Using Bleu-like Algorithms for the Automatic Recognition of Entailment*

Diana Pérez and Enrique Alfonseca

Department of Computer Science,
Universidad Autónoma de Madrid,
Madrid, 28049, Spain
{diana.perez, enrique.alfonseca}@uam.es

Abstract. The Bleu algorithm has been used in many different fields. Another possible application is the automatic recognition of textual entailment. Bleu works at the lexical level, by comparing a candidate text with several reference texts in order to calculate how close the candidate text is to the references. In this case, the candidate is the text part of the entailment and the hypothesis is the unique reference. The algorithm achieves an accuracy of around 50%. Moreover, in this paper we explore the application of Bleu-like algorithms, finding that they can reach an accuracy of around 56%, which proves its possible use as a baseline for the task of recognizing entailment.

1 Introduction

In the framework of the Pascal Challenge, a fairly new and interesting task was tackled: the automatic recognition of textual entailment (RTE). It consists of deciding if a certain expression, a text called the entailment hypothesis (H), can be inferred by another expression, the text (T), and thus whether it can be said that T entails H or not.

This task deals with many different linguistic phenomena, such as language variability, since there are many different possible paraphrasings that can confuse an automatic system. For instance, if T and H are, respectively:

(1) a. Eyeing the huge market potential, currently led by Google, Yahoo took over search company Overture Services Inc last year.
b. Yahoo bought Overture.

a human annotator would know that, in this context, *"to take over"* is another way to say *"to buy"*. Hence, he or she would ignore the rest of the information and would mark the entailment as true. However, this task is not so straightforward for a computer. In fact, if it is not provided with some kind of resource indicating the paraphrase between T and H, it would mark the entailment as false.

* This work has been sponsored by the Spanish Ministry of Science and Technology, project number TIN2004-03140.

J. Quiñonero-Candela et al. (Eds.): MLCW 2005, LNAI 3944, pp. 191–204, 2006.

Obviously, it is a complex task that needs both a preliminary study to find out the most suitable techniques that can be applied to solve it, and the development of new techniques specifically designed for it. This problem has attracted a great deal of attention from the research community. In particular, seventeen different systems have been presented in the Pascal Challenge using several Natural Language Processing (NLP) techniques. These systems can be grouped according to the highest linguistic level in which their NLP techniques work:

- **Lexical:** systems that rely on studying word overlapping and/or statistical lexical relationships. For instance, the MITRE system [1].
- **Syntactic/Semantic:** systems that are based on the use of parsers to analyze and to match the sentences according to their syntactic structure. An example is the UIUC system [2]. They can also be underpinned by the use of world knowledge and/or the application of some kind of logical prover. For example, the Stanford system [3].

It is interesting to observe that according to the metrics given by the challenge organizers [4], the best result was an accuracy of 0.586 achieved by the systems [1, 5] (both of them working only at the lexical level) and a 0.686 Confidence Weight Score (CWS) value achieved by the Stanford system [3] (using statistical lexical relations, WordNet, syntactic matching, world knowledge and logical inference).

These facts lead us to our main motivation, that is to discuss if this problem can be addressed with just shallow techniques. If that is not the case, it will be interesting to know what the advantages of deep analyses are, and how the results differ from just using shallow techniques.

In this paper, we use the Bleu algorithm [6, 7], that works at the lexical level, to compare the entailing text (T) with the hypothesis (H). Once the algorithm was applied, it turned out that, despite its simplicity, it was able to achieve a result as good as an accuracy of 54% for the development set, and of around a 50% for the test set (CWS=52%).

It is important to highlight that Bleu requires less than two hours programming time and it does not use any NLP resource. On the other hand, it is our hypothesis that, in order to improve the results, it is appropriate to apply some NLP techniques. In order to test it, we have also tried other Bleu-like algorithms, increasing the accuracy up to 56% (CWS=54%). These results confirmed the use of Bleu-like algorithms as a possible baseline for the automatic recognition of textual entailment. Furthermore, they show how a shallow technique can reach an accuracy of around 56%.

This article is organized as follows: Section 2 explains how Bleu and other similar algorithms work in general, and next Section 3 details the application of these algorithms for recognizing entailment and gives the results achieved. Section 4 explores how shallow and deeper NLP techniques can contribute to this task. Finally, Section 5 ends with the main conclusions of the paper and some possible lines of future work.

2 BLEU-like Algorithms

The BLEU (BiLingual Evaluation Understudy) algorithm was created by Papineni *et al.* [6] as a procedure to rank systems according to how well they translate texts from one language to another. Basically, the algorithm looks for n-gram coincidences between a candidate text (the automatically produced translation) and a set of reference texts (the human-made translations). This algorithm is as follows:

- For several values of N (typically from 1 to 4), calculate the percentage of n-grams from the candidate translation that appear in any of the human translations. The frequency of each n-gram is limited to the maximum frequency with which it appears in any reference.
- Combine the marks obtained for each value of N, as a weighted linear average.
- Apply a Brevity Penalty factor to penalize short candidate texts (which may have 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.

It can be seen that BLEU is not only a keyword matching method between pairs of text. It considers several other factors that make it more robust:

- It takes into account the length of the text in comparison with the lengths of reference texts. This is because the candidate text should be similar to the reference texts (if the translation has been well done). Therefore, the fact that the candidate text is shorter than the reference texts is indicative of a poor quality translation and thus, BLEU penalizes it with a Brevity Penalty factor that lowers the score.
- The measure of similarity can be considered as a precision value that calculates how many of the n-grams from the candidate appear in the reference texts. This value has been modified, as the number of occurrences of an n-gram in the candidate text is clipped at the maximum number of occurrences it has in the reference texts. Therefore, an n-gram that is repeated very often in the candidate text will not increment the score if it only appears a few times in the references.
- The final score is the result of the weighted sum of the logarithms of the different values of the precision, for n varying from 1 to 4. It is not advisable to use higher values of n since coincidences longer than four-grams are very unusual.

BLEU's output indicates how similar the candidate and reference texts are. In fact, the higher the value is, the more similar they are. Papineni *et al.* report a correlation above 96% when comparing BLEU's scores with the human-made scores [6].

This algorithm has also been applied to evaluate text summarization systems with the modification that, in this case, the stress is put on the recall rather than on the precision [8]. This has motivated us to try a similar change in the original BLEU algorithm.

In particular, BLEU measures the recall in a rough way by penalizing very short translations using the Brevity Penalty factor. We have focused on improving this factor, by calculating it as the percentage of the reference text that is covered by the candidate text.

The resulting algorithm is called BLEU+**recall** and it is as follows:

1. For each value of N (typically from 1 to 4), calculate the Modified Unified Precision (MUP_N) as the percentage of N-grams from the candidate answer which appears in the reference text.
2. Calculate the weighted linear average of MUP_N obtained for each value of N. Store it in $combMUP$.
3. Calculate the Modified Brevity Penalty (MBP) factor, which is intended to penalize answers with a very high precision, but which are too short, to measure the recall:
   (a) For N from a maximum value (e.g. 10) down to 1, look whether each N-gram from the candidate text appears in the reference. In that case, mark the words from the found N-gram, both in the candidate and in the reference.
   (b) The MBP factor is the percentage of the reference that has been found in the candidate text.
4. The final score is the result of multiplying the MBP factor by $e^{combMUP}$.

BLEU+recall has been conveniently applied in the assessment of free-text answers combined with some shallow NLP techniques [9], using the `wraetlic` tools[1] [10]. These techniques are the following:

- **Stemming (ST):** To reduce each word to its stem or root form to facilitate the task of finding words with similar meanings but in different morphological forms. For instance, to match *books* and *book* as the former word is just the plural form of the latter.
- **Removal of closed-class words (CC):** To ignore functional words that have been tagged as closed-class words (e.g. prepositions, conjunctions, determiners, etc.) because they do not convey the main meaning of the sentence.
- **Word Sense Disambiguation (WSD):** To identify the sense in which polysemous words are used, using WordNet as the repository of word senses (see Section 3 for more details).

3 Application of BLEU-like Algorithms for Automatically Recognizing Textual Entailment

The corpus provided by the Pascal RTE Challenge organizers [4] consisted of 567 development entailment pairs and 800 test pairs. They have been gathered so that different linguistic levels were necessary to automatically judge entailment

[1] www.ii.uam.es/∼ealfon/eng/research/wraetlic.html

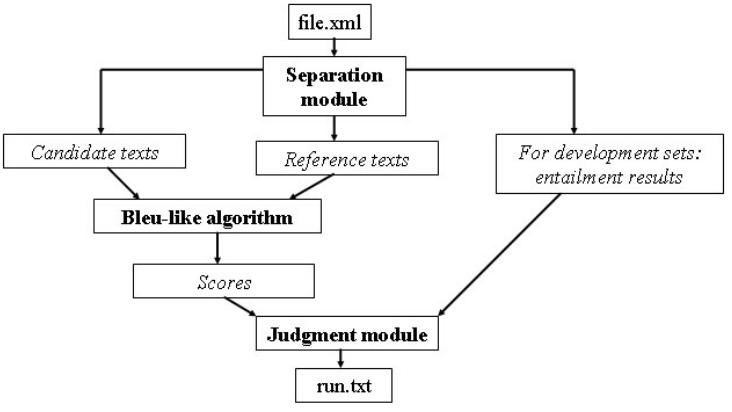

Fig. 1. Procedure to automatically recognize textual entailment using a BLEU-like algorithm

as TRUE or FALSE. They were also selected to produce a balanced corpus in which half of the entailment were TRUE according to human annotators. Whenever there was a disagreement between the human annotators about the nature of a pair, it was discarded.

Figure 1 shows the procedure for recognizing entailment using a BLEU-like algorithm. The first step is to use the *"Separation Module"* to split the initial corpus in two different sets[2], one with the T part of the entailment pairs and the other with the H part.

The second step is to decide whether the candidate text should be considered as the text part of the entailment (T) or as the hypothesis (and, as a consequence whether the reference text should be considered as the H or the T part). In order to make this choice, the length of the T and H parts and the dependency of the BLEU algorithm on the references should be taken into account. Initially, we considered the T part as the reference and the H as the candidate. This setting should have the advantage that the T part is usually longer than the H part and thus the reference would contain more information than the candidate. It could help BLEU's comparison process since the quality of the references is crucial and, in this case, the number of them has been dramatically reduced to only one (when in the rest of the applications of BLEU the number of references is always higher).

Then, the third step is to apply the algorithm as described in Section 2. The output is a score for each pair that enters the *"Judgement module"* to give a TRUE or FALSE value to each pair and also to be used as its confidence score. We performed an optimization procedure for the development set that chose the best threshold according to the percentage of success of correctly recognized entailment pairs. The value obtained was 0.157. Thus, if BLEU's output is higher than 0.157 the entailment is marked as TRUE, otherwise as FALSE.

[2] For the development sets, another output of this module is a file with the human annotators judgment for each pair.

Table 1. Results of using BLEU for recognizing the entailment in the development sets, considering from the first to seventh columns the T part of the entailment as the reference text (threshold = 0.157) and, from the eight to the final column the T part of the entailment as the candidate text (threshold = 0.1). The acronyms in the columns indicate: task id; number of entailment pairs (NTE); accuracy (A); number of pairs correctly judged as true (NTR); number of pairs correctly judged as false (NFR); number of pairs incorrectly judged as true (NTW); and, number of pairs incorrectly judged as false (NFW).

Task	NTE	A	NTR	NFR	NTW	NFW	NTE	A	NTR	NFR	NTW	NFW
CD	98	77%	39	36	12	11	98	72%	40	31	17	10
IE	70	44%	16	15	20	19	70	50%	23	12	23	12
MT	54	52%	18	10	17	9	54	52%	21	7	20	6
QA	90	41%	9	28	17	36	90	50%	22	23	22	23
RC	103	51%	30	23	28	22	103	50%	33	19	32	19
PP	82	57%	22	25	18	17	82	60%	25	24	19	14
IR	70	44%	10	21	14	25	70	41%	8	21	14	27
Total	567	53%	144	158	126	139	567	54%	172	137	147	111

Table 2. Results for the test set using BLEU (threshold = 0.1). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy.

TASK	CWS	Accuracy
CD	0.7823	0.7000
IE	0.5334	0.5000
MT	0.2851	0.3750
QA	0.3296	0.4231
RC	0.4444	0.4571
PP	0.6023	0.4600
IR	0.4804	0.4889
TOTAL	0.5168	0.4950

The results achieved are gathered in Table 1 (left). In order to confirm our insight that considering the T part of the entailment as the reference reaches better results, we repeated the experiment this time choosing the T part of the pair as the candidate and the H part as the reference. The results are shown in Table 1 (right). In this case, the best threshold has been 0.1. This is the value that has been fixed as threshold for the test set.

It can be seen how the results contradict our insight that the best setting would be to have the T part as the reference text. In fact, the results are not so much different for both configurations. A possible reason for this could be that all cases when BLEU failed to correctly judge the entailment are problematic in both settings. BLEU cannot deal with these cases neither taking the T part as the reference text nor taking it as the candidate text.

It is also important to highlight that the average accuracy achieved was of 54%. Moreover, it reached an accuracy of 72% for the Comparable Document

Table 3. Results for the test set using BLEU+**recall** (threshold = 0.9). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy considering the T part of the entailment pairs as the candidate text; and, next, then considering the H part of the pairs as the reference text.

TASK	CWS	Accuracy	CWS	Accuracy
CD	0.5847	0.6333	0.4629	0.4800
IE	0.5524	0.5333	0.4311	0.5000
MT	0.4771	0.4667	0.3632	0.4083
QA	0.5517	0.5846	0.5944	0.5000
RC	0.4976	0.5071	0.5872	0.5000
PP	0.4829	0.4600	0.4954	0.5200
IR	0.5091	0.5444	0.3814	0.5000
TOTAL	0.5194	0.5425	0.4730	0.4838

(CD) task. This result was expected since BLEU's strength relies on making comparisons between texts in which the lexical level is the most important.

The results for the test set (although a slightly lower than for the development test) confirm the same conclusions drawn before. In fact, for the first run in which BLEU was used for all the tasks, it achieved a confidence-weighted score of 52% and an accuracy of 50%. See Table 2 for details.

It can be seen that the results are better choosing the T part as the candidate text, and the H part as the reference, contrary to our initial insight. After analyzing the data set, we have seen that in many cases H is implied by T, but the reverse is not applicable, i.e. the entailment is unidirectional. This implies that it may be the case that most of H is covered by T, but a large portion of T is not covered by H. Therefore, the score returned by BLEU is lower if we consider T as the reference, because in these cases the hypothesis text is penalized by the Brevity Penalty Factor.

As can be seen, not only the overall performance continues being similar to accuracy obtained with the development test. Also, the best task for the test set keeps being the CD. To highlight this fact, we implemented a preliminary step of the algorithm in which there was a filter for the CD pairs, and only they were processed by BLEU. In this way, we created a second run with the CD set that achieved a CWS of 78% and an accuracy of 70%. This high result indicates that, although, in general, BLEU should only be considered as a baseline for recognizing textual entailment, in the case of CD, it can probably be used as a stand-alone system.

As indicated in Section 2, we have also tried several BLEU-like algorithms. For all them the threshold to decide whether the entailment should be judged as TRUE or FALSE was empirically determined as 0.9. They are the following:

- BLEU+**recall:** Following the procedure previously described but, using the algorithm described in Section 2 with the new Modified Brevity Penalty (MBP) factor, which takes into account not only the precision but also the recall. Table 3 shows the results both for considering the T part of the

Table 4. Results for the test set using BLEU+**recall+ST** (threshold = 0.9). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy considering the T part of the entailment pairs as the candidate text; and, next, considering the H part of the pairs as the reference text.

TASK	CWS	Accuracy	CWS	Accuracy
CD	0.5962	0.6200	0.4382	0.4667
IE	0.5475	0.5167	0.4375	0.5000
MT	0.4674	0.5083	0.3735	0.4167
QA	0.5799	0.6000	0.5857	0.5000
RC	0.4746	0.5143	0.5870	0.5000
PP	0.4902	0.4800	0.5257	0.5200
IR	0.5731	0.5556	0.4075	0.5000
TOTAL	0.5333	0.5500	0.4704	0.4825

Table 5. Results for the test set using BLEU+**recall+CC** (threshold = 0.9). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy considering the T part of the entailment pairs as the candidate text; and, next, considering the H part of the pairs as the reference text.

TASK	CWS	Accuracy	CWS	Accuracy
CD	0.5986	0.6067	0.5141	0.5000
IE	0.5522	0.5750	0.4437	0.5000
MT	0.4614	0.5000	0.5058	0.4667
QA	0.5310	0.5385	0.5896	0.5000
RC	0.4457	0.4500	0.5972	0.5000
PP	0.4703	0.5000	0.5124	0.5200
IR	0.5423	0.5111	0.4011	0.5000
TOTAL	0.5152	0.5300	0.5145	0.4963

entailment as the candidate or the reference and the H part as the reference or the candidate. It can be seen that while the CWS is kept of around 52%, the accuracy has been increased up to 54%. Using the T part as the candidate which continues to be the best configuration.

- BLEU+**recall+ST:** The improvement observed with the previous algorithm makes us think that, by further tuning the algorithm, more promising results could be achieved. Hence, we added an initial pre-processing step in which both the T and H part of the entailment pairs were stemmed. The results shown in Table 4 confirm our insight, as with this new step and using the T part as the candidate, an accuracy of 55% is reached and a CWS of 53%.
- BLEU+**recall+CC:** Although the removal of stop-words can produce worse results (e.g. [11]), we were intrigued about the effect of combining this step with BLEU+recall. However, it turned out that in our case it has also a negative effect decreasing the accuracy down to 53% and the CWS down to 52% (see Table 5). Perhaps, it could be solved by only removing certain stop-words and not all of them.

Table 6. Results for the test set using BLEU+**recall+WSD** (threshold = 0.9). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy considering the T part of the entailment pairs as the candidate text; and, next, considering the H part of the pairs as the reference text.

TASK	CWS	Accuracy	CWS	Accuracy
CD	0.6287	0.6200	0.4231	0.4667
IE	0.5804	0.5583	0.4333	0.5000
MT	0.4848	0.5250	0.3690	0.4167
QA	0.5554	0.5846	0.6028	0.5000
RC	0.4795	0.5143	0.5539	0.4929
PP	0.5351	0.4800	0.5173	0.5200
IR	0.5540	0.5444	0.4009	0.5000
TOTAL	0.5405	0.5550	0.4627	0.4813

Table 7. Results for the test set using BLEU+**recall+ST+CC+WSD** (threshold = 0.9). Columns indicate: task id; confidence-weighted score or average precision (CWS); and, the accuracy considering the T part of the entailment pairs as the candidate text; and, next, considering the H part of the pairs as the reference text.

TASK	CWS	Accuracy	CWS	Accuracy
CD	0.6035	0.5867	0.4413	0.4800
IE	0.5631	0.5583	0.4123	0.5000
MT	0.4729	0.4833	0.4613	0.4083
QA	0.5264	0.5615	0.5906	0.5000
RC	0.4739	0.4643	0.6113	0.5000
PP	0.5278	0.5000	0.5063	0.5200
IR	0.5375	0.5222	0.4973	0.5111
TOTAL	0.5267	0.5288	0.4979	0.4925

- BLEU+**recall+WSD:** This variant of the algorithm incorporates the use of WordNet 2.0 to identify the sense in which each word from the entailment pairs is used, using a WSD algorithm similar to [12], as described in [13] that measures the similarity between the context of the polysemous word in the entailment pair and the definition of the glosses in WordNet for its several senses. The gloss more similar to the context of the polysemous word is the one chosen and thus, the sense associated to that gloss is assigned to the word. The similarity metric is the cosine similarity based on the Vector Space Model (VSM). Given that one of the main problems to face when recognizing entailment is to deal with paraphrasings, we believe that this approach should give better results than the previous ones. This insight is proved by the results achieved: an accuracy of 56% and a CWS of 54% (see Table 6). It can be seen how both the accuracy and the CWS have reached with this configuration their highest value.
- BLEU+**recall+ST+CC+WSD:** The last algorithm that we have tried consists in combining BLEU+recall with all the NLP techniques contemplated.

Table 8. Results for several BLEU-like algorithms according to all the metrics used in the Pascal RTE Challenge[4]

Algorithm	Accuracy	CWS	Precision	Recall	f-measure
BLEU+recall	0.5425	0.5194	0.5282	0.7950	0.6347
BLEU+recall+ST	0.5500	0.5333	0.5312	0.8500	0.6538
BLEU+recall+CC	0.5300	0.5152	0.5349	0.4600	0.4946
BLEU+recall+WSD	**0.5550**	**0.5405**	0.5381	0.7775	0.6360
BLEU+recall+ST+CC+WSD	0.5267	0.5288	0.5406	0.3825	0.4480

Thus, the process would be as follows: first, the words of the entailment pairs are stemmed and the polysemous words are disambiguated, then the stop-words are removed and BLEU+recall is applied to give a score to each pair so that the "Judgment module" can decide according to the 0.9 threshold whether the entailment holds or not. Table 7 shows the results for this experiment. Again the configuration that uses T as the candidate gives the best results. It achieves an accuracy of 53% and a CWS of 53% that do not improve the only use of WSD (perhaps because of the negative effect of using the removal of closed-class words is still noticed when combined with other NLP techniques).

Finally, Table 8 summarizes the results for accuracy, CWS, precision, recall and f-measure for the five BLEU-like algorithms under test considering the T part of the entailment as the candidate text and with the 0.9 threshold.

4 Discussion

Automatically recognizing textual entailment is an interesting task that involves many complex linguistic phenomena. Seventeen different systems were presented at the Pascal RTE Challenge. They were based on very diverse techniques working at different linguistic levels. Nonetheless, all the results achieved were in the small range from 50% to 59% of accuracy. This section discusses how far shallow approaches can deal with this task and whether it is worthwhile to use deeper NLP techniques.

First of all, it is unclear whether this task can be completely solved just with automatic techniques. As indicated, the pairs used in the test set were those on which all the human annotators agreed. Even so, when several human researchers were asked to manually recognize entailment they only achieved an agreement of 91% [1]. Therefore, the complete task, including the discarded examples, can be considered difficult even for human judges. Perhaps, a possible solution for this can be to mark the entailment pairs not only as TRUE or FALSE, but also as DON'T KNOW, as proposed by Bos and Markert [14].

Our approach in the article has been to use BLEU-like algorithms. They only work at the lexical level and, thus, they cannot deal with examples in which the syntactic or semantic level are crucial to correctly solve the entailment. For

example, those cases in which the T and H parts are the same except for just one word that reverses the whole meaning of the text, as in the pair number 148 in the development set, whose T and H parts are

(2) a. The Philippine Stock Exchange Composite Index rose 0.1 percent to 1573.65
b. The Philippine Stock Exchange Composite Index dropped.

This is a very difficult case for BLEU-like algorithms. It will be misleading since they would consider that both T and H are saying something very similar, while in fact, the only words that are different in both texts, *"rose"* and *"dropped"*, are antonyms, making the entailment FALSE.

Another example is the pair number 1612 of the development set, whose T and H part are

(3) a. With South Carolina being Jesse Jackson's home state, there was a very strong incentive in the black community.
b. Jesse Jackson was born in South Carolina.

Any human annotator would know that this pair is true since in the T part it is said that South Carolina is Jesse Jackson's home state which is another way to say that Jesse Jackson was born in South Carolina. However, no BLEU-like algorithm would be able to identify this relationship without having any knowledge about this paraphrasing.

Other authors have found similar results such as Jijkoun *et al.* [15] that claimed their need for exploring deeper text features, Akhmatova [16] that stated that a deep semantic and syntactical analysis is vital to solve this problem and Herrera *et al.* [17] that declared that matching-based approaches were not enough (except perhaps for CD tasks) since a higher lexical overlap does not imply a higher semantic entailment.

On the other hand, it can be observed that despite the simplicity of BLEU and that it only works at the lexical level, it could be considered as a baseline for recognizing textual entailment [7]. In fact, this was our motivation to test similar algorithms such as BLEU+recall and combinations of BLEU+recall with NLP techniques such as stemming, removal of closed-class words and WSD. The results confirm our insight. In fact, BLEU+recall+WSD has reached an accuracy of 56% and a CWS of 54%, that are better than chance at the 0.05 level.

Some examples that are easily solved by these BLEU-like algorithms are:

- The pair of the development test with identifier 583, with the following T and H snippets:
  (4) a. While civilians ran for cover or fled to the countryside, Russian forces were seen edging their artillery guns closer to Grozny, and Chechen fighters were offering little resistance.
  b. Grozny is the capital of Chechnya.
  Since only the word Grozny is present both texts it will correctly mark it as false.
- The pair number 950 of the development set, with the following T and H snippets:

(5) a. Movil Access, a Grupo Salinas company, announced today that Gustavo Guzman will appoint Jose Luis Riera as company CFO of Grupo Iusacell.
b. Movil Access appoints Jose Luis Riera as CFO.

As the T part is included in the H part, the entailment will be correctly judge as true.

Furthermore, Bos and Markert [14] have observed that, when a shallow system is extended with deep NLP methods, the difference between the results they achieve is small. In fact, the accuracy of the first system is 0.5550, and that of the second system is just slightly higher, 0.5625.

5 Conclusion and Future Work

The discovery of entailment relationships is important for many NLP tasks [18]. In the framework of the Pascal RTE Challenge, an overview of the state-of-the-art of the field and, a study of which are the most promising techniques that should be used to face this task, took place.

Our approach is based on the use of the Bleu algorithm. Some conclusions that can be drawn from the experiments described in Section 2 are:

- Bleu can be used as a baseline for the task of recognizing entailment pairs, considering the candidate text as T and the reference text as the H part of the entailment, since it has achieved an accuracy of around 50%.
- Bleu's results depend greatly on the task considered. For example, for the Comparable Documents (CD) task it reaches its maximum value (77%) and for Information Retrieval (IR) the lowest (41%).
- Bleu has a slight tendency to consider a hypothesis as TRUE. In 319 out of 567 pairs, Bleu said the entailment was true. Out of these, it was right in 172 cases, and it was wrong in 147 cases. On the other hand, there were only 111 false negatives.

It is also interesting to observe that, although the origin of Bleu is to evaluate MT systems, the results for the MT task are not specially higher. The reason for that could be that Bleu is not being used here to compare a human-made translation to a computer-made translation, but two different sentences which contain an entailment expression, but which are not alternative translations of the same text in a different language.

Regarding Bleu-like algorithms, it has been seen how the potential of Bleu for this task can be further exploited reaching up to an accuracy of 56% and a CWS of 54% when a modification of Bleu takes into account the recall and incorporates WSD relying on WordNet was used.

The main limit of Bleu is that it does not use any semantic information and, thus, sentences with many words in common but with a different meaning will not be correctly judged.

A main conclusion of this paper is that shallow NLP techniques cannot be disregarded in this task. They have proved how useful they are, not only to serve

as baselines, but also as the basis for more complex systems and to obtain in a simple and fast way fairly good results compared to those reached by deeper techniques. All the same, in order to completely solve this task, we agree with the general opinion of the field that more resources are necessary. In particular, our best configuration used WordNet.

It would be interesting, as future work, to complement the use of BLEU+recall with some kind of syntactic processing and some treatment of synonyms and antonyms. For example, by combining it with a parser that translates all sentences from passive to active and allowed the comparison by syntactic categories such as subject, direct object, indirect object, etc.

As the Pascal Challenge organizers stated, it would be interesting to work towards the building of "semantic engines". This work would not only benefit the automatic recognition of entailment but several related NLP fields that suffer from similar problems such as the need of dealing with paraphrasings in the automatic assessment of free-text answers.

References

1. Bayer, S., Burger, J., Ferro, L., Henderson, J., Yeh, A.: Mitre's submissions to the eu pascal rte challenge. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
2. Salvo-Braz, R., Girju, R., Punyakanok, V., Roth, D., Sammons, M.: An inference model for semantic entailment in natural language. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
3. Raina, R., Haghighi, A., Cox, C., Finkel, J., Michels, J., Toutanova, K., MacCartney, B., Marneffe, M., Manning, C., Ng, A.Y.: Robust textual inference using diverse knowledge sources. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
4. Dagan, I., Glickman, O., Magnini, B.: The pascal recognising textual entailment challenge. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
5. Glickman, O., Dagan, I., Koppel, M.: Web based probabilistic textual entailment. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
6. Papineni, K., Roukos, S., Ward, T., Zhu, W.: BLEU: a method for automatic evaluation of machine translation. Research report, IBM (2001)
7. Perez, D., Alfonseca, E.: Application of the bleu algorithm for recognising textual entailments. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
8. Lin, C., Hovy, E.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of the Human Technology Conference 2003 (HLT-NAACL-2003). (2003)
9. Alfonseca, E., Pérez, D.: Automatic assessment of short questions with a BLEU-inspired algorithm and shallow nlp. In: Advances in Natural Language Processing. Volume 3230 of Lecture Notes in Computer Science. Springer Verlag (2004) 25–35
10. Alfonseca, E.: Wraetlic user guide version 2.0 (2005)
11. Wu, D.: Textual entailment recognition based on inversion transduction grammars. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)

12. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries. In: Proceedings of the 5th International Conference on Systems Documentation. (1986) 24–26
13. Ruiz-Casado, M., Alfonseca, E., Castells, P.: Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In: Advances in Web Intelligence. Volume 3528 of Lecture Notes in Artificial Intelligence. Springer Verlag (2005) 380–386
14. Bos, J., Markert, K.: Combining shallow and deep nlp methods for recognizing textual entailment. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
15. Jijkoun, V., Rijke, M.: Recognizing textual entailment using lexical similarity. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
16. Akhmatova, E.: Textual entailment resolution via atomic propositions. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
17. Herrera, J., Penas, A., Verdejo, F.: Textual entailment recognition based on dependency analysis and wordnet. In: Proceedings of the PASCAL Recognising Textual Entailment workshop, U.K. (2005)
18. Szpektor, I., Tanev, H., Dagan, I., Coppola, B.: Scaling web-based acquisition of entailment relations. In Lin, D., Wu, D., eds.: Proceedings of the EMNLP, Association for Computational Linguistics. (2004) 41–48