Automatic Short Answer Marking

Stephen G. Pulman
Computational Linguistics Group,
University of Oxford.
Centre for Linguistics and Philology,
Walton St., Oxford, OX1 2HG, UK
sgp@clg.ox.ac.uk

Jana Z. Sukkarieh
Computational Linguistics Group,
University of Oxford.
Centre for Linguistics and Philology,
Walton St., Oxford, OX1 2HG, UK
Jana.Sukkarieh@clg.ox.ac.uk

Abstract

Our aim is to investigate computational linguistics (CL) techniques in marking short free text responses automatically. Successful automatic marking of free text answers would seem to presuppose an advanced level of performance in automated natural language understanding. However, recent advances in CL techniques have opened up the possibility of being able to automate the marking of free text responses typed into a computer without having to create systems that fully understand the answers. This paper describes some of the techniques we have tried so far vis-à-vis this problem with results, discussion and description of the main issues encountered.[1]

1. Introduction

Our aim is to investigate computational linguistics techniques in marking short free text responses automatically. The free text responses we are dealing with are answers ranging from a few words up to 5 lines. These answers are for factual science questions that typically ask candidates to state, describe, suggest, explain, etc. and where there is an objective criterion for right and wrong. These questions are from an exam known as GCSE (General Certificate of Secondary Education): most 16 year old students take up to 10 of these in different subjects in the UK school system.

2. The Data

Consider the following GCSE biology question:

Statement of the question	***Marking Scheme (full mark 3)***[2]
The blood vessels help to maintain normal body temperature. **Explain how the blood vessels reduce heat loss if the body temperature falls below normal.**	**any three:** vasoconstriction; explanation (of vasoconstriction); less blood flows to / through the skin / close to the surface; less heat loss to air/surrounding/from the blood / less radiation / conduction / convection;

Here is a sample of real answers:

1. all the blood move faster and dose not go near the top of your skin they stay close to the moses
2. The blood vessels stops a large ammount of blood going to the blood capillary and sweat gland. This prents the presonne from sweating and loosing heat.
3. When the body falls below normal the blood vessels 'vasoconstrict' where the blood supply to the skin is cut off, increasing the metabolism of the

[1] This is a 3-year project funded by the University of Cambridge Local Examinations Syndicate.

[2] X;Y/D/K;V is equivalent to saying that each of X, [L]={Y, D,K}, and V deserves 1 mark. The student has to write only 2 of these to get the full mark. [L] denotes an equivalence class i.e. Y, D, K are equivalent. If the student writes Y and D s/he will get only 1 mark.

body. This prevents heat loss through the skin, and causes the body to shake to increase metabolism.

It will be obvious that many answers are ungrammatical with many spelling mistakes, even if they contain more or less the right content. Thus using standard syntactic and semantic analysis methods will be difficult. Furthermore, even if we had fully accurate syntactic and semantic processing, many cases require a degree of inference that is beyond the state of the art, in at least the following respects:

- **The need for reasoning and making inferences:** a student may answer with *we do not have to wait until Spring*,which only implies the marking key *it can be done at any time*. Similarly, an answer such as *don't have sperm or egg* will get a 0 incorrectly if there is no mechanism to infer *no fertilisation*.
- **Students tend to use a negation of a negation (for an affirmative):** An answer like *won't be done only at a specific time* is the equivalent to *will be done at any time*. An answer like *it is not formed from more than one egg and sperm* is the same as saying *formed from one egg and sperm*. This category is merely an instance of the need for more general reasoning and inference outlined above. We have given this case a separate category because here, the wording of the answer is not very different, while in the general case, the wording can be completely different.
- **Contradictory or inconsistent information:** Other than logical contradiction like *needs fertilisation and does not need fertilisation*, an answer such as *identical twins have the same chromosomes but different DNA* holds inconsistent scientific information that needs to be detected.

Since we were sceptical that existing deep processing NL systems would succeed with our data, we chose to adopt a shallow processing approach, trading robustness for complete accuracy. After looking carefully at the data we also discovered other issues which will affect assessment of the accuracy of any automated system, namely:

- **Unconventional expression for scientific knowledge:** Examiners sometimes accept unconventional or informal ways of expressing scientific knowledge, for example, 'sperm and egg get together' for 'fertilisation'.
- **Inconsistency across answers:** In some cases, there is inconsistency in marking across answers. Examiners sometimes make mistakes under pressure. Some biological information is considered relevant in some answers and irrelevant in others.

In the following, we describe various implemented systems and report on their accuracy.
We conclude with some current work and suggest a road map.

3. Information Extraction for Short Answers

In our initial experiments, we adopted an Information Extraction approach (see also Mitchell et al. 2003). We used an existing Hidden Markov Model part-of-speech (HMM POS) tagger trained on the Penn Treebank corpus, and a Noun Phrase (NP) and Verb Group (VG) finite state machine (FSM) chunker. The NP network was induced from the Penn Treebank, and then tuned by hand. The Verb Group FSM (i.e. the Hallidayean constituent consisting of the verbal cluster without its complements) was written by hand. Relevant missing vocabulary was added to the tagger from the tagged British National Corpus (after mapping from their tag set to ours), and from examples encountered in our training data. The tagger also includes some suffix-based heuristics for guessing tags for unknown words.

In real information extraction, template merging and reference resolution are important components. Our answers display little redundancy, and are typically less than 5 lines long, and so template merging is not necessary. Anaphors do not occur very frequently, and when they do, they often refer back to entities introduced in the text of the question (to which the system does not have access). So at the cost of missing some correct answers, the information extraction components really consists of little more than a set of patterns applied to the tagged and chunked text.

We wrote our initial patterns by hand, although we are currently working on the development of a tool to take most of the tedious effort out of this task. We base the patterns on recurring head words or phrases, with syntactic annotation where neces-

sary, in the training data. Consider the following example training answers:

the egg after fertilisation splits in two	the fertilised egg has divided into two
The egg was fertilised it split in two	One fertilised egg splits into two
one egg fertilised which split into two	1 sperm has fertilized an egg.. that split into two

These are all paraphrases of *It is the same fertilised egg/embryo*, and variants of what is written above could be captured by a pattern like:

singular_det + <fertilised egg> +{<split>; <divide>; <break>} + {in, into} + <two_halves>, where
<fertilised egg> = NP with the content of 'fertilised egg'
singular_det = {the, one, 1, a, an}
<split> = {split, splits, splitting, has split, etc.}
<divide> = {divides, which divide, has gone, being broken...}
<two_halves> = {two, 2, half, halves}
etc.

The pattern basically is all the paraphrases collapsed into one. It is essential that the patterns use the linguistic knowledge we have at the moment, namely, the part-of-speech tags, the noun phrases and verb groups. In our previous example, the requirement that <fertilised egg> is an NP will exclude something like '*one sperm has fertilized an egg'* while accept something like '*an egg which is fertilized ...*'.

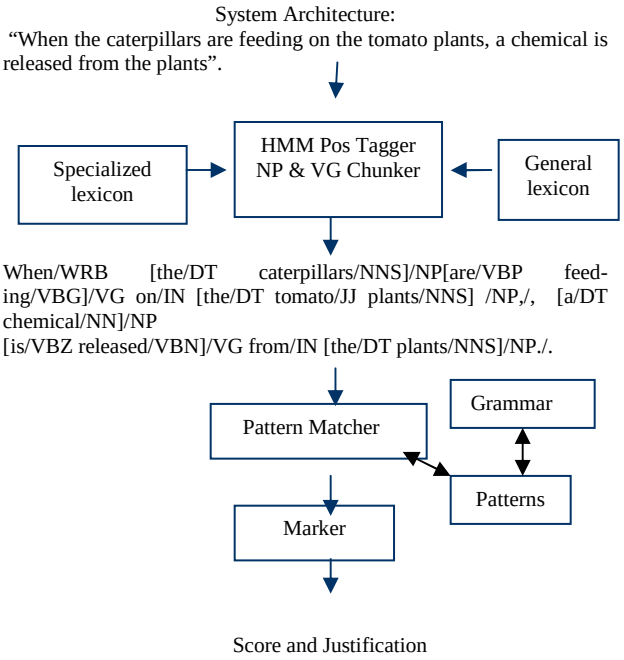


Table 1 gives results for the current version of the system. For each of 9 questions, the patterns were developed using a training set of about 200 marked answers, and tested on 60 which were not released to us until the patterns had been written. Note that the full mark for each question ranges between 1-4.

Question	Full Mark	% Examiner Agreement	% Mark Scheme Agreement
1	2	89.4	93.8
2	2	91.8	96.5
3	2	84	94.2
4	1	91.3	94.2
5	2	76.4	93.4
6	3	75	87.8
7	1	95.6	97.5
8	4	75.3	86.1
9	2	86.6	92
Average	----	84	93

Table 1. Results for the manually-written IE approach.

Column 3 records the percentage agreement between our system and the marks assigned by a human examiner. As noted earlier, we detected a certain amount of inconsistency with the marking scheme in the grades actually awarded. Column 4 reflects the degree of agreement between the grades awarded by our system and those which would have been awarded by following the marking scheme consistently. Notice that agreement is correlated with the mark scale: the system appears less accurate on multi-part questions. We adopted an extremely strict measure, requiring an exact match. Moving to a pass-fail criterion produces much higher agreement for questions 6 and 8.

4. Machine Learning

Of course, writing patterns by hand requires expertise both in the domain of the examination, and in computational linguistics. This requirement makes the commercial deployment of a system like this problematic, unless specialist staff are taken on. We have therefore been experimenting with ways in which a short answer marking system might be developed rapidly using machine learning methods on a training set of marked answers.

Previously (Sukkarieh et al. 2003) we reported the results we obtained using a simple Nearest

Neighbour Classification techniques. In the following, we report our results using three different machine learning methods: Inductive Logic progamming (ILP), decision tree learning(DTL) and Naive Bayesian learning (Nbayes). ILP (Progol, Muggleton 1995) was chosen as a representative symbolic learning method. DTL and NBayes were chosen following the Weka (Witten and Frank, 2000) injunction to `try the simple things first'. With ILP, only 4 out of the 9 questions shown in the previous section were tested, due to resource limitations. With DTL and Nbayes, we conducted two experiments on all 9 questions. The first experiments show the results with non-annotated data; we then repeat the experiments with annotated data. Annotation in this context is a lightweight activity, simply consisting of a domain expert highlighting the part of the answer that deserves a mark. Our idea was to make this as simple a process as possible, requiring minimal software, and being exactly analogous to what some markers do with pencil and paper. As it transpired, this was not always straightforward, and does not mean that the training data is noiseless since sometimes annotating the data accurately requires non-adjacent components to be linked: we could not take account of this.

4.1 Inductive Logic Programming

For our problem, for every question, the set of training data consists of students' answers, to that question, in a Prologised version of their textual form, with no syntactic analysis at all initially. We supplied some `background knowledge' predicates based on the work of (Junker et al. 1999). Instead of using their 3 Prolog basic predicates, however, we only defined 2, namely, *word-pos(Text,Word,Pos)* which represents words and their position in the text and *window(Pos2-Pos1,Word1,Word2)* which represents two words occurring within a *Pos2-Pos1* window distance.

After some initial experiments, we believed that a stemmed and tagged training data should give better results and that *window* should be made independent to occur in the logic rules learned by Progol. We used our POS tagger mentioned above and the Porter stemmer (Porter 1980). We set the Progol noise parameter to 10%, i.e. the rules do not have to fit the training data perfectly. They can be more general. The percentages of agreement are shown in table 2[3]. The results reported are on a 5-fold cross validation testing and the agreement is on whether an answer is marked 0 or a mark >0, i.e. pass-fail, against the human examiner scores. The baseline is the number of answers with the most common mark multiplied by 100 over the total number of answers.

Question	Baseline	% of agreement
6	**51,53**	74,87
7	**73,63**	90,50
8	**57,73**	74,30
9	**70,97**	65,77
Average	**71,15**	**77,73**

Table 2. Results using ILP.

The results of the experiment are not very promising. It seems very hard to learn the rules with ILP. Most rules state that an answer is correct if it contains a certain word, or two certain words within a predefined distance. A question such as 7, though, scores reasonably well. This is because Progol learns a rule such as *mark(Answer) only if word-pos(Answer,'shiver', Pos)* which is, according to its marking scheme, all it takes to get its full mark, 1. ILP has in effect found the single keyword that the examiners were looking for.
Recall that we only have ~200 answers for training. By training on a larger set, the learning algorithm may be able to find more structure in the answers and may come up with better results. However, the rules learned may still be basic since, with the background knowledge we have supplied the ILP learner always tries to find simple and small predicates over (stems of) keywords.

4.2 Decision Tree Learning and Bayesian Learning

In our marking problem, seen as a machine learning problem, the outcome or target attribute is well-defined. It is the mark for each question and its values are {0,1, …, full_mark}. The input attributes could vary from considering each word to be an attribute or considering deeper linguistic features like a head of a noun phrase or a verb group to be an attribute, etc. In the following experiments, each word in the answer was considered to be an attribute. Furthermore, Rennie et al. (2003)

[3] Our thanks to our internship student, Leonie IJzereef for the results in table 2.

propose simple heuristic solutions to some problems with naïve classifiers. In Weka, Complement of Naïve Bayes (CNBayes) is a refinement to the selection process that Naïve Bayes makes when faced with instances where one outcome value has more training data than another. This is true in our case. Hence, we ran our experiments using this algorithm also to see if there were any differences. The results reported are on a 10-fold cross validation testing.

4.2.1 Results on Non-Annotated data

We first considered the non-annotated data, that is, the answers given by students in their raw form. The first experiment considered the values of the marks to be {0,1, …, full_mark} for each question. The results of decision tree learning and Bayesian learning are reported in the columns titled DTL1 and NBayes/CNBayes1. The second experiment considered the values of the marks to be either 0 or >0, i.e. we considered two values only, pass and fail. The results are reported in columns DTL2 and NBayes2/CNBayes2. The baseline is calculated the same way as in the ILP case. Obviously, the result of the baseline differs in each experiment only when the sum of the answers with marks greater than 0 exceeds that of those with mark 0. This affected questions 8 and 9 in Table 3 below. Hence, we took the average of both results. It was no surprise that the results of the second experiment were better than the first on questions with the full mark >1, since the number of target features is smaller. In both experiments, the complement of Naïve Bayes did slightly better or equally well on questions with a full mark of 1, like questions 4 and 7 in the table, while it resulted in a worse performance on questions with full marks >1.

Ques.	Baseline	*DTL1*	*N/CNBayes1*	N/CNBayes2	DTL2
1	69	*73.52*	*73.52 / 66.47*	81.17 / 73.52	76.47
2	54	*62.01*	*65.92 /61.45*	73.18/ 68.15	62.56
3	46	*68.68*	*72.52 / 61.53*	93.95 / 92.85	93.4
4	58	*69.71*	*75.42 / 76*	75.42 / 76	69.71
5	54	*60.81*	*66.66 / 53.21*	73.09 / 73.09	67.25
6	51	*47.95*	*59.18 / 52.04*	81.63 /77.55	67.34
7	73	*88.05*	*88.05 / 88.05*	88.05 / 88.05	88.05
8	42	*41.75*	*43.29 / 37.62*	70.10/ 69.07	72.68
9	60	*61.82*	*67.20 / 62.36*	79.03 / 76.88	76.34
Ave.	60.05	*63.81*	*67.97/62.1*	79.51/77.3	74.86

Table 3. Results for Bayesian learning and decision tree learning on non-annotated data.

Since we were using the words as attributes, we expected that in some cases stemming the words in the answers would improve the results. Hence, we experimented with the answers of 6, 7, 8 and 9 from the list above but there was only a tiny improvement (in question 8). Stemming does not necessarily make a difference if the attributes/words that make a difference appear in a root form already. The lack of any difference or worse performance may also be due to the error rate in the stemmer.

4.2.2 Results on Annotated data

We repeated the second experiments with the annotated answers. The baseline for the new data differs and the results are shown in Table 4.

Question	Baseline	DTL	NBayes/CNBayes
1	58	74.87	86.69 / 81.28
2	56	75.89	77.43 / 73.33
3	86	90.68	95.69 / 96.77
4	62	79.08	79.59 / 82.65
5	59	81.54	86.26 / 81.97
6	69	85.88	92.19 / 93.99
7	79	88.51	91.06 / 89.78
8	78	94.47	96.31 / 93.94
9	79	85.6	87.12 / 87.87
Average	69.56	84.05	88.03 / 86.85

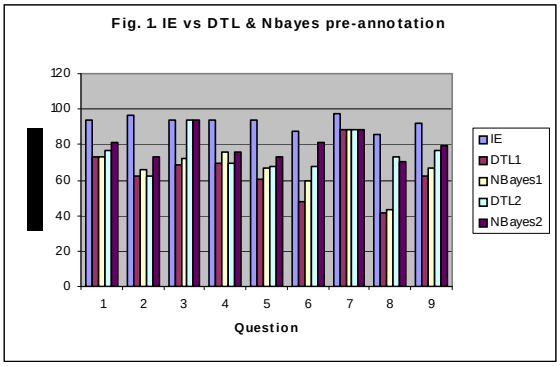
Table 4. Results for Bayesian learning and decision tree learning on annotated data.

As we said earlier, annotation in this context simply means highlighting the part of the answer that deserves 1 mark (if the answer has >=1 mark), so for e.g. if an answer was given a 2 mark then at least two pieces of information should be highlighted and answers with 0 mark stay the same. Obviously, the first experiments could not be conducted since with the annotated answers the mark is either 0 or 1. Bayesian learning is doing better than DTL and 88% is a promising result. Furthermore, given the results of CNBayes in Table 3, we expected that CNBayes would do better on questions 4 and 7. However, it actually did better on questions 3, 4, 6 and 9. Unfortunately, we cannot see a pattern or a reason for this.

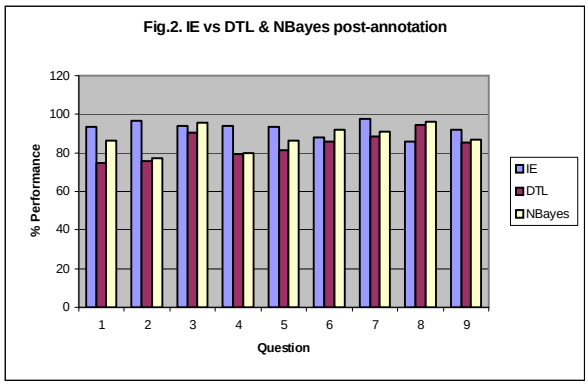
5. Comparison of Results

IE did best on all the questions before annotating the data as it can be seen in Fig. 1. Though, the training data for the machine learning algorithms is

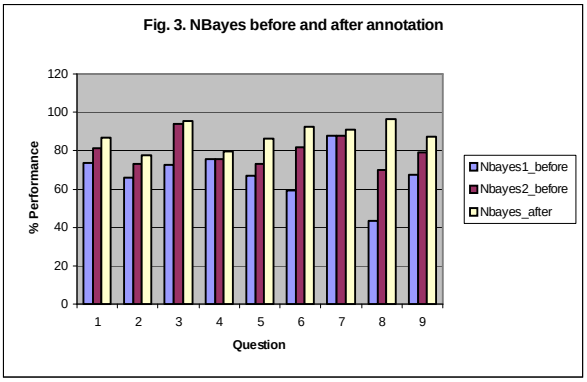
tiny relative to what usually such algorithms consider, after annotating the data, the performance of NBayes on questions 3, 6 and 8 were better than IE. This is seen in Fig. 2. However, as we said earlier in section 2, the percentages shown for IE method are on the whole mark while the results of DTL and Nbayes, after annotation, are calculated on pass-fail.



In addition, in the pre-annotation experiments reported in Fig. 1, the NBayes algorithm did better than that of DTL. Post-annotation, results in Fig. 2 show, again, that NBayes is doing better than the DTL algorithm. It is worth noting that, in the annotated data, the number of answers whose marks are 0 is less than in the answers whose mark is 1, except for questions 1 and 2. This may have an effect on the results.



Moreover, after getting the worse performance in NBayes2 before annotation, question 8 jumps to best performance. The rest of the questions maintained the same position more or less, with question 3 always coming nearest to the top (see Fig. 3). We noted that Count(Q,1)-Count(Q,0) is highest for questions 8 and 3, where Count(Q,N) is, for question Q, the number of answers whose mark is N. Also, the improvement of performance for question 8 in relation to Count(8,1) was not surprising, since question 8 has a full-mark of 4 and the annotation's role was an attempt at a one-to-one correspondence between an answer and 1 mark.



On the other hand, question 1 that was in seventh place in DTL2 before annotation, jumps down to the worst place after annotation. In both cases, namely, NBayes2 and DTL2 after annotation, it seems reasonable to hypothesize that P(Q1) is better than P(Q2) if Count(Q1,1)-Count(Q1,0) >> Count(Q2,1)-Count(Q2,0), where P(Q) is the percentage of agreement for question Q.

As they stand, the results of agreement with given marks are encouraging. However, the models that the algorithms are learning are very naïve in the sense that they depend on words only. Unlike the IE approach, it would not be possible to provide a reasoned justification for a student as to why they have got the mark they have. One of the advantages to the pattern-matching approach is that it is very easy, knowing which patterns have matched, to provide some simple automatic feed-back to the student as to which components of the answer were responsible for the mark awarded.

We began experimenting with machine learning methods in order to try to overcome the IE customisation bottleneck. However, our experience so far has been that in short answer marking (as opposed to essay marking) these methods are, while promising, not accurate enough at present to be a real alternative to the hand-crafted, pattern-

matching approach. We should instead think of them either as aids to the pattern writing process – for example, frequently the decision trees that are learned are quite intuitive, and suggestive of useful patterns – or perhaps as complementary supporting assessment techniques to give extra confirmation.

6. Other work

Several other groups are working on this problem, and we have learned from all of them. Systems which share properties with ours are C-Rater, developed by Leacock et al. (2003) at the Educational Testing Service(ETS), the IE-based system of Mitchell et al. (2003) at Intelligent Assessment Technologies, and Rosé et al. (2003) at Carnegie Mellon University. The four systems are being developed independently, yet it seems they share similar characteristics. Commercial and resource pressures currently make it impossible to try these different systems on the same data, and so performance comparisons are meaningless: this is a real hindrance to progress in this area. The field of automatic marking really needs a MUC-style competition to be able to develop and assess these techniques and systems in a controlled and objective way.

7. Current and Future Work

The manually-engineered IE approach requires skill, much labour, and familiarity with both domain and tools. To save time and labour, various researchers have investigated machine-learning approaches to learn IE patterns (Collins et al. 1999, Riloff 1993). We are currently investigating machine learning algorithms to learn the patterns used in IE (an initial skeleton-like algorithm can be found in Sukkarieh et al. 2004).

We are also in the process of evaluating our system along two dimensions: firstly, how long it takes, and how difficult it is, to customise to new questions; and secondly, how easy it is for students to use this kind of system for formative assessment. In the first trial, a domain expert (someone other than us) is annotating some new training data for us. Then we will measure how long it takes us (as computational linguists familiar with the system) to write IE patterns for this data, compared to the time taken by a computer scientist who is familiar with the domain and with general concepts of pattern matching but with no computational linguistics expertise. We will also assess the performance accuracy of the resulting patterns.

For the second evaluation, we have collaborated with UCLES to build a web-based demo which will be trialled during May and June 2005 in a group of schools in the Cambridge (UK) area. Students will be given access to the system as a method of self-assessment. Inputs and other aspects of the transactions will be logged and used to improve the IE pattern accuracy. Students' reactions to the usefulness of the tool will also be recorded. Ideally, we would go on to compare the future examination performance of students with and without access to the demo, but that is some way off at present.

References

Collins, M. and Singer, Y. 1999. *Unsupervised models for named entity classification.* Proceedings Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, pp. 189-196.

Junker, M, M. Sintek & M. Rinck 1999. *Learning for Text Categorization and Information Extraction with ILP*. In: Proceedings of the 1st Workshop on Learning Language in Logic, Bled, Slovenia, 84-93.

Leacock, C. and Chodorow, M. 2003. *C-rater: Automated Scoring of Short-Answer Questions*. Computers and Humanities 37:4.

Mitchell, T. Russell, T. Broomhead, P. and Aldridge, N. 2003. *Computerized marking of short-answer free-text responses*. Paper presented at the 29th annual conference of the International Association for Educational Assessment (IAEA), Manchester, UK.

Muggleton, S. 1995. *Inverting Entailment and Progol*. In: New Generation Computing, 13:245-286.

Porter, M.F. 1980. *An algorithm for suffix stripping*, Program, 14(3):130-137.

Rennie, J.D.M., Shih, L., Teevan, J. and Karger, D. 2003 *Tackling the Poor Assumptions of Naïve Bayes TextClassifiers*.
http://haystack.lcs.mit.edu/papers/rennie.icml03.pdf.

Riloff, E. 1993. *Automatically constructing a dictionary for information extraction tasks*. Proceedings 11th National Conference on Artificial Intelligence, pp. 811-816.

Rosé, C. P. Roque, A., Bhembe, D. and VanLehn, K. 2003. A hybrid text classification approach for analysis of student essays. In Building Educational Applications Using Natural Language Processing, pp. 68-75.

Sukkarieh, J. Z., Pulman, S. G. and Raikes N. 2003. *Auto-marking: using computational linguistics to score short, free text responses*. Paper presented at the 29th annual conference of the International Association for Educational Assessment (IAEA), Manchester, UK.

Sukkarieh, J. Z., Pulman, S. G. and Raikes N. 2004. *Auto-marking2: An update on the UCLES-OXFORD University research into using computational linguistics to score short, free text responses*. Paper presented at the 30th annual conference of the International Association for Educational Assessment (IAEA), Philadelphia, USA.

Witten, I. H. Eibe, F. 2000. *Data Mining*. Academic Press.