Automatic generation of Multiple Alternate Choice (MAC) test item stems by applying Causal Relation Explication, Addition and Manipulation (CREAM) to pre-processed source documents

Robert Michael Foster
University of Wolverhampton,
Wulfruna Street, Wolverhampton, WV1 1LY,
Research Institute in Information and Language Processing
R.M.Foster@wlv.ac.uk

Abstract

A new method for generating the stem components of Multiple Alternate Choice (MAC) test items is proposed which makes use of a theory about causal coherence relation primitives. A trial of the Causal Relation Explication, Addition and Manipulation (CREAM) method is described which uses a domain specific evaluation method. The domain expert assigns a usability category to each MAC stem from a bank consisting of a total of 40 relevant MAC stems. 20 of the stems contained in the item bank were generated using the CREAM methodology while the other 20 MAC stems were created using a traditional approach. The results show 5 generated stems were assigned category A (use without changes) and a further 9 were assigned category B (use following minor changes). Further performance improvements are expected in future experiments by applying new pre-processing techniques to the source documents.

1. Introduction

Multiple Choice Question test items (MCQ) are used by the UK Company featured in this study to regularly confirm staff knowledge of documents from the company's Policy Library. The MCQ test items are delivered in the form of pre and post tests associated with training courses and field audits. The stored responses from these tests allow the company to demonstrate that training has been received by staff in accordance with requirements stated in UK Legislation. However an internal study proved that creating and updating the item bank manually is an expensive process. In response to these results we are investigating various ways to automatically generate MCQ test items.

Several systems that suggest methods for automatic generation of MCQs from source documents have recently been proposed in the literature [1]. [2], [3], [4], [5], [6]. This paper describes and tests a new approach to this task entitled Causal Relation Explication, Addition and Manipulation (CREAM).

The CREAM method involves working with target clauses and sentences from source documents and could either be incorporated into existing systems or be used as the sole method for generating test item stems. Thanks to the binary nature of the causal coherence relation primitives [7] that form the basis for CREAM, the proposed approach is particularly relevant to the task of generating component stems for the more specific Multiple Alternative Choice test item (MAC) [8] format. The use of MACs is supported in many textbooks that include item-writing guidelines [9] and a recent study into workplace assessment [10] provides further evidence in support of MACs.

In order to examine the possible benefits in the featured domain that arise from applying CREAM during MCQ test item creation, analysis of selections made by a domain expert from a bank of MCQ stems is carried out. The relative proportion of CREAM-generated test item stems to manually created MCQ test item stems in the selections made by the domain expert is used as an evaluative measure of the CREAM method.

The rest of this paper is organized as follows: section 2 describes the motivation for the study and provides a description of the methodology. Section 3 describes the experiment, including an example while sections 4 and 5 give the results and conclusions.

2. Context

2.1 Motivation

The Policy Library for the featured UK Company consists of a small number of general policy statements (POLs) and a large number of Standard Techniques (STs). The STs are intended to give precise instructions for the correct methods that the

staff must apply when they are carrying out work on behalf of the company. Several of the Standard Techniques contain requirements for staff to complete sequences of MCQ test items (called 'CBT tests'). For example:
ST:OS7D – Relating to Audits of Operational field staff

> *"3.1 All Senior Authorised and Authorised Persons who hold an authorisation for HV Operational Work (11SW, 33SW, 66SW, 132SW and restricted variations) shall complete an annual CBT test to the satisfaction of an Examining Officer qualified to examine for that authorisation."*

In 2007 a review was carried out of the costs of producing and maintaining an item bank of 130 MCQ test items that were first created in 1991. The study demonstrated that item production and maintenance is particularly time consuming. A follow up research project has therefore been set up to analyse and improve the process for creating and maintaining the MCQ assessment tests used by this company.

2.2 A Review of Existing Systems

Many of the MCQ test item generation systems recently presented in the literature focus upon language familiarization or vocabulary assessment. These assessment objectives are too narrow for the systems that serve them to be useful in the featured domain. For example, Brown's study [2] provides a system that is specifically targeted towards testing of vocabulary knowledge and although there might be a place in our item banks for a few items that test vocabulary within our corporate sub-language, particularly in the light of results from one of our experiments [6], our requirement is for a more flexible system that can produce test items that address a range of learning objectives.

Any system that we reviewed that required the construction (or pre-existence) of some form of knowledge base, eg [3] was discounted since the construction costs for such a system would be very hard to justify for our domain in the current economic climate. The maintenance burden of a knowledge base is also likely to present a barrier.

The most promising approach identified so far has been the use of a MCQ test item generator [4], [5] to generate MCQ test items and post edit them to form the item bank. It has been reported in [4] and [5] that generating MCQ items using the generator can speed up the process by 4 times without compromising the quality of the output. However preliminary experiments applying the system [4], [5] to the policy library from the featured company delivered no usable MCQ test items so significant improvements in performance are necessary before the system could be adopted. The MCQ test item generator [4].[5] uses the following steps to generate MCQ test items:

1) Identify significant terms within the source
2) Apply a clause filtering module
3) Transform the filtered clauses into questions
4) Use semantic similarity to select distractors.

During initial experiments with a particular policy document from our policy library, most of its clauses were filtered out and so the number of usable MCQ test items produced was very small. In order to improve upon this performance, a new source document pre-processing technique [1] was applied to source documents which sought to improve the output from step 2 of the above process. The results gave some improvement in the output and so further pre-processing techniques have been investigated and published [6] to improve performance in step 1.

The question generation patterns applied to source texts during step 3 in the original [4], [5] methodology are applicable to many educational contexts. However in an industrial training context a greater emphasis needs to be placed upon precise procedural knowledge and memory of specific measurements. This can be seen from a comparison of MCQ test items that have been created manually by industrial trainers in the featured company and MCQ test items that were generated by the system during initial experiments:

Source Sentence: *"Make sure you complete all sections of the diary page.In the 'Work Carried Out' section you must give comprehensive details of your day's achievements."*

Manually created question: *"A brief description is all that is required in the Work Carried out' section - True or False?"* (Correct: False)

Generated question: *"What kind of details of your day's achievements must you give in the 'Work Carried out' section"* (Correct: Comprehensive)

The process featured in this paper makes an attempt to improve the quality of output from steps 3 and 4 of the above process by applying another theory from the literature which concerns Causal Coherence Relations [7].

2.3 The New Approach – CREAM

A useful method for causal coherence relations analysis is proposed in the literature which requires the assumption that all relations are cognitively basic [7]. The proposal is that we only need four cognitive primitives to express the primitive causal coherence relations necessary for communication. Combination of these four primitives by a writer can then present more sophisticated types of causal coherence relation between the information units within a text. The primitives are described in detail in the literature [7], but can be summarized as follows:

1) Basic operation (causal vs additive)
2) Source of coherence. (semantic vs pragmatic)
3) Order of information units (basic vs complex).
4) Polarity (positive vs negative)

The utility of this theory in the context of MCQ test item stem creation is that a standard can be applied to policy documents insisting that source texts clearly define causal coherence relation clauses. Once a trigger phrase has been identified that indicates a likely causal coherence relation linking one or more significant information units then a NLP system can apply pre-prepared patterns to produce predictable effects, including question stems.

The first step in applying CREAM is to identify all information units within the source document that are significant to the person who has requested MCQ test item generation and then to ensure that all Causal Relations between these significant information units are *Explicit (Explication)*. If the original form of the source document contains *implicit* causal coherence relations between significant information units then explicit statements of the causal coherence relations must be inserted (Addition).

The second step in applying CREAM is to Manipulate each of the primitives individually. The most obvious manipulation that might be tried is reversal. Reversal would produce four stems that are likely to be regarded as 'opposites' to the original source sentence. If further stems are required (for example if the generated stems are likely to be too obviously false) then a combination of two of the primitives can be applied consecutively.

Many different combinations are possible because not only the combination but also the sequence of application of two primitives can significantly alter the meaning of a generated stem. It has been noted that the current generator [4], [5] sometimes generates unwieldy, confusing question stems [1]. A similar effect is observed when more than two primitives are combined in the second step in the CREAM method. However, the expectation is that CREAM will generate a sufficient number of acceptable stems to satisfy most assessment contexts.

3. Experiment

3.1 Hypothesis

The hypothesis is that MCQ test item stems that have been generated using the CREAM methodology are indistinguishable from MCQ test item stems that have been created manually. This will be considered to have been proved if an equal or greater number of CREAM generated stems appear in the top 20 stems selected by the domain expert on the basis of usability score allocations.

3.2 Methodology

CREAM is a new application of causal coherence relations [7]. The application of the theory is achieved within a simulation as opposed to a reprogramming of the question generator in order to ensure careful and thorough application of the methodology. The source documents used in the experiment are taken from the policy library of the UK Company that was referred to in the introduction.

Application to the sentence from section 2.2 is presented below in order to illustrate the process that produced the generated stems used in the experiment. Step 1 of the CREAM method involves pre-processing the source documents in order to establish whether the causal coherence relations that link significant information units are Explicit, and if they are not then they need to be Added. Applying Explication and Addition processes to the example quoted in section 2.2 yields the following Explicit Causal coherence relations (**bold** indicates additions)

You must complete all sections of the diary page.
- Risk Assessment Box is not optional
- Location Box is not optional
Etc.
A brief description of the job is not sufficient for the 'Work Carried out' section, *you must give comprehensive details of your day's achievements in the 'work carried out' section.*

Similar pre-processing was applied to all identified source texts. The output sentences from the pre-processing were used as source documents during the manual simulation of the test item generator [4], [5] processes which included application of step 2 of the CREAM methodology. Example MAC test item stem generated are as follows:

(i) No manipulation: *"You must complete all sections of the diary page.* (Correct Response: True)

(ii) Primitive 4 Manipulated: *"Risk Assessment Box is optional"* (Correct Response: False)

(iii) Primitive 1 & 4 Manipulated: *"A brief description of the job is sufficient for the 'Work Carried out' section"* (Correct Response: False)

The simulated run of the MCQ test item generator generated 20 such candidate MAC test item stems. These 20 new MAC stems were added to the 20 manually created MAC test items stems covering the same source documents to form the bank of 40 MAC item stems.

3.3 Evaluation

The final selection by the domain expert was to consist of at least 20 item stems with the aim of confirming apprentices' ability to recognise and recall facts following their attendance at his training sessions. He had no involvement in the creation of either the manually or automatically generated items and had no prior knowledge of which were generated MAC item stems, therefore these factors could not have any bearing upon his decision about which items to include in his MCQ test item routine. The

following usability scores were used to record the domain expert's assessments of the items:

A= Use the item stem unchanged

B= Make minor changes and then use the item

C= Do not use the item

4. Results and Discussion

On the day of the experiment the full set of 40 MAC test item stems was presented to the domain expert. In the case of the MAC test item stem examples described in section 3, (i) was placed in category B and (ii) and (iii) were placed in category A. Once usability categories had been assigned for each of the 40 items in the item bank, the following comparison table was produced:

Table 1 – Usability categorization decisions for CREAM-generated vs Manually created MAC stems.

Usability Score categories	***CREAM-Generated MAC stems***	***Manually Created MAC stems***
A=Use the item stem unchanged	**25%** (5 stems)	**75%** (15 stems)
B=make minor changes then use this item stem	**45%** (9 stems)	**15%** (3 stems)
C=Do not use this item stem	**30%** (6 stems)	**10%** (2 stems)

There is a significant difference between the percentages in all three categories, and significantly more generated items were placed in category C compared to the manually created items. Thus a simple judgment on the basis described in section 3.1 might lead to dismissal of the CREAM method. However, when compared with the outcome of previous experiments from which most source sentences were filtered out, a more positive view can be taken. It might be argued that items generated from source documents that have been so extensively pre-processed are more likely to be acceptable to the human reviewers who wrote the source documents. However for the featured domain, this control is judged to feasible because the domain is sufficiently well defined by the policy document library which has clearly defined boundaries and is protected by a well organized change management system.

5. Summary and Future Work

The CREAM method has been applied and a pragmatic, domain specific evaluation method of output from the system has been used. The most encouraging outcome from the analysis of the results is that 5 immediately-usable MAC test items have been generated (category A) and a significant number of items have been generated and then used following small alterations (category B). Although this does not meet the criteria specified earlier in section 3.1 whereby generated items need to be indistinguishable from manually created items when viewed by a domain expert. This experiment does provide some evidence to support continuation of investigations into this method of generating MAC test items automatically.

The development effort likely to be involved in creating the software to implement refined versions of the CREAM method is not insignificant, however it is my intention to persevere. Planned future work will include combining the CREAM method with previously published [1], [6] pre-processing methods and other relevant theories of human learning, controlled language and cognitive linguistics. Modifications will be applied both within the pre processing stage and the item generation stage as I continue to seek to improve the quality of the output from the MCQ test item generator software [4], [5] in the domain featured in this paper.

6. References

[1] Foster, R.M. 2009 *"Improving the Output from Software that Generates Multiple Choice Question (MCQ) Test Items Automatically using Controlled Rhetorical Structure Theory"* RANLP 2009, Borovets – Student Conference

[2] Brown J.C., Frishkoff G.A. Eskenazi M., 2005 *"Automatic Question Generation for Vocabulary Assessment"* Processing (HLT/EMNLP), pages 819–826, Vancouver, October 2005. © 2005 Association for Computational Linguistics

[3] Tsumori S., Kaijiri K., 2007 *"System Design for Automatic Generation of Multiple-Choice Questions Adapted to Students' Understanding"* 8th International Conference on Information Technology Based Higher Education and Training, 10th to 13th July 2007, Kumamoto, JAPAN

[4] Mitkov, R., and L. A. Ha. 2003. *"Computer-Aided Generation of Multiple-Choice Tests."* In Proceedings of HLT-NAACL 2003 Workshop on Building Educational Applications Using Natural Language Processing, pp. 17-22. Edmonton, Canada.

[5] Mitkov, R., L. A. Ha, and N. Karamanis. 2006. *"A computer-aided environment for generating multiple-choice test items."* Natural Language Engineering 12(2): 177-194.

[6] Foster, R.M. 2010 *"Improve the output from a MCQ test item generator using Statistical NLP"* ICALT 2010, Tunisia

[7] Sanders, T.J.M., Spooren, W.P.M., & Noordman, L.G.M. (1993). *"Coherence relations in a cognitive theory of discourse representation."* Cognitive Linguistics, 8, 93-133.

[8] Haladyna TM, Downing SM. *"How many options is enough for a multiple-choice test item?"* Educ Psychol Meas. 1993;53:999–1009

[9] Haladyna, T.M., Downing, S.M., Rodriguez, M.C., 2002 *"A Review of Multiple-Choice Item-Writing Guidelines for Classroom Assessment"* Applied measurement in education 15(3), 309–334

[10] Marie Tarrant M., Ware J.,Mohammed A.M. 2009 *"An assessment of functioning and non-functioning distractors in multiple-choice questions: a descriptive analysis"* BMC Med Educ. 2009; 9: 40. (10.1186/1472-6920-9-40. PMCID: PMC2713226)