# A Region-based Approach to the Automated Marking of Short Textual Answers

* Raheel Siddiqi

*Abstract*— **Automated marking of short textual answers is a challenging task due to the difficulties involved in accurately "understanding" natural language text. However, certain purpose-built Natural Language Processing (NLP) techniques can be used for this purpose. This paper describes an NLP-based approach to automated assessment that extends an earlier approach [1] to enable the automated marking of longer answers as well as answers that are partially correct. In the extended approach, the original Question Answer Language (QAL) is augmented to support the definition of *regions* of text that are expected to appear in a student's answer. In order to explain the extensions to QAL, we present worked examples based on real exam questions. The system's ability to accurately mark longer answer texts is shown to be on a par with that of existing state-of-the-art short-answer marking systems which are not capable of marking such longer texts.**

Received: January 12, 2011, Accepted: June 27, 2011

*Index Terms*— **Automated assessment, short textual answers, automated free-text marking, region, Question Answer Language**

## I. INTRODUCTION

Automated marking of short textual answers has been an area of research for some years now and a number of systems have been developed [1]. Since the current state-of-the-art in automated marking does not allow a high degree of inference, all these systems are designed for close-ended factual questions rather than open-ended questions. One approach to marking such questions [2] exploited a language, called QAL, which was purpose-designed for the task. The following is a brief review of the system and its evaluation. The system's limitations are then used to justify extensions to QAL that are presented later in this paper.

The architecture of the system is depicted in Fig. 1. A number of components work together to analyze and compute the final score for the student's answer text. The processing of the student's answer text is performed in three phases: (1) spell-checking and correction, (2) parsing, and (3) comparison.

Spell-checking is performed by Jortho which is an Open Source spell-checker. The *Stanford Parser* is used to parse the student's answer text. This parser is a Treebank-trained statistical parser developed by Dan Klein and Christopher Manning at Stanford University and is capable of generating parses with high accuracy [3]. The Stanford Parser produces two types of output: (1) the part-of-speech tagged text, and (2) typed dependency grammatical relations between individual words. The part-of-speech tagged text is then "chunked" into noun phrases and verb groups by the *Noun Phrase & Verb Group chunker*.

In the third phase of processing, the tagged and chunked text is compared with the required syntactical structures. This task is performed by a *syntax analyzer*.

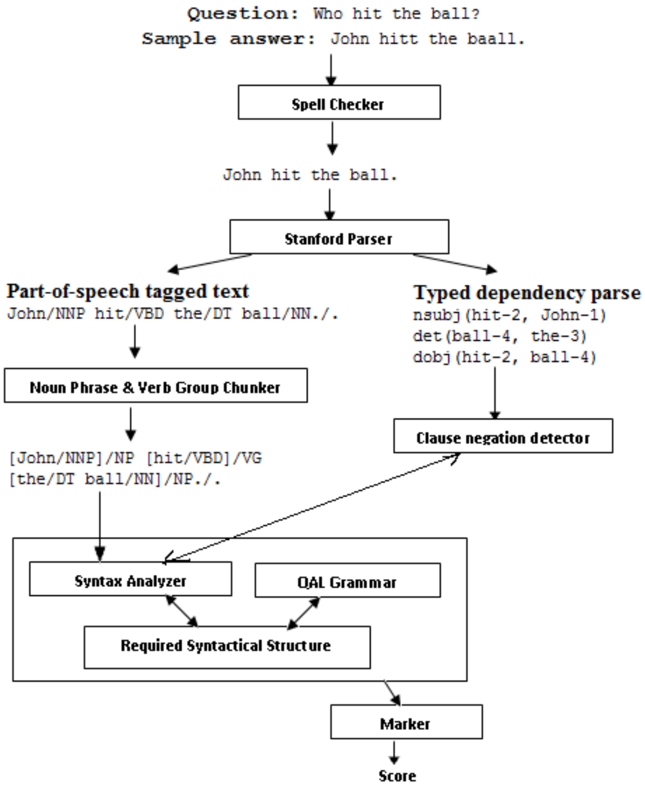

Question: Who hit the ball?
Sample answer: John hitt the baall.
Spell Checker
John hit the ball.
Stanford Parser
Part-of-speech tagged text
John/NNP hit/VBD the/DT ball/NN./.
Typed dependency parse
nsubj(hit-2, John-1)
det(ball-4, the-3)
dobj(hit-2, ball-4)
Noun Phrase & Verb Group Chunker
Clause negation detector
[John/NNP]/NP [hit/VBD]/VG
[the/DT ball/NN]/NP./.
Syntax Analyzer
QAL Grammar
Required Syntactical Structure
Marker
Score

Fig. 1. Architecture of the system developed

* *Department of Computer Engineering, Sir Syed University of Engineering and Technology, Karachi, Pakistan.* raheelsiddiqi@yahoo.com

Table 1. Evaluation results for the questions used [2]

| Question | Average answer length | Human-system agreement percentage |
|---|---|---|
| 1 | 4.8 words | 94.32% |
| 2 | 8.99 words | 95.19% |
| 3 | 4.73 words | 97.81% |
| 4 | 1.85 words | 99.56% |
| 5 | 17.4 words | 93.04% |

The required syntactical structures are specified in QAL. A *clause negation detector* is used to detect whether or not a particular clause in the student's answer text has been negated. In some situations it is important to detect this to ensure the correctness of a sentence. The comparison results from the syntax analyzer are passed on to the *marker* which computes the student's answer score [4], [5].

The system was evaluated using five questions that appeared in an undergraduate biology exam at *The University of Manchester (UK)*. All five questions worth a maximum of one mark each and therefore the student's answer text was normally relatively short [2]. Table 1 above gives the average length of the student's answer text for each question used, together with the evaluation result for that question. A total of 280 students' answers were used for each question (the training set consisted of around 50 students' answers while the remaining around 230 students' answers were utilized in the testing phase).

The state of the QAL and the system in was such that it was incapable of marking questions worth more than one mark [2]. It was also not possible to mark a multi-part question and award marks to answers that were partially correct. This means the capabilities of the system were quite limited and enhancements were required to enable the system to mark answers to a larger range of questions. The purpose of this paper is to present the extensions made to QAL, to illustrate the use of the extended QAL through worked examples, and to present and analyze evaluation results. Section 5 summarizes the contribution made and also gives possible directions for future research.

## II. MARKING PARTIALLY CORRECT ANSWERS

Question Answer Language (QAL) was designed to express the required syntactical structure in a suitable notation. A brief explanation of the QAL notation is given in table 2.

Consider the following example of how the expected syntactical structures are written in QAL.

**Question:** What single measurement would you make to confirm that an individual is anaemic?

**Model answer:** 1. Haemoglobin Concentration in whole blood.
2. Red cell count.

Table 2. Explanation of the QAL notation

| Notation | Meaning | Example |
|---|---|---|
| + | Sequence. | Alan+hit<br>*Meaning:* In the student answer text, the word “Alan” should appear before “hit”. |
| NP_containing (……) | Noun phrase containing any of the strings specified in the enclosed brackets. | NP_containing(“Karachi” \| “Lahore”)<br>*Meaning:* Noun phrase containing either “Karachi” or “Lahore”. |
| VG_containing(……) | Verb group containing any of the strings specified in the enclosed brackets. | VG_containing("facilitate" \| "assist" \| "ease" \| "help")<br>*Meaning:* Verb group containing either “facilitate” or “assist” or “ease” or “help”. |
| [……] | Condition. The only two allowed conditions are NO_NP and NO_VG which means “no noun phrase” and “no verb group” allowed at a particular location in the student answer text. | <DBMS>+[NO_VG]+<organize>+[NO_NP & NO_VG]+data<br>*Meaning:* There should be no verb group between the sub-patterns <DBMS> and <organize> and there should also be no noun phrase as well as no verb group between the sub-pattern <organize> and the word “data”. |
| {…} | Alternative options. Any one of the alternatives. | {count, counting}<br>*Meaning:* Any one of the two alternatives. |
| (…)*MinNum | A specified minimum number of options should appear in the student answer text. | (<organization>;storage;access;security;<integrity>)*3<br>*Meaning:* At least 3 of the 5 options should appear in the student answer text in any order. |

**Required syntactical structure specified in QAL:**

*Possibility #1*

```
<main>=NP_containing("haemoglobin
concentration"  |  "hb  concentration"  |
"hemoglobin concentration");
```

*Possibility #2*

```
<main>=NP_containing("concentration")+of+
NP_containing("haemoglobin"   |   "hb"   |
"hemoglobin");
```

*Possibility #3*

```
<main>={count,counting}+NP_containing("re
d blood cell" | "red cell" | "red blood
corpuscle” | “RBC”);
```

*Possibility #4*

```
<main>=NP_containing("red blood cell" |
"red cell" | "red blood corpuscle” |
“RBC”)+{count,counting};
```

This is a short-answer question worth one mark. There are two equally correct alternative model answers. The required syntactical structure written in QAL represents the structure of all paraphrases of these two model answers. The structure of each paraphrase is represented as a possibility.

When the paper was written, the system was not capable of marking partially correct answers [2]. All the questions used in evaluation were worth one mark each and the system either awarded the full mark or zero mark. However, it is important to award students appropriate marks for their partially correct answers. For example, consider the following question:

"*A blood sample was taken from a patient and he was found to have a high white cell count. On further investigation the patient was found to have a neutrophil count of 22 x10^9/L. Give two examples of what could this be indicative of?"*

The question is worth a maximum of two marks and the students are asked to provide two examples (i.e. each example is worth a maximum of one mark). If a student's answer contains only one correct example, only one mark should be awarded. The examiner specifies the required structure in QAL based on the model answer and also examplar marked students' answers. The following is the required syntactical structure for the question under consideration:

```
<main>=NP_containing("infection"):1:NP_contain
ing("inflammation" | "burn" |
"injury"):1:{cancer,leukemia}:1:{stress,exerci
se,physical}:1:<tissue>
:1:{corticosteroid,medicine,medication}:1:NP_c
ontaining("pregnancy"):1:NP_containing("bone
marrow disease" | "bone marrow disorder" |
"myeloproliferative"):1:;

<tissue>=
(<tissuePossibility1>;
<tissuePossibility2>)*1;

<tissuePossibility1> =
NP_containing("tissue injury" | "tissue
damage" | "tissue burn" | "tissue death");

<tissuePossibility2> =
{injury,damage,burn,death}+tissue;
```

There are four *patterns* in this structure. Each pattern is specified in the form of an equation. The left-hand side of the equation is the *pattern identifier* whereas the right-hand side is the *pattern body*. The main pattern is specified first and the word *"main"* is always used as its identifier. The main pattern consists of multiple parts and a numerical *mark* is associated with each part. Table 3 gives the parts and their associated marks when present in the main pattern of the above structure. The other three patterns, i.e. `<tissue>`, `<tissuePossibility1>` and `<tissuePossibility2>`, are *sub-patterns* of either the main pattern or another sub-pattern.

For example, if a student gives this answer:

*"This could be an indication of Leukemia or an individual who is just recovering from flu"*.

Table 3. The main pattern's parts and their associated marks

| S. No. | Parts of the main pattern | Associated Marks |
|---|---|---|
| 1 | NP_containing("infection") | 1 |
| 2 | NP_containing("inflammation" \| "burn" \| "injury") | 1 |
| 3 | {cancer,leukemia} | 1 |
| 4 | {stress,exercise,physical} | 1 |
| 5 | <tissue> | 1 |
| 6 | {corticosteroid,medicine,medication} | 1 |
| 7 | NP_containing("pregnancy") | 1 |
| 8 | NP_containing("bone marrow disease" \| "bone marrow disorder" \| "myeloproliferative") | 1 |

This answer matches with only one part of the main pattern (i.e. `{cancer,leukemia}:1:`). The mark associated with this part is 1 and therefore only 1 mark is awarded to the student by the system. This resolves the problem of marking partially correct answers. The new feature introduced into QAL is the ability to specify multiple parts of the main pattern together with their associated marks. The associated marks of the parts that are matched with the student's answer text are added to the student's score for that answer. Since there are many parts of the main pattern and each part is associated with some mark, the total of these associated marks is greater than the maximum marks for that question. So, if multiple parts of the main pattern matches with the student's answer text and the associated marks (of the matching parts) are added to the student's score for that question, the student's score may get greater than the maximum marks for that question. In such a case, the student's score is equated to the maximum marks for that question.

## III. USING THE CONCEPT OF "REGIONS" TO MARK LONGER ANSWERS

Another more important concept introduced into QAL is the use of a "regions"-like specification [6]. The syntactical structure for longer, multi-part answers can be easily represented if the expected answer text is considered to consist of various regions. Consider a question and its model answer given in table 4 (on the next page).

The question has three parts and therefore the model answer text can also be divided into three parts. Table 5 (on the next page) gives the three parts of the question and the associated model answer text.

For the last part of the question (i.e. "What could be the cause of this?"), the model answer text can be divided into two parts since two causes are provided. All these parts of the model answer text may be represented as "regions" that are expected in the student's answer text. A notation for the specification of expected "regions" in the student's answer text has been devised. As an example, consider the following "regions" specification for the question under consideration:

```
Begin_regions;
  Begin_region(marks=1);
```

```
    "Percentage of volume of whole blood
    occupied by red blood cells"
  End_region;
  Begin_region(marks=1);
    "Yes"
  End_region;
  Begin_region(marks=1.5);
    "if cells were smaller than usual for
    example microcytic due to iron deficiency"
  End_region;
  Begin_region(marks=1.5);
    "if higher than normal volume of plasma
    due to water loading, large infusion of
    fluids."
  End_region;
End_regions;
```

Table 4. A question (and its model answer) taken from an undergraduate biology exam at the University of Manchester (UK)

| | |
|---|---|
| **Question** | What do you understand by the term Haematocrit? Could a person have a normal RBC count but a low Haematocrit? What could be the cause of this? *(5 marks)* |
| **Model answer** | Percentage of volume of whole blood occupied by red blood cells.<br>Yes if cells were smaller than usual for example microcytic due to iron deficiency. Or if higher than normal volume of plasma due to water loading, large infusion of fluids. |

In the above example, the “regions” specification denotes that the structure of the answer should be such that its component parts are expected to contain the regions defined by the “Begin_region” and “End_region” symbols, i.e. a region is denoted by a matching pair of such symbols (containing the designated text) which must be nested within an enclosing “Begin_regions” and “End_regions” symbol pair. The software’s task is to search for these regions in the student’s answer text [4]. If a region is found in the student’s answer text, its associated marks are added to the total student’s score for that question. If it is required that the regions should appear in a specific order in the student’s answer text, then ‘Begin_sequence’….. ‘End_sequence’ symbols should be used instead of ‘Begin_regions’ ….. ‘End_regions’ to denote that the enclosed regions are to be in that specific sequence.

Once the “regions” specification has been developed, the next step is to specify the required syntactical structure for each region [4]. Each region’s syntactical structure consists of one or more possibilities [5]. The syntactical structure for each region of the “regions” specification is given below:

**Region:** Percentage of volume of whole blood occupied by red blood cells.

*Possibility #1*

```
<main>=({percentage,proportion,ratio,%};b
lood;<RBC>)*3:1:;
```

```
<RBC>=NP_containing("red blood cell" |
"red cell" | "RBC" | "red blood
corpuscle" | "erythrocyte" | "blood cell"
| "cellular" | "blood corpuscle" |
"cell");
```

Table 5. The question parts and the model answer text associated with each part

| Question part | Associated model answer text |
|---|---|
| What do you understand by the term Haematocrit? *(1 mark)* | Percentage of volume of whole blood occupied by red blood cells. |
| Could a person have a normal RBC count but a low Haematocrit? *(1 mark)* | Yes. |
| What could be the cause of this? *(3 marks)* | If cells were smaller than usual for example microcytic due to iron deficiency. Or if higher than normal volume of plasma due to water loading, large infusion of fluids. |

**Region:** Yes

*Possibility #1*

```
<main>=yes:1:true:1:correct:1:right:1:;
```

*Possibility #2*

```
<main>=<rbcCount>+<lowHematocrit>:1:<lowH
ematocrit> +<rbcCount>:1: {haematocrit,
hematocrit}+low+<rbcCount>:1: <rbcCount>
+{haematocrit,hematocrit} + low:1:;

<rbcCount>=NP_containing("RBC count" |
"red blood cell count" | "red cell count"
| "red blood corpuscle count");

<lowHematocrit>=NP_containing("low
hematocrit" | "low haematocrit");
```

*Possibility #3:*

```
<main>=number+<RBC>+<lowHematocrit>:1:
<lowHematocrit>+number+<RBC>:1:number+
<RBC>+{haematocrit,hematocrit}+low:1:
{haematocrit,hematocrit}+low+number
+<RBC>:1:;

<RBC>=NP_containing("red blood cell" |
"red cell" | "RBC" | "red blood
corpuscle" | "erythrocyte" | "blood cell"
| "cellular" | "blood corpuscle" |
"cell");

<lowHematocrit>=NP_containing("low
hematocrit" | "low haematocrit");
```

**Region:** If cells were smaller than usual for example microcytic due to iron deficiency.

*Possibility #1:*

```
<main>={microcytic,microcytosis,microcyte
}:1:iron:0.5:;
```

<u>*Possibility #2:*</u>

```
<main>=<RBC>+smaller:1:iron:0.5:;
```

```
<RBC>=NP_containing("red blood cell" |
"red cell" | "RBC" | "red blood
corpuscle" | "erythrocyte" | "blood cell"
| "cellular" | "blood corpuscle" |
"cell");
```

<u>*Possibility #3:*</u>

```
<main>=smaller+<RBC>:1:iron:0.5:;
```

```
<RBC>=NP_containing("red blood cell" |
"red cell" | "RBC" | "red blood
corpuscle" | "erythrocyte" | "blood cell"
| "cellular" | "blood corpuscle" |
"cell");
```

**Region:** If higher than normal volume of plasma (plasma volume) due to water loading, large infusion of fluids.

<u>*Possibility #1*</u>

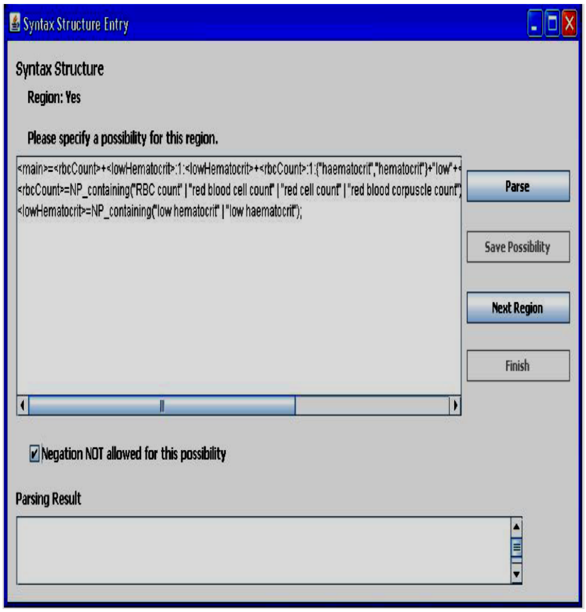
```
<main>={high,great,increase,more,large,mu
ch,expand,expansion,excess,extra,addition
}+{plasma,fluid,water}:1:{water,fluid,hyp
erhydration,overhydration,
hyperhydrated,overhydrated}:0.5:;
```

<u>*Possibility #2*</u>

```
<main>={plasma,fluid,water}+{high,great,i
ncrease,more,large,much,expand,
expansion,excess,extra,addition}:1:
{water, fluid, hyperhydration,
overhydration,
hyperhydrated,overhydrated}:0.5:;
```

For the region "yes" in the regions specification, the second and third possibilities of the required syntactical structure are problematic in certain cases. The purpose here is to specify the required syntactical structure for the second part of the question (i.e. "Could a person have a normal RBC count but a low Haematocrit?"). These two possibilities will make the system give credit to answers such as "A person can't have a normal RBC count but a low Haematocrit" which is not desirable. This problem can be resolved if the system analyzes the typed dependency parse of the student's answer text [2], [3]. Every syntactical structure possibility where negation is not allowed has to be marked (see fig. 2). This is recorded by the system and when the student's answer text is compared with the possibility structure, the system ensures that credit is only awarded if the matching string is not negated.

After explaining the extensions to QAL, we now present the Extended Backus Naur Form (EBNF) grammar of the extended QAL. The extended QAL is used for two purposes, i.e. "regions" specification and "syntactical structure" specification. The following is the EBNF grammar of the extended QAL (for both the "regions" and "syntactical structure" specifications):



Region: Yes

Please specify a possibility for this region.

```
<main>=<rbcCount>+<lowHematocrit>:1:<lowHematocrit>+<rbcCount>:1:{"haematocrit","hematocrit"}+"low"+<
<rbcCount>=NP_containing("RBC count" | "red blood cell count" | "red cell count" | "red blood corpuscle count"
<lowHematocrit>=NP_containing("low hematocrit" | "low haematocrit");
```

☑ Negation NOT allowed for this possibility

Fig. 2. The syntactical structure possibility where negation is not allowed is marked by ticking the checkbox shown in this figure

EBNF grammar of QAL for "regions" specification:

```
<region specification> ::=
('Begin_sequence;' | 'Begin_regions;'),
<white space>, <body>, <white space>,
('End_sequence;' | 'End_regions;');
```

```
<body> ::= 'Begin_region(marks=', number,
');', <white space>, <text>, <white
space>, 'End_region;',[{ <white space>,
'Begin_region(marks=', number, ');',
<white space>, <text>, <white space>,
'End_region;'}];
```

EBNF grammar of QAL for "syntactical structure" specification:

```
<syntactical structure specification> ::=
<main pattern name>, '=', <main pattern>,
';', [{<pattern name>, '=', <pattern>,
';'}];
```

```
<main pattern name> ::= '<','main','>';
```

```
<main pattern> ::= <pattern>, ':',
<number>, ':', [{<pattern>, ':',
<number>, ':'}];
```

```
<pattern name> ::= ‘<’, <alphabetic
character>, [{(<alphabetic character> |
<digit>)}], ‘>’;

<pattern> ::= (<noun phrase> | <verb
group> | <alternate> | <variable
sequence> | <pattern name> |
<condition>), [{‘+’ , (<noun phrase> |
<verb group> | <alternate> | <variable
sequence> | <pattern name> |
<condition>)}];

<verb group> ::= ‘VG_containing’,’(‘ ,
<text>, [{‘|’, <text>}]’)’;

<noun phrase> ::= ‘NP_ containing’,’(’,
<text>, [{‘|’, <text>}]’)’;

<alternate> ::= ‘{’ , (<string> |
<pattern name>), [{‘&’, (<string> |
<pattern name>)}], [{‘,’, (<string> |
<pattern name>), [{‘&’, (<string> |
<pattern name>)}]}]’}’;

<variable sequence> ::= ‘(’ , (<pattern
name> | <string> | <alternate>) , [{‘;’,
(<pattern name> | <string> |
<alternate>)}], ‘)’, ’*’, <digit>;

<condition> ::= ‘[’ , (‘NO_NP’ | ‘NO_VG’)
, [‘&’ , (‘NO_NP’ | ‘NO_VG’)] , ‘]’;

<text> ::= ‘“’ , <string> , ‘”’;

<string> ::= <word> [{<white space>
<word>}];

<word> ::= (<alphabetic character> |
<digit>), [{(<alphabetic character> |
<digit>)}];

<alphabetic character> ::= ‘A’ | ‘B’ |
‘C’ | ‘D’ | ‘E’ | ‘F’ | ‘G’ | ‘H’ | ‘I’ |
‘J’ | ‘K’ | ‘L’ | ‘M’ | ‘N’ | ‘O’ | ‘P’ |
‘Q’ | ‘R’ | ‘S’ | ‘T’ | ‘U’ | ‘V’ | ‘W’ |
‘X’ | ‘Y’ | ‘Z’ | ‘a’ | ‘b’ | ‘c’ | ‘d’ |
‘e’ | ‘f’ | ‘g’ | ‘h’ | ‘i’ | ‘j’ | ‘k’ |
‘l’ | ‘m’ | ‘n’ | ‘o’ | ‘p’ | ‘q’ | ‘r’ |
‘s’ | ‘t’ | ‘u’ | ‘v’ | ‘w’ | ‘x’ | ‘y’ |
‘z’;

<digit> ::= ‘0’ | ‘1’ | ‘2’ | ‘3’ | ‘4’ |
‘5’ | ‘6’ | ‘7’ | ‘8’ | ‘9’;

<white space> ::= ‘ ’ , [{‘ ’}];

<number> ::= <digit>, {<digit>}, [‘.’,
<digit>, {<digit>}];
```

## IV. THE SYSTEM EVALUATION

The system incorporating the enhancements presented in this paper was tested using questions taken from an undergraduate Biology exam at the University of Manchester (UK). Table 6 gives the questions used in the system's evaluation.

Table 6. The questions used in system evaluation

| S. No. | Question | Maximum Marks |
|---|---|---|
| 1 | A blood sample was taken from a patient and he was found to have a high white cell count. On further investigation the patient was found to have a neutrophil count of 22 x10^9/L. Give two examples of what could this be indicative of? | 2 |
| 2 | What do you understand by the term Haematocrit? Could a person have a normal RBC count but a low Haematocrit? What could be the cause of this? | 5 |
| 3 | What is hemolytic disease of the newborn? How can this be prevented? | 5 |

The salient features of the question set used are: (i) all the three questions are worth more than one mark, (ii) the last two questions are multi-part questions, and (iii) all three are close-ended, factual questions. A model answer and 60 manually marked students' answers were provided for each question and were used to write the "syntactical structure" and "region" specifications. Table 7 below gives the summary of system evaluation results. The students' answers used in this evaluation are significantly longer than those used in [2] and also longer than those used in other studies [7], [8], [9].

Table 7. Summary of the system evaluation results

| Question | | | | | |
|---|---|---|---|---|---|
| 1 | 16.81 words | 217 answers | 200 | 92.16% | 0.8678 |
| 2 | 67.07 words | 220 answers | 191 | 86.81% | 0.8303 |
| 3 | 147.11 words | 219 answers | 186 | 84.93% | 0.8017 |

= Average students' answer length
= Size of the test data
= Number of correct judgments
= Human-system agreement percentage
= Kappa

The human-system agreement rate is encouraging for all three questions given that an extremely strict measure, requiring an exact match, was adopted. Even a 0.5 mark difference between human and system-awarded marks was considered as an incorrect judgment by the system. It is also important to note that as the average answer length increases, the human-system agreement rate decreases.

The average human-system agreement percentage is approximately 88% and is better than that of the other state-of-the-art short-answer marking systems [8], [9]. Both the IE-based system [9] and C-rater [8] had an average human-system agreement rate of 84%. Our system appears to have better performance than the other systems but since the data set used to evaluate each system was different, it is difficult to accurately compare the performances of these respective systems. Commercial and resource constraints hinder us from testing all these systems with the same data.

Even though there is a limitation (as mentioned above) regarding any comparison of the different systems' performances, adopting Fleiss' [10] benchmark enables the regions-based system's performance to be assessed. According to Fleiss, "(Kappa) values greater than 0.75 or so may be taken to represent excellent agreement beyond chance, (kappa) values below 0.4 or so may be taken to represent poor agreement beyond chance, and (kappa) values between 0.4 and 0.75 may be taken to represent fair to good agreement beyond chance". Kappa values (in our case) represent inter-rater, i.e. human 'vs' automated marking, agreement beyond chance. Kappa values for all the three questions (used in the evaluation) are given in table 7. The Kappa value is greater than 0.75 in each case, therefore reflects excellent agreement beyond chance.

## V. CONCLUSION

This paper has presented an extension to the *Question Answer Language* (QAL) that enables QAL to be used to specify the structure of longer answer texts. Examples have been used to illustrate the extended-QAL. Evaluation results have been shown to be satisfactory with Kappa values indicating excellent human-system agreement beyond chance.

However, it is important to remember that assessment is a task in which compromise is not an option. It is expected to be extremely difficult to convince exam boards of respected educational institutions to solely rely on automated assessment because no matter how accurate an automated assessment system is, there will always be the possibility of a wrong judgment by the system. But, such a system can still be very effectively used as a "second check" on human marking, i.e. human marking is also imperfect because humans also make mistakes. If there is automated validation of human marking then the marking process has the potential to become increasingly accurate. Examiners will need to reassess answers where there is a discrepancy between human and automated marking. When used in this manner, such systems can make a more practical contribution as a tool to locate and correct inconsistencies and errors in human marking rather than as a direct substitute for human marking. In order to adapt the system for this role, a number of changes to the design of the software will be necessary and these modifications provide a basis for future research.

## ACKNOWLEDGMENT

The author would like to thank Dr. Christopher J. Harrison (of School of Computer Science, The University of Manchester, UK) for supervising the research work presented in this paper. The author is also indebted to Dr. Mary McGee Wood (of School of Computer Science, The University of Manchester, UK) for providing data used in the system evaluation.

## REFERENCES

[1] R. Siddiqi and C. J. Harrison (2008) "On the Automated Assessment of Short Free-Text Responses". Paper presented at the 34th annual conference of the International Association for Educational Assessment (IAEA), Cambridge, UK.

[2] R. Siddiqi and C. J. Harrison (2008) "A Systematic Approach to the Automated Marking of Short-Answer Questions". *Proceedings of the 12th IEEE International Multi topic Conference (IEEE INMIC 2008), Karachi, Pakistan,* pp. 329-332.

[3] M. C. de Marneffe, B. MacCartney and C. D. Manning, "Generating Typed Dependency Parses from Phrase Structure Parses," *Proc. The fifth international conference on Language Resources and Evaluation*, pp. 449-454, 2006.

[4] R. Siddiqi, C. J. Harrison and R. Siddiqi, "Improving Teaching and Learning through Automated Short-Answer Marking," *IEEE Transactions on Learning Technologies*, vol. 3, no. 3, pp. 237-249, July-September 2010.

[5] R. Siddiqi, "Improving Learning and Teaching through Automated Short-Answer Marking", PhD Thesis. The University of Manchester (UK), pp. 40-54, 2010.

[6] M. S. Powell, "An Input/Output Primitive for Object-Oriented Systems", *Information and Software Technology*, vol. 30, no. 1, pp 44-56, January/February 1988.

[7] T. Mitchell, T. Russel, P. Broomhead and N. Aldridge (2002) "Towards robust computerized marking of free-text responses". *Proceedings of the Sixth International Computer Assisted Assessment Conference, Loughborough, UK: Loughborough University* pp. 231-249.

[8] C. Leacock and M. Chodorow (2003) C-rater: Automated Scoring of Short-Answer Question. *Computers and the Humanities*, 37 (4), pp. 389-405.

[9] J. Z. Sukkarieh and S. G. Pulman (2005) "Automatic Short Answer Marking". *Proceedings of the 2nd Workshop on Building Educational Applications Using NLP, June 2005, Association for Computational Linguistics*, pp. 9-16.

[10] J. L. Fleiss, *Statistical Methods for Rates and Proportions.* New York: John Wiley & Sons, pp. 212-236, 1981.