RevUP: Automatic Gap-Fill Question Generation from Educational Texts

Girish Kumar
NUS High School of Math and Science
Singapore 129957
girishvill@gmail.com

Rafael E. Banchs, Luis F. D'Haro
Institute for Infocomm Research
Singapore 138632
{rembanchs,luisdhe}@i2r.a-star.edu.sg

Abstract

This paper describes RevUP which deals with automatically generating gap-fill questions. RevUP consists of 3 parts: Sentence Selection, Gap Selection & Multiple Choice Distractor Selection. To select topically-important sentences from texts, we propose a novel sentence ranking method based on topic distributions obtained from topic models. To select gap-phrases from each selected sentence, we collected human annotations, using the Amazon Mechanical Turk, on the relative relevance of candidate gaps. This data is used to train a discriminative classifier to predict the relevance of gaps, achieving an accuracy of 81.0%. Finally, we propose a novel method to choose distractors that are semantically similar to the gap-phrase and have contextual fit to the gap-fill question. By crowdsourcing the evaluation of our method through the Amazon Mechanical Turk, we found that 94% of the distractors selected were good. RevUP fills the semantic gap left open by previous work in this area, and represents a significant step towards automatically generating quality tests for teachers and self-motivated learners.

1 Introduction

In today's educational systems, a student needs to recall and apply major concepts from study material to perform competently in assessments. Crucial to this is practice and self-assessment through questions. King [1992] found that questioning is an effective method of helping students learn better. However, the continued crafting of varied questions is extremely time consuming for teachers as mentioned in Mitkov et al. [2006]. Furthermore, learners are increasingly moving from the traditional classroom setting to an independent learning setting online. Here, there is a need for leveraging upon online educational texts to provide practice material for students. Automatic Question Generation (AQG) shows promise for both these use-cases.

1.1 Related Work

Work in Automatic Question Generation(AQG) has mostly involved transforming sentences into questions and can be divided into two categories: Wh-Question Generation (WQG) and Gap-Fill Question Generation (GFQG). Most work in WQG has involved transforming sentences into grammatically correct Wh-questions (Why, What, How, etc.) with little attention given to the semantics and educational relevance of the questions (Heilman and Smith [2009], Mitkov et al. [2006], Mostow and Chen [2009], Wolfe et al. [1975], Wyse and Piwek [2009]). On the other hand, previous works in GFQG have generally worked with vocabulary-testing and language learning (Smith et al. [2010], Sumita et al. [2005]). Smith et al. presented TedClogg which took gap-phrases as input and found multiple choice distractors from a distributional thesaurus. 53.3% of the questions generated were acceptable. Our work aligns more closely to that of Aggarwal et al. where a weighted sum of lexical, syntactic features were utilised to select sentences, gaps and distractors from informative texts (Agar-

Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications, 2015, pages 154–161,
Denver, Colorado, June 4, 2015.

wal and Mannem [2011]). Becker et al. [2012] built upon the former's work by collecting human ratings of questions generated from a Wikipedia-based corpus. A machine-learning model was trained to effectively replicate these judgments, achieving a true positive rate of 83% and false positive rate of 19%.

RevUP focuses on GFQG which overcomes WQG's need for grammaticality by blanking out meaningful words (gaps) in known good sentences.

1.2 Key Contributions

Our key contribution is the employment of data-driven but domain independent methods to construct RevUP: an automated system for GFQG from educational texts. RevUP consists of 3 components: Sentence Selection, Gap Selection & Distractor Selection.

Sentence Selection

Current systems use extractive summarization methods which may not suitable as they aim to choose sentences that cover the most content, which could result in complexity or incoherence. As such, we propose selecting *topically important* sentences by ranking them based on topic distributions obtained from a topic model.

Gap Selection

Here, we train a machine learning classifier to replicate human judgements on the relevance of gaps. We propose collecting human rankings of the educational relevance of gaps. This is because ratings of gaps on a points scale resulted in inter-rater agreement issues in past work as each annotator had different thresholds for each point. We then propose semantic and domain-independent features for classifier training on these rankings and the trained classifier predicts the educational relevance of gap candidates with an accuracy of 81.0%.

Distractor Selection

Contrary to previous work which use thesauruses or syntactic features, we propose using vector representation of words (word2vec), language model probabilities and dice coefficients to find semantically similar distractors with contextual fit to the question. 94% of the distractors selected by RevUP were found to be good.

A Biology text book titled *Campbell Biology, 9th Edition* has been used for work throughout this paper. The textbook consists of 35621 sentences, with each sentence consisting of an average of 20 words.

2 Sentence Selection

Previous work in AQG used extractive summarisation for selecting sentences Becker et al. [2012]. Since these methods aim to select sentences that maximise content coverage, they might not be suitable as such sentences can be complex and incoherent. As such, we aim to choose topically-important sentences that have a peaked topic distribution and $w = [0.5, 0.3, 0.2]$. Sentences with the top-n scores are selected. This is because sentence with peaked distributions have the following two properties.

1. The sentence belongs only to a few topics
2. These topics are expressed to a high degree

The first property implies that the sentence is coherent in terms of the ideas and content it expresses.The second property implies that the sentence contains important and interesting information. Each sentence is assigned a score as follows.

$$\text{score} = \sum_{i=1}^{k} w_i \cdot \max(\mathbf{t}, i) \tag{1}$$

where $\max(\mathbf{t}, i)$ is the i^{th} largest probability in topic distribution $\mathbf{t}$ obtained from a topic model and w_i is its associated weight. For RevUP, we set $k = 3$. Table 1 shows a list of good and bad sentences with their scores.

It is to be noted that the assumption that topically important and coherent sentences make good questions does not always hold. We leave it to future work to account for more factors.

3 Gap Selection

We over-generated a list of candidate gap-phrases from every sentence and trained a binary classifier on human judgements of the relative relevance of the gap-words. Though similar to Becker et al., we

Good Sentences	Score	Bad Sentences	Score
Within the cortex, sensory areas receive and process sensory information, association areas integrate the information, and motor areas transmit instructions to other parts of the body.	0.48	As the water warms or cools, so does the body of the bass.	0.14
Roots were another key trait, anchoring the plant to the ground and providing additional structural support for plants that grew tall.	0.41	The scientific community reflects the cultural standards and behaviors of society at large.	0.14
Each nucleotide added to a growing DNA strand comes from a nucleoside triphosphate, which is a nucleoside with three phosphate groups.	0.29	In one study, researchers spread low concentrations of dissolved iron over 72 km 2 of ocean and * C uptake by cultures measures primary production.	0.16

Table 1: Good and bad sentences according to proposed sentence ranking metric

propose ranking gap-phrases instead of rating them to improve inter-rater agreement. Furthermore, we propose semantic features for classifier training. We used sentences from the Campbell Biology Textbook.

3.1 Methodology

3.1.1 Candidate Extraction

We extracted candidate gap-phrases that span up to three words. To prevent a skew towards irrelevant gap-phrases, we employed domain-independent syntactic rules. We first ran the Stanford Part-of-Speech (POS) Tagger to obtain the POS tags for each word in the sentence and the Stanford Parser to obtain a syntactic parse tree (Toutanova et al. [2003a,b]). We extracted all the nouns, adjectives, cardinals and noun-phrases with a Wikipedia page.

3.1.2 Crowd-Sourcing Annotations

Pinpointing a relevant gap is a complex task which relies on human judgement. Amazon Mechanical Turk, MTurk, was used to collect such human annotations in a cost and time efficient manner. In MTurk, requesters can pay human workers (Turkers) a nominal fee to complete Human Intelligence Tasks (HITs). To gather quality annotations, a HIT must be easy to complete and must take into account limitations with human judgement. We first piloted a HIT where a turker was tasked to rate gap-phrases from a source sentence on a scale from 1 to 5. However, we found very poor inter-annotator agreement as the task was tedious (up to 10 candidate gap-phrases per task) and each annotator had different thresholds for each point on the scale. However, the ratings preserved the relative educational relevance of the gaps. As such, we decided to redesign the HIT as a ranking task. Also, for shortening purposes, each HIT involved the ranking of 3 gap-phrases from one source-sentence. As such, for every source sentence,we created multiple sets of gap-phrase triplets as in Figure 1.

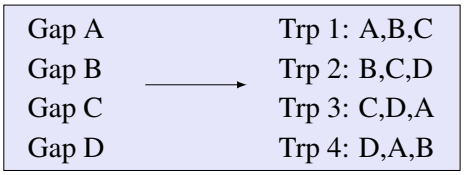

Figure 1: Triplet Generation. Trp refers to Triplet.

Each gap-phrase is part of three ranking HITs and each triplet shares two gap phrase pairs with two other triplets. In Figure 1,Trp 1 shares A,B with Trp 4 and B,C with Trp 2. Since conventional inter-annotator agreement metrics, e.g. Cohen's Kappa, cannot be used for a ranking task, we proposed an inter-ranker agreement measure as in Equation 2.

$$\text{Agreement} = \frac{\sum_{X,Y \in Gap-Pairs} \begin{cases} 1, & \text{if } \operatorname{sgn}(r_1(X) - r_1(Y)) = \operatorname{sgn}(r_2(X) - r_2(Y)) \\ 0, & \text{otherwise} \end{cases}}{\text{Num. of HITs}} \quad (2)$$

Sentence	Selected Gap
Sister chromatids are attached along their lengths by protein complexes called _____.	cohesins
Using an ATP-driven pump, the _____ expel hydrogen ions into the lumen	parietal cells
Unlike _____, leukocytes are also found outside the circulatory system, patrolling both interstitial fluid and the lymphatic system.	erythrocytes
A shoot apical meristem is a _____ mass of dividing cells at the shoot tip.	dome-shaped

Table 2: Gaps selected by RevUP. Red indicates bad gaps.

where $\mathrm{sgn}(\cdot)$ is the sign function and $r_n(Z)$ is the rank assigned by ranker n to gap Z.

To collect sentences for HIT deployment, we first ranked all the sentences from the Campbell's Biology textbook as in Section 2.2. From the top sentences, we hand-picked sentences to ensure a good mix of topics, sentence-lengths and gap-phrase lengths so as not to introduce a bias. 200 sentences were deployed with rankings collected for 1306 gaps in total. The inter-ranker agreement was high at 0.783.

3.1.3 Automatic Gap Classification

Since every gap was ranked thrice, we assigned each gap a score by summing up the three ranks. Ranks ranged from 1 to 3: 1 for best and 3 for worst. Scores ranged from 3 to 9. For binary classification, gap-phrases with scores less than 6 were considered good and the rest bad. Data filtering was done by removing gap-phrases that had been ranked first, second and third due to the uncertainty associated with the relevance of the gap. Gap-phrases that were part of triplets that showed no agreement with both the triplets that they shared gap-phrase pairs with, were removed. 285 gaps were removed. Our final dataset had a slight skew towards bad gaps with 554 bad gaps and 468 good gaps.

A good set of features are vital for training a good classifier. Table 4 lists all the features used for classification. Note that all the features are domain-independent. Using the scikit-learn python package, we trained a Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel (Pedregosa et al. [2011]).

3.2 Results

Table 3 details the average accuracy, precision, recall and F1 score achieved for a 10-fold cross validation test. Given an accuracy of 81%, we can conclude that RevUP performs fairly well for gap selection, on par with Becker et al. [2012].

	Filtered Gaps	All Gaps
Accuracy	0.81 ± 0.024	0.77 ± 0.026
Precision	0.81 ± 0.061	0.74 ± 0.045
Recall	0.77 ± 0.066	0.71 ± 0.082
F1-Score	0.79 ± 0.032	0.72 ± 0.043

Table 3: SVM Cross-Validation Results.

Besides, the results prove the huge impact preprocessing had on classifier performance. To understand impact of each feature on classifier performance, we obtained the classifier accuracy without each feature over 10-folds (Figure 2).

We can observe that most features have an equal effect on classifier performance with the exception of WordVec (Feature 10). Without WordVec, classifier performance drops to 76.6%. The large impact of WordVec is mainly because it strongly encodes the semantics of candidate gap-phrases. Word2Vec employs a Skip-gram model to learn and obtain distributed representations of words, from input texts, in a vector space which spatially encodes the semantic information and meaning of words. We believe that interesting and important words are separated from unimportant words in this vector space. This could have also helped in improving classifier accuracy.

Examples of gaps selected by RevUP are in Table 2.

4 Distractor Selection

The final component of RevUP pipeline involves the selection of relevant multiple-choice distractors to ensure that the learner has a good grasp of the relevant concepts put to test. Past work has involved the usage of thesauruses, LSA and rule-based approaches. Contrary to this, we propose a domain-

No.	Name	Description
0	Char Length	Number of characters in gap-phrase
1	Char Overlap	Character length of gap divided by character length of sentence
2	Height	Height of the gap-phrase in the syntactic parse tree
3	TF	Number of times gap-phrase occurs in the source sentence
4*	Corpus TF	Number of times gap-phrase occurs in the biology textbook
5*	Corpus IDF	Inverse document frequency of the gap-phrase in the biology textbook. Sentences are treated as documents.
6*	Sent. Words	Number of words in the source sentence
7*	Word Overlap	No. of Words in the gap-phrase divided by Sent. Words
8	Index	Position of the gap-phrase in the source sentence
9*	WN Synsets	Number of WordNet synsets of the gap-phrase
10*	WordVec	Vector of the gap-phrase as computed with the Word2Vec Tool. Refer to Section 4.1 for more details on word2vec.
11	Prev. POS Tag	Part-of-Speech Tags of the two words before the gap-phrase
12	Post. POS Tag	Part-of-Speech Tags of the two words after the gap-phrase
13	NER Tag	Name-Entity Tag of the gap-phrase
14	SRL	Semantic Role Label of the gap-phrase
15*	Topic Distribution	Topic Distribution of the gap phrase as computed by the proposed deep learning model
16*	Topic Distribution Change	Jensen Shannon Divergence between topic distribution of the gap phrase and the source sentence
17*	Transition Prob.	Transition probability from Kneser Ney Back-off Language Model trained on the biology textbook corpus

Table 4: Features Used to Train Binary Classifier. * represents features proposed by the authors. The rest correspond to that by Becker et al. [2012]

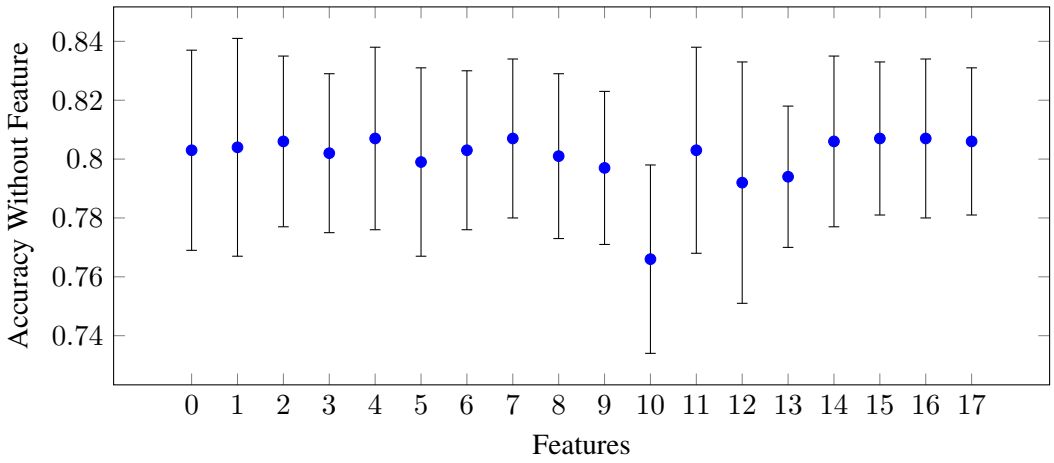

Figure 2: The effect of features on classifier performance. Note that Feature Number Correspond to Table 4

independent, data-driven approach to select distractors with semantic similarity and contextual fit. We leave it to future work to reject distractors that are correct answers to their respective questions.

Sentence	Selected Gap	Distractor
Sister chromatids are attached along their lengths by protein complexes called _____.	cohesins	1) spindle microtubules 2) myosin filaments 3) thick filaments 4) kinetochores
_____ worsen pain by increasing nociceptor sensitivity to noxious stimuli.	Prostaglandins	1) nitric oxides 2) steroid hormones 3) signaling molecules 4) lipid-soluble hormones
Instead, a hypha grows into a tube formed by _____ of the root cells membrane.	invagination	1) vegetal pole 2) undifferentiated cell 3) neural plate 4) frog embryo
_____ bodies are reinforced by ossicles, hard plates composed of magnesium carbonate and calcium carbonate crystals.	Echinoderms	1) sense organs 2) salamanders 3) birds 4) turtles

Table 5: Examples of distractors generated by RevUP. Red indicates bad distractors.

4.1 Methodology

To choose distractors semantically similar to the gap-phrase, we used the word2vec tool (Mikolov et al. [2013]). However, word2vec requires input texts with millions of words to learn quality vector representations. To rapidly expand our biology training dataset, we downloaded and processed the latest dumps of Wikipedia. Thereafter, to ensure that we only obtained texts relevant to the textbook used, we implemented a TF-IDF search engine through the gensim python package (Řehůřek and Sojka [2010]). The Campbell's Biology textbook was split into 548 batches of 50 sentences each and texts from the top 50 Wikipedia pages for each batch were used. The final data-set consisted of 900,000 sentences and 21 million words. This data-augmentation method keeps our proposed solution domain-independent as only the relevant textbook is needed. For word2vec training, the dimension of the vector space was set to be 70. A n-best list of candidate distractors can be chosen by ranking words in the vocabulary according to the cosine similarity of their vectors with respect to that of the gap-phrase. Thereafter, we removed candidates that already appear in the question sentence and that are of different parts-of-speech. Finally, we validated the semantic similarity of each candidate with the gap-phrase with WordNet (Miller [1995]). WordNet is a lexical graph database where words are grouped into sets of synonyms (synsets). Synsets are linked through a number of relations. We measured the semantic similarity of two terms, x, y, using path similarity.

$$\text{pathsim}(x, y) = \frac{1}{1 + \text{len(shortest\_path(x,y))}} \quad (3)$$

where $len(shortest\_path(x, y))$ is the shortest path between words x and y in WordNet. We eliminated candidates with path_sim < 0.1. We then proceeded to re-rank the candidates to obtain the 4 best distractors. Often, syntactic similarities between distractors and their respective gap-phrases help confuse students. For example, *s-phase* is a good distractor for *g-phase*. We captured such syntactic similarities by computing the Dice Coefficient, DC, for the gap-phrase and each candidate (Equation 4).

$$DC(X, Y) = \frac{2 \cdot |X \cap Y|}{|X| + |Y|} \quad (4)$$

To take into account the context of the question, we calculated the language model probability of the

candidate given the words that appear before the gap-phrase in the question-sentence. We trained a 5-gram Kneser Ney Back-off Language Model with the data used for word2vec training. Finally, we re-weighted and ranked the candidates according to their word2vec similarity, dice coefficient and language model probabilities and we picked the top 4 candidates as the final distractors.

4.2 Results

Amazon Mechanical Turk was used to evaluate our distractor selection method. Turkers were presented with a Gap-Fill Question, gap-phrase and were tasked to evaluate whether each of the top 4 distractors were good or bad. 75 sentences with 300 distractors from the Campbell's Biology Textbook were deployed. Since every distractor was rated by 5 turkers, we assigned each distractor a score by summing up the five ratings (1 for Good and 0 for Bad). Scores ranged from 0 to 5. Results are summarized in Table 7.

	Very Good	**Fair**	**Bad**
Percentage of Distractors	43%	51%	6%

Table 6: Distractor Selection Results

Mean	**Variance**
3.19	1.51

Table 7: Distractor Rating Statistics

Distractors with a score $>$ 3 were considered very good, score $<$ 2 were considered bad and the rest fair. We found that 51% of the distractors had a score of 2 or 3 which meant that there was low inter-annotator agreement. This reflects the complexity of the task as well as a lack of biological . As such, a more precise evaluation of our system can be performed with students/teachers as our annotators instead. Nonetheless, with 94% of the distractors being at least fair, RevUP's distractor selection component works fairly well.

Examples of distractors selected by RevUP are in Table 5.

5 Conclusion & Future Work

In summary, we have leveraged upon data-driven machine learning methods to propose RevUP: an automated, domain-independent pipeline for GFQG. Leveraging on topic models, a new topic-distribution based ranking method was proposed for sentence selection. For gap-selection, a discriminative binary classifier was trained on human annotations. With the classifier, RevUP could predict the relevance of a gap-phrase with an accuracy of 81.0%. We finally proposed a novel method for generating semantically-similar distractors with contextual fit and demonstrated that a 94% of the generated distractors were fair.

For future work, we hope to utilise more parameters to more accurately pinpoint better sentences. As for gap selection, we could explore the usage of more features and the usage of learning-to-rank methods e.g. SVMRank. We intend to cast the distractor selection problem as a machine learning problem to be trained from human judgments. Another possibility is the integration of RevUP into e-learning platforms such as Moodle to allow public usage of the tool. This could pave the way for usability tests to be conducted to understand the impact RevUP has on the learning process and educational performance of students. Furthermore, RevUP could be used to generate questions from transcribed lectures on MOOC platforms such as Coursera and Udacity.

References

Manish Agarwal and Prashanth Mannem. Automatic gap-fill question generation from text books. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 56–64. Association for Computational Linguistics, 2011.

Lee Becker, Sumit Basu, and Lucy Vanderwende. Mind the gap: Learning to choose gaps for question generation. In *HLT-NAACL*, pages 742–751. The Association for Computational Linguistics, 2012. ISBN 978-1-937284-20-6.

Michael Heilman and Noah A Smith. Question generation via overgenerating transformations and

ranking. Technical report, DTIC Document, 2009.

Alison King. Comparison of self-questioning, summarizing, and notetaking-review as strategies for learning from lectures. *American Educational Research Journal*, 29(2):303–323, 1992.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

George A. Miller. Wordnet: A lexical database for english. *COMMUNICATIONS OF THE ACM*, 38: 39–41, 1995.

Ruslan Mitkov, Le An Ha, and Nikiforos Karamanis. A computer-aided environment for generating multiple-choice test items. *Nat. Lang. Eng.*, 12(2):177–194, June 2006. ISSN 1351-3249. doi: 10.1017/S1351324906004177.

Jack Mostow and Wei Chen. Generating instruction automatically for the reading strategy of self-questioning. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, pages 465–472, Amsterdam, The Netherlands, The Netherlands, 2009. IOS Press. ISBN 978-1-60750-028-5.

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

Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA.

Adam Kilgarriff Simon Smith, PVS Avinesh, and Adam Kilgarriff. Gap-fill tests for language learners: Corpus-driven item generation. In *Proceedings of ICON-2010: 8th International Conference on Natural Language Processing*, 2010.

Eiichiro Sumita, Fumiaki Sugaya, and Seiichi Yamamoto. Measuring non-native speakers' proficiency of english by using a test with automatically-generated fill-in-the-blank questions. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, pages 61–68. Association for Computational Linguistics, 2005.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003a.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003b.

John H. Wolfe, Navy Personnel Research, and CA. Development Center, San Diego. *An Aid to Independent Study through Automatic Question Generation (AUTOQUEST) [microform] / John H. Wolfe*. Distributed by ERIC Clearinghouse [Washington, D.C.], 1975.

Brendan Wyse and Paul Piwek. Generating questions from openlearn study units. 2009.