GFS). The word or words which are removed from a GFS to form the QS is referred to as the key while the three alternatives in the question are called as distractors.

Text Adaptation helps in language skills development [1]. It can benefit second-language learners and children learning to read texts of different genres. Also, text adaptation can benefit audiences with special needs, such as low-literacy readers, people undertaking Distance Education. Studies in Text Adaptation try to answer two questions: What is modified? and How is it modified? Researchers have investigated modifications at different linguistic levels can be done. These are: phonology, lexis, syntax, and discourse.

Concerned to second question, there are two general approaches of text adaptation: Text Simplification (TS) and Text Elaboration (TE) [2]. The first can be defined as any task that reduces the complexity of a text (lexical and syntactic complexity), while trying to preserve meaning and information [3].TE aims at clarifying and explaining information and making connections explicit in a text, for example, providing synonyms for words known to only a few speakers of a language or short definitions for complex concepts. TS enhance text readability, i.e., it makes the text easier to be read, TE is devoted to enhance text comprehensibility, i.e., it helps to increase easiness to understand concepts in a text.

AQG system takes a document with its title as an input and produces a list of gap-fill and WH-type questions as output. It doesn't use any external resource for distractors selection [4] [5], making it adaptable to text from any domain. Its simplicity makes it useful not only as an aid for teachers to prepare gap-fill and WH-type questions but also for students who need an automatic question generator to aid their learning from a textbook.

II BACKGROUND AND RELATED WORK

There already exists a large body of work in automatic question generation (AQG) for educational purposes dating back to the Autoquest system [6], which used an entirely syntactic approach to generate Wh-questions, from individual sentences. In addition to Autoquest, there are other systems created for Wh-question generation using approaches including transformation rules [9], template based generation and overgenerate-and-rank [5].

Earlier works in GFQG [4] [7] [8] [9] [10] have worked in the domain of English language learning. Gap-fill questions have been generated to test student's knowledge of English in using the correct verbs [7], prepositions [8] and adjectives [9] in sentences. [10] and [4] have generated GFQs to teach and evaluate student's vocabulary.

In this paper, we move away from the domain of English language learning and work on generating gap-fill questions from any document of textbook used for Advanced Placement (AP) exams.

The aim is to go through the textbook, identify informative sentences and generate gap-fill questions from them to aid students' learning. The scans through the text in the chapter and identifies the informative sentences in it using features inspired by summarization techniques. Questions from these sentences (GFSs) are generated by first choosing a key in each of these and then finding appropriate distractors for them from the chapter.

There are prominent studies on TE for the English language. The Automated Text Adaptation Tool [1] [11], is a Natural Language Processing application for educational purposes, which is used by **English language learners (ELLs)** in content-area classrooms beyond elementary school. Since ELLs must learn the specialized, academic vocabulary which often includes low-frequency, more difficult words far beyond their English reading level, Text Adaptor includes an easier synonym adjacent to a difficult word and marginal notes (a kind of summary) translated into Spanish, besides other functionalities related to Text Simplification. [2] investigated the effects of lexical simplification and elaboration on sentence comprehension and incidental vocabulary acquisition by Japanese learners of English as a second language (L2). The modifications were carried out substituting unknown words (very low frequency words) with high-frequency synonyms, and adding synonyms of the unknown words in opposition to them, respectively. The results of this study suggest that both lexical simplification and elaboration can improve learner comprehension at the sentence level. However, lexical elaboration resulted in incidental vocabulary acquisition, while simplification did not; and learners of higher proficiency benefited more from lexical elaboration in terms of the acquisition of word meanings. [12] addressed low literacy readers accessing Web pages and proposed a **web content adaptation tool**, named Educational Facilita. They used lexical elaboration and provided short definitions from Wikipedia to define named-entities which appear in the text besides highlighting these entities.

In this paper, we present a new technique for TE intended to enable detailed reading of a text and accurate information extraction. Our aim is to build an automatic question categorization system which assigns Wh-question labels to verbal arguments in a sentence. It will help users that can hardly comprehend a text, including children who are learning to read. Wh-question generation is like semantic annotation which involves the subtasks of making delimitation of verbs and arguments, and linking verbs to their arguments through question labels.

Recent work in Natural Language Processing has shown the benefit of using statistical language processing techniques for the task of semantic role labeling (SRL), which is strongly related to Wh-type question generation. In this paper, we present several machine learning experiments to build an automatic question categorization system which assigns Wh-question labels to verbal arguments in a sentence. Different question labels in English language will be used in this process.

III PROPOSED APPROACH

Approaches for generation of gap-fill and Wh-type question are different. The first stage is same that is Sentence Selection. The Sentence is selected based on number of features.

There are three stages for gap-fill question generation: **sentence selection, key selection and distractors selection.** Sentence Selection involves identifying informative sentences and question generable sentences in the document which can be used to generate a gap fill question. These sentences are then processed in the Key Selection stage to identify the key on which to ask the question. In the final stage, the distractors for the selected key are identified from the given chapter by searching for words with the same context as that of the key. In each stage, the system identifies a set of candidates i.e. all sentences in the document in stage I, words in the

previously selected sentence in stage II and words in the chapter in stage III and extracts a set of features relevant to the task. Data generated in one stage is used in the next stage.

Proposed System Design is as follow:

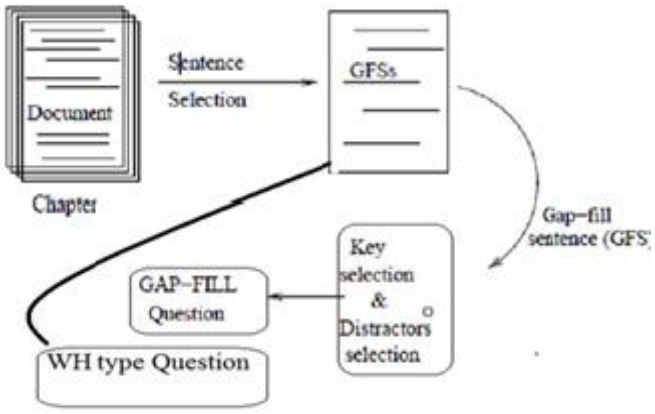

Fig 1 shows system design with all components and the flow of operation.

3.1 Sentence Selection

A good GFS should be informative and Gap-fill question-generable. An informative sentence in a document is one which has relevant knowledge that is useful in the context of the document? A sentence is gap-fill question-generable if there is sufficient context within the sentence to predict the key when it is blanked out. An informative sentence might not have enough contexts to generate a question from and vice versa. The sentence selection module goes through all the sentences in the documents and extracts a set of features from each of them. These features are defined in such a way that the two criterion defined above are accounted for.

Features are selected in such a way that the underlying algorithm can be modified to gain more accuracy and performance than previous one. Feature set for Sentence Selection helps to capture informative sentences, the potential candidate for generating a GFQ. The features are as follow:

1. **First sentence:** $f(s_i)$ is a binary feature to check whether the sentence s_i is the first sentence of the document or not.
2. **Common tokens:** $sim(s_i)$ is the count of words that the sentence and the title of the document have in common.
3. **Superlatives:** $super(s_i)$ features capture those sentences which contain words in superlative degree. The binary features determine the degree of the importance of a sentence in terms of the presence of superlatives.
4. **Sentence position:** $pos(s_i)$ is position of the sentence s_i. Sentences in the middle of the document are preferred since topic is elaborated in the middle of the document.
5. **Discourse connective at the beginning:** $discon(s_i)$'s value is 1 if first word of sentence si is a discourse connective and 0 otherwise. There are no enough contexts for QS, if the discourse connective is at the beginning of the sentence.

6. **Length:** $l(s_i)$ is the number of words in the sentence. A very short sentence might not have enough context for question and a very long sentence might have enough context to make the question generated from it is trivial. That's why this feature is considered.
7. **Number of nouns and pronouns:** Features nouns(s_i) and pronouns(s_i) represent the amount of context present in a sentence. More number of pronouns in a sentence reduces the contextual information; instead more number of nouns increases the number of potential keys to ask a gap-fill question on.

The Table1 gives summary of the features.

Table 1.Features for sentence

Feature Symbol	Description Criterion
$f(s_i)$	Is s_i the first sentence of the document?
$sim(s_i)$	No. of tokens common in s_i and title / length(s_i)
$super(s_i)$	Does(s_i) contain a word in its superlative degree?
$pos(s_i)$	(s_i)'s position in the document
$discon(s_i)$	Is (s_i) beginning with a discourse connective?
$l(s_i)$	Number of words in (s_i)
$nouns(s_i)$	No. of nouns in (s_i) / length(s_i)
$pronouns(s_i)$	No. of pronouns in (s_i) / length(s_i)

3.2 Key Selection

Key is selected from each sentence selected in the previous stage. Previous works in this area, [4] take keys as an input and, [14] and [15] select keys on the basis of term frequency and regular expressions on nouns. Then they search for sentences which contain that particular key in it. Since their approaches generate gap-fill questions only with one blank, they could end up with a trivial GFQ, especially in case of conjunctions.

That's why sentence selection is performed before key selection. Our system can generate GFQs with multiple blanks unlike previous works described above.

Key selection from a GFS takes place into two steps. In the first step the module generates a list of potential keys from the GFS (key-list) in the second step it selects the best key from this key-list.

3.2.1 Key-list formation

A list of potential keys is created using the part of speech (POS) tags of words and chunks of the sentence in the following manner:

1. Each sequence of words in all the noun chunks is pushed into key-list.
2. For each sequence in the key-list, the most important word(s) is selected as the potential key and the other words are removed. The most important word in a noun chunk in the context can be cardinal, adjective and noun is considered for the potential key. In case where there are multiple nouns, the first noun is chosen as the potential key. If the noun chunk is NP coordination, both the conjuncts are selected as a single potential key making it a case of multiple gaps in QS.

An automatic POS tagger and a noun chunker have been used to process the sentences selected in the first stage. It was observed that if words of a key are spread across a chunk then there might not be enough contexts left in QS to answer the question. The noun chunk boundaries ensure that the sequence of words in the potential keys is not disconnected.

3.2.2 Best Key selection

Following features are used to select the best key from the key-list.

1. **Term frequency:** term(key_p) is number of occurrences of the keyp in the document. term(key_p) is considered as a feature to give preference to the potential keys with high frequency.
2. **In title:** title(key_p) is a binary feature to check whether key_p is present in the title of the document or not. A common word of GFS and the title of the document serve as a better key for gap-fill question than the ones that are not present in both.
3. **Height:** height(key_p) denotes the height of the key_p in the syntactic tree of the sentence. Height gives an indirect indication of the importance of the word. It also denotes the amount of text in the sentence that modifies the word under consideration.

   An answerable question should have enough contexts left after the key blanked out. A word with greater height in dependency tree gets more score since there is enough context from its dependent words in the syntactic tree to predict the word.

   The score of each potential key is normalized by the number of words present in it and the best key is chosen based on the scores of potential keys in keylist. All the features are summarized in the Table 2 as below:

Table 2. Features for key selection

Feature Symbol	Description
term(key_p)	Number of occurrences of the key_p in the document
title(key_p)	Does title contain key_p ?
height(key_p)	height of the key_p in the syntactic tree of the sentence

3.3 Distractor Selection

Karamanis [13] defines a distractor as; an appropriate distractor is a concept semantically close to the key which, however, cannot serve as the right answers itself.

For distractor selection, [4] and [5] used WordNet, [16] used their in-house thesauri to retrieve similar or related words (synonyms, hypernyms, hyponyms, antonyms, etc.). However, their approaches can't be used for those domains which don't have ontology. Smith [4] does not select distractors based on the context of the keys. Distractor should come from the same context and domain, and should be relevant. Module uses three features, shown in Table 3, to select three distractors from the set of all as that of the key.

1. Contextual similarity: context($distractor_p$, key_s) gets the contextual similarity score of a potential distractor and the keys on the basis of context in which they occur in their respective sentences. Value of the feature depends on how similar are the key and the potential distractor contextually. The previous two and next two words along with their POS tags are compared to calculate the score.

2. Sentence Similarity: sim($distractor_p$, key_s) feature value represents similarity of the sentences in which the keys and the distractorp occur in. Dice Coefficient (Dice, 1945) has been used to assign weights to those potential distractors which come from sentences similar to GFS because a distractor coming from a similar sentence will be more relevant.

$$\text{dice coefficient}(s1, s2) = \frac{2 \times \text{commontokens}}{l(s1) + l(s2)} \quad \text{----------(1)}$$

3. Difference in term frequencies: diff ($distractor_p$, key_s) is used to find distractors with comparable importance to the key. Term frequency of a word represents its importance in the text and words with comparable importance might be close in their semantic meanings. So, a smaller difference in the term frequencies is preferable.

All the features are summarized in the table as below:

Table 3.Features foe distractor selection

Feature Symbol	Description
context($distractor_p$,key_s)	measure of contextual similarity of $distractor_p$ and the keys in which they are present
sim($distractor_p$, key_s)	Dice coefficient score between GFS and the sentence containing the $distractor_p$
diff($distractor_p$, key_s)	difference in term frequencies of $distractor_p$ and keys in the chapter

Key_s is the selected key for a GFS, $distractor_p$ is the potential distractor for the key_s

3.4 Assignment Wh-question Label and Feature Selection

Wh-question assignment is a type of semantic annotation that links verbs to their arguments through Wh-question labels such as who, what, which, when, where, why, how, how much, how many, how long, how often and what for. Consider this annotation for the sentence "**Naredra Modi went to Brazil last summer**." Question label assignment for this sentence will be as follow:

Who?" is the question label that links the verb "went" to the argument "Narendra Modi". Similarly, "Where? links the verb "went" to the argument "to Brazil" and "When?" links the same verb to the argument "last summer".

Commercial system is used to annotate actions and named-entities with Wh-questions to support text mining. But our task is different from the task performed by this system, because I link verbs to all their arguments even if they are not named-entities. Named entities and other arguments in the sentence are link to the verbs. I use the term "argument" here in the same way it is used in the Propbank project [16], i.e., on referring to both: arguments predicted by verb senses and adjuncts that modify verb senses adding information about circumstances of time (when), place (where), quantity (how much and how many), manner (how), purpose (what for), direction (in which direction) and cause (why).

The main task is to choose the question label that links properly the verb to each argument. Features are selected mostly those proposed by [17], with some adaptation. [18] present some features introduced in recent SRL systems, besides the core features used by [17]. The features are:

1. **Phrase type**: Different question types tend to be realized by different syntactic categories. In general, Noun Phrases (NP) answer the questions "What?" and "Who?" while Prepositional Phrases (PP) answer questions with prepositions, such as "for what?", "of what?", "to where?", "in what?", "with whom?",.
2. **Side** (or position): This indicates whether the constituent to be labeled occurs before (left) or after (right) the verb in focus. Therefore, there are two values: left and right for this feature.
3. **Argument order**: This feature is an integer indicating the position of a constituent in the sequence of arguments for a given verb.
4. **Sub categorization of syntactic functions**: This feature refers to the set of a verb´s syntactic argument in the sentence. Since the parser Palavras has a large set of syntactic labels, this feature can have 26 values as: direct object, indirect object, prepositional object, subject, predicator, utterance statement, subject complement, object complement.
5. **Specific syntactic function**: This feature presents a sub categorization of the feature (4). For example, we have two types of direct object (DO), two types of indirect objects, two types of verbs (main verb and auxiliary verb), This feature has 17 possible values.
6. **Question at the Left side?:** This Boolean feature allows the identification of sentences without subject or subjects at the right side of the verb.
7. **Number of arguments:** indicates the number of arguments of a sentence.
8. **Principal verb token**: an important lexical feature to determine the question type.
9. **First two Part Of Speech (POS) and Last POS of an argument**: These 3 features help to refine the type of NP involved, since the POS categories distinguish proper from common nouns and singular from plural nouns.
10. **First and Second tokens of an argument:** These features are used if the POS of the first and second tokens are from a closed class or open class, they receive.
11. **Simple or Multiword verb:** The number of tokens of a Verb.
12. **Number of tokens of the argument:** This feature is an integer indicating the number of tokens of the argument.

Above mention features are selected with the use of different algorithm as IBk, J48, JRip, SMO, and NaiveBayes, InfoGainAttributeEval. The Information Gain algorithm was chosen to rank the features because it is one of the most used methods.

SMO, SimpleLogistic (Maximum Entropy) and J48, K-NN. The worst results for JRIP and Naïve Bayes

IV CONCLUSION AND FUTURE WORK

AQG system will generate the GFQ and WH-type question as per the hypothesis. The system give sentences which are informative and question generable are used for generation of GFQ. Syntactic and lexical features

give quality questions from the document. The main focus will be on the generation of WH-type question from document. The hypothesis gives the result as it stated.

This will be useful for teachers as well as student in their learning process; low literacy readers; language skills development, people taking Distance Education. It doesn't use external resources for Distractors selection and finds them in the chapter only that makes it adaptable for any domain. It helps in improving the quality of the Question Answering (QA).Gap-fill questions can be used for Advanced Placement (AP) exams.

Extending experiments to the question types that have not yet assessed is an important next step and part of future work.

REFERENCES

[1] Burstein, J.: Opportunities for Natural Language Processing Research in Education. In the Proceedings of CICLing, 6--27 (2009)

[2] Urano, K.: Lexical simplification and elaboration: Sentence comprehension and incidental vocabulary acquisition.

[3] Siddharthan, A.: Syntactic Simplification and text Cohesion. PhD thesis, University of Cambridge(2003)

[4] Simon Smith, P.V.S Avinesh and Adam Kilgarriff. 2010. Gap-fill Tests for Language Learners: Corpus-Driven Item Generation.

[5] Jonathan C. Brown, Gwen A. Frishkoff, Maxine Eskenazi. 2005: Automatic Question Generation for Vocabulary Assessment, Proc. of HLT/EMNLP '05, pp. 819-826.

[6] Wolfe, J.H.: Automatic question generation from text - an aid to independent study. SIGCUE Outlook 10(SI) (1976)

[7] Eiichiro Sumita, Fumiaki Sugaya, and Seiichi Yamamoto 2005:Measuring Non-native Speakers Proficiency of English by Using a Test with Automatically-Generated Fill-in-the-Blank Questions, 2ndWkshop on Building Educational Applications using NLP, Ann Arbor

[8] John Lee and Stephanie Seneff. 2007: Automatic Generation of Cloze Items for Prepositions, CiteSeerX - Scientific Literature Digital Library and Search Engine .

[9] Lin, Y. C., Sung, L. C., Chen and M. C. 2007: An Automatic Multiple-Choice Question Generation Scheme for English Adjective Understanding, CCE 2007 Workshop Proc. of Modeling, Management and Generation of Problems / Questions in eLearning, pp. 137-142.

[10] Juan Pino, Michael Heilman and Maxine Eskenazi. 2009: A Selection Strategy to Improve Cloze Question Quality, Wkshop on Intelligent Tutoring Systems for Ill-Defined Domains. 9th Int. Conf. on ITS.

[11] Burstein, J., Shore, J., Sabatini, J., Lee, Y.W., Ventura, M.: The automated text adaptation tool. In NAACL '07: Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations on XX, 3--4 (2007)

[12] Watanabe, W.M., Candido Jr, A., Amancio, M.A., Oliveira, M., Pardo, T.A.S., Fortes, R.P.M., Aluísio, S.M.: Adapting web content for low literacy readers by using lexical elaboration and named entities

labeling. In the Proceedings of the W4A-7th International Cross-Disciplinary Conference on Web Accessibility 2010, (2010).Nova York: ACM Press, v. 1, 1--9 (2010)

[13] Karamanis, Le An Ha and Ruslan Mitkov.2006 Generating Multiple-Choice Test Iterms from Medical Text: A Pilot Study, In Proceedings of INLG 2006, Sydney, Australia.

[14] Ruslan Mitkov, Le An Ha and Nikiforos Karamanis. 2006 A computer-aided environment for generating multiple-choice test items, Natural Language Engineering 12(2): 177-194

[15] Hidenobu Kunichika, Minoru Urushima,Tsukasa Hirashimaand Akira Takeuchi. 2002. A ComputationalMethod of Complexity of Questions on Contents of English Sentences and its Evaluation, In: Proc. ofICCE 2002, Auckland, NZ, pp. 97101 (2002).

[16] Palmer, M., Gildea, D., Kingsbury, P.: The Proposition Bank: A Corpus Annotated with Semantic Roles, Computational Linguistics journal, 1:1, 71--106 (2005)

[17] Gildea. D., Jurafsky, D.: Automatic Labeling of Semantic Roles. Computational Linguistics Volume 28, Number 3, 1--45 (2002)

[18] Palmer, M., Gildea, D., Xue, N.: Semantic Role Labeling. Synthesis Lectures on Human Language Technology Series, ed. Graeme Hirst, Mogan & Claypoole (2010).