Automatic Assessment of Open Ended Questions with a BLEU-Inspired Algorithm and Shallow NLP*

Enrique Alfonseca and Diana Pérez

Department of Computer Science
Universidad Autónoma de Madrid 28049 Madrid (Spain)
{Enrique.Alfonseca, Diana.Perez}@ii.uam.es

Abstract. This paper compares the accuracy of several variations of the BLEU algorithm when applied to automatically evaluating student essays. The different configurations include closed-class word removal, stemming, two baseline word-sense disambiguation procedures, and translating the texts into a simple semantic representation. We also prove empirically that the accuracy is kept when the student answers are translated automatically. Although none of the representations clearly outperform the others, some conclusions are drawn from the results.

1 Introduction

Computer-based evaluation of free-text answers has been studied since the sixties [1], and it has attracted more attention in recent years, mainly because the popularisation of e-learning courses. Most of these courses currently rely only on simple kinds of questions, such as multiple choices, fill-in-the-blanks or yes/no questions, although it has been argued that this way of assessment is not accurate enough to measure the student knowledge [2].

[3] classifies the techniques to automatically assess free-text answers in three main kinds:

- Keyword analysis, that only looks for coincident keywords or n-grams. These can be extended with the Vector Space Model and with Latent Semantic Indexing procedures [4].
- Full natural-language processing, which performs a full text parsing in order to have information about the meaning of the student's answer. This is very hard to accomplish, and systems relying on this technique cannot be easily ported across languages. On the other hand, the availability of a complete analysis of the student's essay allows them to be much more powerful. For instance, E-rater [5] produces a complete syntactic representation of the answers, and C-rater [6] evaluates whether the answers contain information related to the domain concepts and generates a fine-grained analysis of the logical relations in the text.
- Information Extraction (IE) techniques, that search the texts for some specific contents, but without doing a deep analysis. [3] describe an automatic system based on IE.

* This work has been sponsored by CICYT, project number TIC2001-0685-C02-01.

J. L. Vicedo et al. (Eds.): EsTAL 2004, LNAI 3230, pp. 25–35, 2004.

[7] provide a general overview of CAA tools.

In previous work, we have applied BLEU [8] to evaluate student answers [9, 10], with surprisingly good results, considering the simplicity of the algorithm. In this paper we focus on improving the basic BLEU algorithm with different representations of the student's text, by incorporating increasingly more complex syntactic and semantic information into our system.

The paper is organised as follows: in Section 2 we describe the variations of the original algorithm. Section 3 describes how the algorithm could be ported through languages automatically with a very slight loss in accuracy. Section 4 explains how it could be integrated inside an e-learning system. Finally, conclusions are drawn in Section 5.

2 Variations of the Scoring Algorithm

2.1 The Original BLEU Algorithm

BLEU [8] is a method originally conceived for evaluating and ranking Machine Translation systems. Using a few reference translations, it calculates an n-gram precision metric: the percentage of n-grams from the candidate translation that appear in any of the references. The procedure, in a few words, is the following:

1. For several values of N (typically from 1 to 4), calculate the percentage of n-grams from the candidate translation which appears in any of the reference texts. The frequency of each n-gram is limited to the maximum frequency with which it appears in any reference.
2. Combine the marks obtained for each value of N, as a weighted linear average.
3. Apply a brevity factor to penalise the short candidate texts (which may have many n-grams in common with the references, but may be incomplete). If the candidate is shorter than the references, this factor is calculated as the ratio between the length of the candidate text and the length of the reference which has the most similar length.

The use of several references, made by different human translators, increases the probability that the candidate translation has chosen the same words (and in the same order) as any of the references. This strength can also be considered a weakness, as this procedure is very sensitive to the choice of the reference translations.

2.2 Application in e-Learning

In the case of automatic evaluation of student answers, we can consider that the students' responses are the candidate translations, and the teacher can write a set of correct answers (with a different word choice) to be taken as references [9]. Contrary to the case of Machine Translation, where the automatic translation is expected to follow more or less rigidly the rhetorical structure of the original text, the students are free to structure their answers as they fancy, so it is to be expected a lower performance of BLEU in this case.

For evaluation purposes, we have built six different benchmark data from real exams, and an additional one with definitions obtained from Google Glossary[1]. The seven sets,

[1] http://www.google.com, writing "define:" in the query.

Table 1. Answer sets used in the evaluation. Columns indicate: set number; number of candidate texts (NC), mean length of the candidate texts (MC), number of reference texts (NR), mean length of the reference texts (MR), language (En, English; Sp, Spanish), question type (Def., definitions and descriptions; A/D, advantages and disadvantages; Y/N, Yes-No and justification of the answer), and a short description of the question

SET	NC	MC	NR	MR	Lang	Type	Description
1	38	67	4	130	En	Def.	"Operating System" defs. from "Google Glossary"
2	79	51	3	42	Sp	Def.	Exam question about Operating Systems
3	96	44	4	30	Sp	Def.	Exam question about Operating Systems
4	143	48	7	27	Sp	A/D	Exam question about Operating Systems
5	295	56	8	55	Sp	A/D	Exam question about Operating Systems
6	117	127	5	71	Sp	Y/N	Exam question about Operating Systems
7	117	166	3	186	Sp	A/D	Exam question about Operating Systems

which include more than 1000 answers altogether, are described in Table 1. All the answers were scored by hand by two different human judges, who also wrote the reference texts. The instructions they received were to score each answer in a scale between 0 and a maximum score (e.g. 1 or 10), and to write two or three reference answers for each question. We are currently transcribing other three sets corresponding to three more questions, but we still have only a few answers for each.

We have classified the ten questions in three distinct categories:

- Definitions and descriptions, e.g. *"What is an operative system?", "Describe how to encapsulate a class in C++"*.
- Advantages and disadvantages, e.g. *"Enumerate the advantages and disadvantages of the token ring algorithm"*.
- Yes/No question, e.g. *"Is RPC appropriate for a chat server? (Justify your answer)"*.

All the answers were marked manually by at least two teachers, allowing for intermediate scores if the answer was only partially correct. For instance, if the maximum score for a given question is defined as 1.5, then teachers may mark it with intermediate values, such as 0, 0.25, 0.5, 0.6, etc..

The discourse structure of the answer is different for each of these kinds. Definitions (and small descriptions) are the simplest ones. In the case of enumerations of advantages and disadvantages of something, students can structure the answer in many ways, and an Ngram-based procedure is not expected to identify mistakes such as citing something which is an advantage as a disadvantage.

2.3 Modified Brevity Penalty Factor

As discussed above, BLEU measures the n-gram precision of a candidate translation: the percentage of n-grams from the candidate that appear in the references. This metric is multiplied by a Brevity Penalty factor; otherwise, very short translations (which miss information) might get higher results than complete translations that are not fully accurate. In a way, this factor is a means to include recall intro the metric: if a candidate translation has the same length as some reference, and its precision is very high, then its recall is also expected to be high.

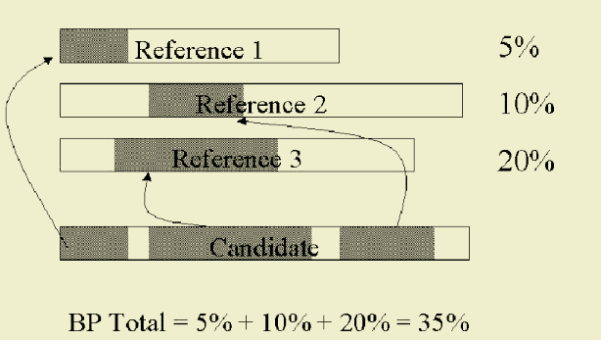

Fig. 1. Procedure for calculating the Modified Brevity Penalty factor

In contrast, ROUGE, a similar algorithm used for evaluating the output of automatic summarisation systems, only measures recall as the percentage of the n-grams in the references which appear in the candidate summary, because the purpose of a summary is to be maximally informative [11, 12]. For extract generation, in many cases precision can be taken for granted, as the summary is obtained from parts of the original document.

We argued, in previous work, that the BLEU's brevity penalty factor is not the most adequate one for CAA [9, 10]. In the case of student answers, both recall and precision should be measured, as it is important both that it contains all the required information, and that all of it is correct.

We currently encode the recall of the answer with a modified Brevity Penalty factor, that we calculate using the following procedure [10]:

1. Order the references in order of similitude to the candidate text.
2. For N from a maximum value (e.g. 10) down to 1, repeat:
   (a) For each N-gram from the candidate text that has not yet been found in any reference,
   (b) If it appears in any reference, mark the words from the N-gram as found, both in the candidate and the reference.
3. For each reference text, count the number of words that are marked, and calculate the percentage of the reference that has been found.
4. The Modified Brevity Penalty factor is the sum of all the percentage values.

Figure 1 describes how the factor is calculated. The results using this modified Brevity Penalty factor are better, statistically significant with 0.95 confidence. Surprisingly for us, the best result using this modified penalty factor was obtained just for unigrams. In automatic summarisation evaluations, unigrams have been found also to work better than n-grams in some cases [11, 12].

2.4 Extensions Proposed

There are a number of simple modifications to the original algorithm:

1. **Stemming:** to be able to match nouns and verbs inflected in different ways, e.g. *to manage* and *for managing*.

2. **Removal of Closed-Class Words.** These are usually important for finding matching N-grams for long values of N; however, in the case of unigrams, they are probably present in every kind of text and are not very informative. Given that the best correlation was obtained just for unigrams, these are not that important.
3. **Word-Sense Disambiguation.** If we were able to identify the sense intended by both the teacher and the student, then the evaluation would be more accurate. We do not have any Word-Sense Disambiguation procedure available yet, so we have tried with the following baseline methods:
   - For English, we have used the SEMCOR corpus [13] to find the most popular word sense for each of the words in WordNet, which is the sense we always take. In this case, we substitute every word w_i in the candidate and the references by the identifier of the synset such that w_i is tagged with that identifier in SEMCOR more times than with any other.
   - For Spanish, as we do not have a corpus with semantic annotations, we always choose, for every word, the first sense in the Spanish WordNet database [14].
   - For both languages, we have also tried by substituting each word by the list of the identifiers of all the synsets that contain it. In this case, we shall consider that two n-grams match if their intersection is not empty.

Figure 2 (in the next page) shows how the input text is modified before sending it to the modified BLEU algorithm. In any of these cases, the unigram co-occurrence metric is calculated after this processing. The algorithm is only modified in the last case, in which the procedure to check whether two unigrams match is not a string equality test, but a test that the set intersection is not empty.

2.5 Representing Syntactic Dependences

In order to extend the system with information about the syntactic dependences between the words in the texts, we have tried an extended version of the system in which the references and the candidate answer are analysed with a parser and next the dependences between the words are extracted. The library we have used for parsing is the `wraetlic` tools [15][2].

Figure 3 shows the dependences obtained from the candidate text from Figure 2. This can be taken as a first step in obtaining a logical representation of the text, but there are currently some limitations of our parser which do not allow us to produce a more reliable semantic analysis: it does not currently support prepositional-phrase attachment or coreference resolution.

2.6 Analysis and Discussion

Table 2 shows, for each data set and configuration of the algorithm, the results measured as the correlation between the teacher's scores and the scores produced automatically. Correlation is a metric widely used for evaluating automatic scoring systems [16, 6, 17, 7]. Given the array of scores X assigned by the teacher, and the array of scores Y assigned automatically, the correlation is defined as

[2] Available at www.ii.uam.es/~ealfon/eng/download.html

Original: Collection of programs that supervises the execution of other programs and the management of computer resources. An operating system provides an orderly input/output environment between the computer and its peripheral devices. It enables user-written programs to execute safely. An operating system standardizes the use of computer resources for the programs running under it.

Stemmed: [Collection, of, program, that, supervise, the, execution, of, other, program, and, the, management, of, computer, resource, An, operating, system, provide, an, orderly, input, environment, between, the, computer, and, its, peripheral, device, It, enable, user-written, program, to, execute, safely, An, operating, system, standardize, the, use, of, computer, resource, for, the, program, run, under, it]

Without closed-class words: [Collection, programs, supervises, execution, other, programs, management, computer, resources, operating, system, provides, orderly, input/output, environment, computer, peripheral, devices, enables, user-written, programs, execute, safely, operating, system, standardizes, use, computer, resources, programs, running]

Stemmed, no closed-class words: [Collection, program, supervise, execution, other, program, management, computer, resource, operating, system, provide, orderly, input, environment, computer, peripheral, device, enable, user-written, program, execute, safely, operating, system, standardize, use, computer, resource, program, run]

Most-common synset: [n06496793, of, n04952505, that, v01821686, the, n00842332, of, a02009316, n04952505, and, the, n00822479, of, n02625941, n11022817, An, operating, n03740670, v01736543, an, a01621495, n05924653, n11511873, between, the, n02625941, and, its, a00326901, n02712917, It, v00383376, user-written, n04952505, to, v01909959, r00152042, An, operating, n03740670, v00350806, the, n00682897, of, n02625941, n11022817, for, the, n04952505, v01433239, under, it]

Most-common synset, no closed-class words: [n06496793, n04952505, v01821686, n00842332, a02009316, n04952505, n00822479, n02625941, n11022817, operating, n03740670, v01736543, a01621495, n05924653, n11511873, n02625941, a00326901, n02712917, v00383376, user-written, n04952505, v01909959, r00152042, operating, n03740670, v00350806, n00682897, n02625941, n11022817,n04952505,v01433239]

All synsets: [[Collection], [of], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [that], [v01615271, v01821686], [the], [n00068488, n00817656, n00842332, n11140581], [of], [a02009316], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [and], [the], [n00822479, n06765853], [of], [n02625941, n07941303], [n04334536, n04749592, n11022817], ...]

All synsets, no closed-class words: [[Collection], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [v01615271, v01821686], [n00068488, n00817656, n00842332, n11140581], [a02009316], [n00391804, n04952505, n04952916, n05335777, n05390435, n05427914, n05472858, n05528119], [n00822479, n06765853], [n02625941, n07941303], [n04334536, n04749592, n11022817], ...]

Fig. 2. Modification of a candidate answer depending on the configuration of the automatic scorer. The synset identifiers in the last four cases are taken from WordNet 1.7

```
supervise([Collection], [execution, management]);
provide([Operating_System], [environment, devices]);
enable([It], [programs]);
execute([], []);
standardize([Operating_System], [use]);
run([resources], []);
user-written(program);
computer(resource);
of(programs);
of(resources);
for(programs);
under(it);
```

Fig. 3. Dependences obtained from the syntactic representation of the candidate answer

Table 2. Correlation between BLEU and the manual scores (left column), and correlations for the modified algorithm in different configurations (those from Figure 2) and, finally, the results using the syntactic dependences

Set	BLEU	Basic	stem	cc	stem-cc	WSD	WSD-cc	All	All-cc	Deps.
1	0.5886	0.5859	0.6189	0.5404	0.5821	**0.6322**	0.5952	0.2076	0.1125	0.3243
2	0.3609	0.5244	0.4832	**0.5754**	0.4797	0.4706	0.4655	0.1983	0.2968	0.2404
3	**0.3694**	0.3210	0.2364	0.3234	0.2917	0.2211	0.2844	0.1107	0.1431	0.1560
4	0.4159	0.6608	0.6590	0.6811	**0.7000**	0.6634	0.6933	0.6349	0.6702	0.4139
5	0.0209	0.1979	0.2410	0.2437	0.3013	0.2434	**0.3040**	-0.0201	0.0450	0.1884
6	0.2102	0.4027	0.3977	**0.4159**	0.4046	0.3822	0.3838	0.2607	0.3297	0.1302
7	0.4172	0.3970	0.4634	0.4326	0.4910	0.4727	**0.5261**	0.2880	0.3337	0.1726

$$correlation(X,Y) = \frac{covariance(X,Y)}{standardDev(X) \times standardDev(Y)}$$

Some observations can be drawn from these data:

- There is not any configuration that clearly outperforms the others.
- The removal of closed-class words improves the results, although it is not statistically significant.
- The rather simple Word-Sense Disambiguation procedure that we have used has attained the best results in three cases, and does not produce much loss for the remaining questions. We can take this as an indication that if we had a better algorithm these results might be better than the other configurations.
- The technique of choosing all the synsets containing a given word (with or without closed-class words) was clearly poorer than all the others.
- The metric obtained looking for coincidences of the dependences between words in the candidate answer and in the references was also inferior to the other configurations. This may be due to the fact that the dependences are only collected for some

Table 3. Correlation between BLEU and the manual scores (left column), and correlations for the modified algorithm in different configurations (those from Figure 2), using an automatic Machine Translation system

Set	BLEU	Basic	stem	cc	stem-cc	WSD	WSD-cc	All	All-cc	Deps.
1	0.5886	0.6174	0.6007	0.5663	0.5702	**0.6194**	0.5919	0.1519	0.0516	0.4081
2	0.3609	0.5330	0.4337	**0.5479**	0.5310	0.4176	0.4841	0.2276	0.2068	0.2501
3	**0.3694**	0.1660	0.1736	0.2892	0.2814	0.1734	0.3264	0.0789	0.2035	0.1210
4	0.4159	0.5937	0.6899	0.6066	0.7567	0.6998	**0.7655**	0.6008	0.6216	0.3897
5	0.0209	0.2449	0.2426	0.3213	**0.3459**	0.2358	0.3282	-0.0102	0.0220	0.0674
6	0.2102	0.3649	0.3326	0.3450	**0.3754**	0.3150	0.3586	0.1716	0.3070	0.1607
7	0.4172	0.4583	0.4635	0.4515	**0.4850**	0.4510	0.4699	0.2859	0.3452	0.1691

words because the parses are incomplete, and much information is lost using this representation.

– The correlation values obtained, although they are not high enough to be used in practical applications, are better than those obtained with other keyword-based evaluation metrics used in existing systems in combination with other techniques. Therefore, we believe that this procedure is very adequate to replace other keyword-based scoring modules in more complex evaluation environments (e.g. [5]).

3 Multilingual Evaluation

In order to check whether the evaluation system can be automatically ported across languages without the need of rewriting the reference texts, we have performed the following experiment: for each of the questions, we have translated the references manually into other language (from Spanish into English and vice versa) and we have translated the candidate texts using Altavista Babelfish[3]. Table 3 shows the results obtained.

In a real case of an e-learning application, the authors of a course would simply need to write the reference texts in their own language (e.g. Spanish). An English student would see the question translated automatically into English, would write the answer, and next the system would automatically translate it into Spanish and score it against the teacher's references. As can be seen from the results, the loss of accuracy is small; for some of the questions, the correlation in the best configuration even increases. Again, the removal of closed-class words seems to give better results, and there are two cases in which Word-Sense Disambiguation is useful.

4 Application on an On-line e-Learning System

Ideally, we would like that a student could submit the answer to a question and receive his or her score automatically. This system is not intended to substitute a teacher, but might help students in their self-study. We have built an on-line system for open-ended questions called Atenea.

[3] Available at http://world.altavista.com/

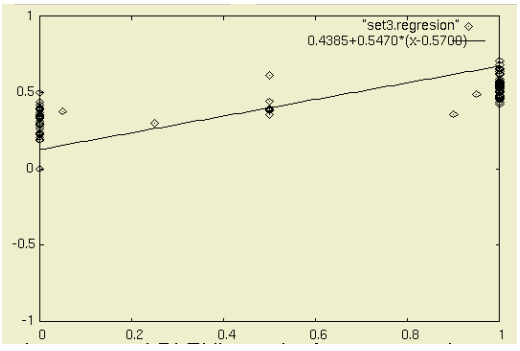

Fig. 4. Regression line between the teacher's scores and the system's scores

Given a student answer, BLEU provides a numerical score. This score needs to be put in the teacher's scale so the student can understand its meaning. For instance, Figure 4 shows the regression line for question 3. It can be seen that BLEU's scores (Y axis) are between 0 and 0.7, while the teacher marked all the answers between 0 and 1 (X axis). It is also interesting to notice that most of the teacher's scores are either 0 or 1: only a few answers received intermediate marks. In this example, if a student receives a BLEU score of 0,65, he or she would think that the answer was not very good, while the regression line indicates that that score is one of the best.

Therefore, it is necessary to translate BLEU's scores to the scale used by the teacher (e.g. between 0 and 1).

We propose the following methods:

- Evidently, if we have a set of student answers marked by a teacher, then we can calculate the regression line (as we have done in Figure 4) and use it. Given BLEU's score, we can calculate the equivalent in the teacher's score automatically. The regression line minimises the mean quadratic error.
- In same cases it may not be possible to have a set of answers manually scored. In this case, we cannot calculate the regression line, but we can estimate it in the following way:

  We take the student answers for which BLEU's scores, b_1 and b_2 are minimal and maximal, and we assume that their corresponding scores in the teacher's scale are 0 and 1 (i.e. they are the worst and the best possible answers). The estimated regression line will be the line that crosses the two points $(0, b_1)$ and $(1, b_2)$. This is only an approximation, and it has the unwanted feature that if a student produces an answer that scores best, then the remaining students will see their scores lowered down automatically, as the line will change.

Table 4 shows the mean quadratic errors produced by the regression line and the way to estimate the line unsupervisedly.

Table 4. Mean quadratic error for the several regression lines

Set	Regression	Two-points
1	0.81	6.33
2	8.29	15.03
3	6.78	8.50
4	17.41	22.73
5	25.77	48.08
6	15.59	16.13
7	5.10	29.63

5 Conclusions and Future Work

In previous work [9, 10] we described a novel application of the BLEU algorithm for evaluating student answers. We compare here several variations of the algorithm which incorporate different levels of linguistic processing of the texts: stemming, a tentative word-sense disambiguation and a shallow dependency-based representation obtained directly from a syntactic analysis. The results show that in nearly every case some of these modifications improve the original algorithm. Although no configuration clearly outperforms all the others, we can see that closed-class words removal is usually useful, and that improving the word-sense disambiguation module seems a very promising line to follow, given that a baseline procedure for WSD has been found effective for some datasets.

We also describe a feasible way in which this system might be integrated inside e-learning systems, with a little effort on behalf of the course authors, who would only be required to write a few correct answers for each of the questions. Although a set of manually corrected student answers might be desirable to minimise the mean quadratical error, there are roundabouts to omit that work. The coincident n-grams between the students' answers and the references can be useful so they can see which parts of their answers have improved their score. Finally, the characteristics of the algorithm make it very natural to be integrated in adaptive courses, in which the contents and tasks that the students must complete depend on their profiles or actions. Just by providing a different set of the reference answers (e.g. in other language, or for a different subject level), the same question can be evaluated in a suitable way depending on the student model. Furthermore, we have seen that it can be automatically ported across languages using a state-of-the-art Machine Translation system with no or small loss in accuracy.

Future work include the following lines:

- Improving the word-sense disambiguation module, and integrating it with a logical formalism of representation, so the predicates and their arguments are not words but synset identifiers.
- Study better models for estimating the regression line when the answers corrected by the teacher are not available.
- Extend the algorithm so that it is capable of discovering the internal structure of the answer. This would be desirable, for instance, when evaluating enumerations of advantages or disadvantages, where it is necessary to discover if the student is referring to one of the other.

– Explore the multilingual evaluation, to discover why is it the case that the correlation increases in some cases. A possible reason may be that the automatic translations employ a more reduced vocabulary.
– Perform a full integration of Atenea with the web-based adaptive e-learning system TANGOW [18], which has also been developed at our home university.

References

1. Page, E.B.: The use of computer in analyzing student essays. International review of education **14** (1968)
2. Whittington, D., Hunt, H.: Approaches to the computerized assessment of free text responses. In: Proceedings of the Int. CAA Conference. (1999)
3. Mitchell, T., Russell, T., Broomhead, P., Aldridge, N.: Towards robust computerised marking of free-text responses. In: Proceedings of the 6th International Computer Assisted Assessment (CAA) Conference, Loughborough, UK (2002)
4. Laham, D.: Automated content assessment of text using Latent Semantic Analysis to simulate human cognition. Ph.D. thesis, University of Colorado, Boulder (2000)
5. Burstein, J., Kukich, K., Wolff, S., Chi, L., Chodorow, M.: Enriching automated essay scoring using discourse marking. In: Proceedings of the Workshop on Discourse Relations and Discourse Marking, ACL, Montreal, Canada (1998)
6. Burstein, J., Leacock, C., Swartz, R.: Automated evaluation of essay and short answers. In: Proceedings of the Int. CAA Conference. (2001)
7. Valenti, S., Neri, F., Cucchiarelli, A.: An overview of current research on automated essay grading. Journal of I.T. Education **2** (2003) 319–330
8. Papineni, K., Roukos, S., Ward, T., Zhu, W.: Bleu: a method for automatic evaluation of machine translation (2001)
9. Pérez, D., Alfonseca, E., Rodríguez, P.: Application of the BLEU method for evaluating free-text answers in an e-learning environment. In: Proceedings of the Language Resources and Evaluation Conference (LREC-2004). (2004)
10. Pérez, D., Alfonseca, E., Rodríguez, P.: Upper bounds and extension of the BLEU algorithm applied to assessing student essays. In: IAEA-2004 Conference. (2004)
11. Lin, C.Y., Hovy, E.H.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003). (2003)
12. Lin, C.Y.: Rouge working note v. 1.3.1 (2004)
13. Fellbaum, C.: Analysis of a handtagging task. In: Proceedings of ANLP-97 Workshop on Tagging Text with Lexical Semantics: Why, What, and How?, Washington D.C., USA (1997)
14. Vossen, P.: EuroWordNet - A Multilingual Database with Lexical Semantic Networks. Kluwer Academic Publishers (1998)
15. Alfonseca, E.: Wraetlic user guide version 1.0 (2003)
16. Foltz, P., Laham, D., Landauer, T.: The intelligent essay assessor: Applications to educational technology. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning (1999)
17. Rudner, L., Liang, T.: Automated essay scoring using bayes' theorem. In: Proceedings of the annual meeting of the National Council on Measurement in Education. (2002)
18. Carro, R.M., Pulido, E., Rodríguez, P.: Dynamic generation of adaptive internet-based courses. Journal of Network and Computer Applications **22** (1999) 249–257