# The Role and Resolution of Textual Entailment in Natural Language Processing Applications

Zornitsa Kozareva and Andrés Montoyo

Departamento de Lenguajes y Sistemas Informáticos
Universidad de Alicante
{zkozareva,montoyo}@dlsi.ua.es

**Abstract.** A fundamental phenomenon in Natural Language Processing concerns the semantic variability of expressions. Identifying that two texts express the same meaning with different words is a challenging problem. We discuss the role of entailment for various Natural Language Processing applications and develop a machine learning system for their resolution. In our system, text similarity is based on the number of consecutive and non-consecutive word overlaps between two texts. The system is language and resource independent, as it does not use external knowledge resources such as WordNet, thesaurus, semantic, syntactic or part-of-speech tagging tools. In this paper all tests were done for English, but our system can be used with no restrains by other languages.

## 1 Introduction

Natural Language Processing applications, such as Question Answering, Information Extraction, Information Retrieval, Document Summarization and Machine Translation need to identify sentences that have different surface forms but express the same meaning. The semantic variability task is very important and its resolution can lead to improvement in system's performance. For this reason, researchers [15], [16], [7] draw attention of the semantic variability problem.

Major components in the modelling of semantic variability are paraphrase rules, where two language expressions can replace each other in a sentence without changing its meaning. Paraphrase rules range from synonyms, such as "purchase" and "buy", to complex expressions, such as "to kick the bucket" and "to die", "X assassinates Y " and "Y is the murderer of X". There are numerous paraphrase rules in a language and it is a laborious task to collect them all manually. This perception led in the last few years to a substantial effort in the direction of automatic discovery of paraphrase rules [3], [14], [4].

More general notion needed for applications that handle semantic variability is that of entailment rules [7]. An entailment rule is a directional relation between two language expressions, where the meaning of one can be entailed from the meaning of the other. According to the entailment definition of [7] for the sentence "Jane bought a car" entails the meaning of the sentence "Jane owns a car", but not vice versa. Entailment rules provide a broad framework for representing and recognizing semantic variability and are a generalization

of paraphrases, which correspond to bidirectional entailment rules (e.g. "X purchase Y " "X buy Y"). A text t is said to textually entail a hypothesis h if the truth of h can be most likely inferred from t [10]. Textual entailment recognition is a complex task that requires deep language understanding.

## 2 Entailment in NLP

Textual Entailment Recognition was proposed by [7] as a generic task that captures major semantic inference needs across many natural language processing applications. The textual entailment task requires to recognize, given two text fragments, whether the meaning of one text can be inferred from the other text.

The aim of current Question Answering (QA) system is to return brief answers in response to natural language questions. Given the question "Who is Albert Einstein?", the QA module has to find answers from large text collections related to this question. However, a question might be formulated using certain words and expressions while the corresponding answer in the corpus might include variations of the same expressions. Entailment can be seen in QA as identifying texts that entail the expected answer.

Given some static templates, Information Extraction (IE) systems try to extract the most salient elements in a text and identify the existing relations among these silent elements. A silent element can refer to a name of a person, organization, location etc., while the relations among them can be expressed in various ways: "Jane bought a car" or "A car is owned by Jane". In IE, textual entailment represent different text variants that express the same target relation.

The primary task of Information Retrieval (IR) is to retrieve set of relevant documents corresponding to given query search. The user who formulates the query is expecting to find documents containing these terms. However, a document may not contain all query terms and still to be relevant. A document about "orange" may be relevant to a query about "tropical fruit" yet the words "tropical" or "fruit" may be absent in that document. Entailment in IR are needed to help identifying when a document is relevant regardless of the occurrence or absence of the query tokens in the document. For IR, the entailment task can be seen as a query expression that should be entailed from relevant retrieved documents.

In Summarization entailment can be used to compute the informativity of one text segment compared to another one [20]. This is used to avoid redundancy, when one segment entails another segment, only the entailing segment should be included in the summary. Multi-document summarization systems need to deduce that different expressions found in several documents express the same meaning. For this application, entailment can be seen as omitting redundant sentence or expression from the summary that should be entailed from other expressions in the summary.

For Machine Translation (MT) this problem is expressed by identifying which of the produced translations is acceptable translations of a given source sentence. The translations may vary in word choice or in word order even when they use

the same words. Human MT are time consuming and expensive to produce, this lead current research to focus on the automatic MT evaluation. It consists of comparing the output of a MT system and one or more reference translations. The entailment task for MT can be seen as the evaluation of a correct translation that should be semantically equivalent to the gold standard translation, and thus both translations have to entail each other.

So far, we mentioned the role of textual entailment for various Natural Language Processing applications, and how their resolution can lead to improvement in the system's performance. Thus, in this paper we propose and develop a textual entailment system that is entirely based on machine learning. To our knowledge it is the first system that uses machine learning for the resolution of textual entailment. Our system does not use external resources such as WordNet, thesaurus, semantic, syntactic or part-of-speech tagging tools, which makes it resource independent. This system is evaluated on a standard textual entailment evaluation test. The obtained results are analyzed and future work is discussed.

## 3 System Overview

The entailment resolution system we develop, considers the number of common words or sequences of words between the entailing text ($T$) and the hypothesis ($H$). To each $(T, H)$ pair a set of features is associated. Based on these features an instance-based machine learning algorithm assesses the entailment relation as true or false. In Figure 1, we show the modules of our system.

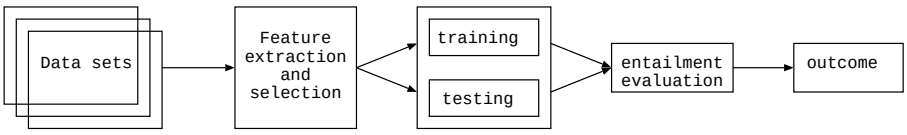


**Fig. 1.** Modules of the entailment system

### 3.1 Attributes

The characteristics we modelled for our machine-learning entailment system include some well known machine translation and text summarisation benchmark evaluation measures. The idea behind these measures is to evaluate how close an automatic machine translation is to a human one. This task is similar to the textual entailment task we are resolving, therefore we considered the usage of these attributes as proper for out textual entailment approach.

• Unigrams: The first attributes look for common unigram matches between $(T, H)$ pair. The measures $unigramT = \frac{1}{m}$ and $unigramH = \frac{1}{n}$, where $m$ corresponds to the number of words in T, $n$ corresponds to the number of words

in H and 1 stands for unigrams[1], detect common unigrams for the both texts. According to these measures, two sentences are not similar, when there is no common unigram, i.e. $unigramT = 0$ and $unigramH = 0$. For the sentences *Andres drinks tea.* and *John buys tea and donut.*, the only common unigram is *tea*.

• Bigrams: The more common n-grams two texts have, the more similar they are. Unigrams search for one consecutive common word, while bigrams discover two such consecutive common words. For this reason, bigrams are more reliable than unigrams. The measures are $bigramT = \frac{2}{m}$ and $bigramH = \frac{2}{n}$, where $m$ is the number of words in T, $n$ is the number of words in T and 2 stands for two consecutive common words between T and H. We do not apply word lemmatization to the data sets, so the occurance of more than two consecutive word matches is not frequent.

The measures we discussed so far are based on position-independent n-gram matches and are not sensitive to word order and sentence level structure. This may lead to errors and insufficient information for the correct similarity assignment between texts. Therefore, we introduce measures such as longest common subsequence and skip-grams.

• LCS: Longest common subsequence (LCS) measure looks for non-consecutive word sequences of any length. The intuition is that the longer the LCS is, the more similar the entailment text and the hypothesis are. LCS estimates the similarity between text T with length m and hypothesis H with length n, as $\frac{LCS(T,H)}{m}$ and $\frac{LCS(T,H)}{n}$ . These measures are known as ROUGE-L [13]. LCS does not require consecutive matches but in-sequence matches that reflect the sentence level word order. It automatically includes the longest in-sequence n-gram and therefore no predefined n-gram length is needed. By this measure we reflect the proportion of ordered words found in T and also present in H. An important characteristic of LCS is that it captures the sentence level structure in a natural way.

• Skip-grams: The skip-gram co-occurance statistics measure is known as ROUGE-S [13]. The skip-grams represent any pair of words in sentence order that allow arbitrary gaps. For the entailment text T and hypothesis H, we calculated bi, tri and four skip-grams. We did not go to upper n-gram level due to the high computational cost and the fact that the skip-grams with order higher than four occur rarely.

The measures are $skip\_gramT = \frac{skip\_gram(T,H)}{C(m,number\_of\_skip\_gram)}$ and $skip\_gramH = \frac{skip\_gram(T,H)}{C(n,number\_of\_skip\_gram)}$, where $skip\_gram(T,H)$ refers to the number of common skip grams found in T and H, $C(x, number\_of\_skip\_gram)$ is a combinatorial function, where $x$ is the number of words in the entailment text T (or the hypothesis H) and $number\_of\_skip\_grams$ corresponds to the number of common n-grams between $(T,H)$ [2].

[1] unigram means one word

[2] (e.g. $number\_of\_skip\_grams$ is 1 if there is a common unigram between T and H, 2 if there is a common bigram etc.)

In comparison with $LCS$ values which look for one longest common subsequence, the skip-grams find common non-consecutive words. In the example

$S_1$: John loved Mary.

$S_2$: John loves Mary.

$S_3$: Mary loves John.

the skip-gram measures identify that the similarity between the sentences $S_2$ and $S_1$ is stronger than the similarity between the sentences $S_3$ and $S_1$. However, the previous measures fail in measuring this similarity correctly. The results given by the $skip\_gram$ measures are more intuitive than $LCS$, $unigram$ or $bigram$.

• Negations: The whole entailment relation is changed when a negation is present. Two texts may be very similar, containing numerous common words, but when one of the texts has a negation, the entailment relation is transformed from true to false, or vice versa. To handle such cases, we introduced binary negation attributes for the T and H texts.

A negation present in T and not present in H, transforms the $(T, H)$ pair from true to false, or respectively from false to true. The same occurs when a negation is found in H and not in T, as in the following example "John knows Kate" and "John does not know Kate". The binary negation attributes are robust to cases where there is no negation at all or both text contain it. For such cases the entailment relation outcome of the $(T, H)$ pair depends on the values of the other attributes. In our experimental setup a pair $(T, H)$ with one or more negations has more weight than a pair with zero negations.

After we described the features of our machine learning system, we constructed feature vectors $\phi_i = \{f_1, f_2, ..., f_n\}$, where $i$ corresponds to the number of instances in the data set and $f_n$ is the number of features. We did not know from all the designed attributes, which would be the most informative ones for the resolution of the textual entailment task. Therefore, we applied a feature selection algorithm.

### 3.2 Feature Selection Algorithm

An important issue for every machine learning algorithm is the feature selection process. There are various feature selection algorithms, but for out system we used the algorithm described in Figure 2.

The output of the algorithm is the set of $unigramT$, $bigramT$, $LCS$ for T, $skip-gramT$, $skip-gramH$ and the two negation attributes. These attributes were determined as the most informative ones and were used for the final evaluation of our system.

### 3.3 Machine learning module

The machine learning algorithm we worked with is called Memory-based learning developed by [6]. It stores every training example in the memory. During testing, a new case is classified by extrapolating the most similar stored examples. The similarity between a new instance $X$ and all examples $Y$ in the memory is

Given:

- a set of all the designed features $F$=$\{f_1$,$f_2$,...,$f_n\}$;
- a set of selected features $SF$=$\emptyset$;

1. select a feature $f_i$ from $F$;
2. construct a classifier with the selected feature using 10-fold cross validation only on the training data set;
3. determine the feature $f_i$ leading to the best accuracy;
4. remove $f_i$ from $F$ and add it to $SF$;
5. go to 1 until no improvement is obtained.

**Fig. 2.** Feature selection algorithm

computed by the distance metric $\triangle(X,Y) = \sum_{i=1}^{n} \delta(x_i, y_i)$, where $\delta(x_i, y_i) = |\frac{x_i - y_i}{max_i - min_i}|$. To every test examples is assigned the category of the most similar training examples (k-nearest neighbors). We used the Memory-based learning algorithm with its default parameter settings[3].

## 4 Experiments

In order to estimate the performance of our developed entailment system, several experiments were conducted. We used the development and test data sets provided by the First Textual Entailment Recognition Challenge (RTE)[4] [8]. The examples in these data sets have been extracted from real Information Extraction, Information Retrieval, Question Answering, Machine Translation, Comparable Documents, Paraphrase Acquisition and Reading Comprehension applications.

The development set consisted of 567 text-hypothesis pairs, which we used as training examples and for testing we had another set of 800 text-hypothesis pairs. The provided data sets were prepared only for the English language. The next subsections describe the evaluation measures and the obtained results from the conducted experiments.

### 4.1 Entailment evaluation Measures

The returned classifications by our machine learning system were compared to the manually annotated test data set, and evaluated through the official RTE evaluation site[5]. The RTE evaluation script calculates accuracy, precision, recall and f-score measures for the whole system and individually for each one of the NLP tasks. A system is ranked according to its accuracy.

[3] the k-nearest neighbor is equal to 1
[4] http://www.pascal-network.org/Challenges/RTE/
[5] http://132.70.1.54:64080/cgi/rte_eval.pl

### 4.2 Results and Error Analysis

Every machine learning based system consists of training and testing phase. Once the most informative attributes were found by the feature selection algorithm and the classifier was trained, the set of 800 examples was tested. The achieved results are shown in Table1. In the same table are placed the performances of several systems participating in the RTE challenge. For each system, we listed systems' complete accuracy, precision and recall, as well as the accuracy scores for each one of the seven NLP tasks. Each system is denoted with the name of the first author and in the brackets can be found the reference number to the authors' paper. For two systems the precision and recall scores were not available, so we denoted them with X to indicate that they are missing.

Taking into consideration that the examples in the test data represent different levels of entailment reasoning, such as lexical, syntactic, morphological and logical, the obtained 54.13% accuracy using only word overlaps are very promising. The highest results per individual NLP application were for Comparable Documents 60.67% and for Paraphrase Acquisition 60.00%. While other systems had varying precision and recall measures, we obtained quite stable scores and significantly similar results for the various NLP tasks. Compared to the other systems, we achieved high score for the Paraphrase Acquisition task. This was due to the bidirectional n-gram measures we modelled and especially to the skip-gram measures.

| Systems | Acc. | Prec. | Rec. | CD | IE | MT | QA | RC | PP | IR |
|---|---|---|---|---|---|---|---|---|---|---|
| ourEnt | **54.13** | **54.11** | **54.25** | **60.67** | **53.33** | **53.33** | **52.31** | **53.57** | **60.00** | **45.56** |
| Pérez[19] | 49.50 | X | X | 70.00 | 50.00 | 37.50 | 42.31 | 45.71 | 46.00 | 48.89 |
| Wu[23] | 51.25 | X | X | 71.33 | 55.00 | 40.00 | 40.77 | 47.14 | 56.00 | 46.67 |
| Andreevska[2] | 51.90 | 55.00 | 18.00 | 63.00 | 52.00 | 47.00 | 45.00 | 48.00 | 50.00 | 53.00 |
| Akhmatova[1] | 51.88 | 61.19 | 10.25 | 58.67 | 50.83 | 49.17 | 47.69 | 52.14 | 52.00 | 51.11 |
| Zanzotto[18] | 52.40 | 52.65 | 49.75 | 76.51 | 46.67 | 52.10 | 39.53 | 48.57 | 54.00 | 44.44 |
| Kouylekov[12] | 56.60 | 55.00 | 64.00 | 78.00 | 48.00 | 50.00 | 52.00 | 52.00 | 52.00 | 47.00 |

**Table 1.** *Entailment recognition for various NLP applications*

From the RTE challenge, the system of Pérez [19], was the only one entirely relying on word overlaps. Their approach calculated the BLEU measure between a text T and a hypothesis H. They evaluated the entailment relation by a hand-made threshold. The $z'$ statistical test [9] showed that the 5% difference between our system and the one developed by [19], is significant. The disadvantage of their system comes from the BLEU measure which cannot handle nonconsecutive word matches, overcame in our case by the skip-gram measures. Another disadvantage is the hand-made threshold setting which their and many other systems relied on. In our system this process is completely automatic, handled by the machine learning approach.

The other systems listed in Table 1 incorporated resources such as WordNet, lexical chains, logical forms, syntactic parsing etc. The conducted experiment showed that a knowledge poor method like the one we presented, outperformes some systems utilizing external knowledge. However, other participating systems as the one of [22], [21], [17] combined various information sources and covered more entailment relations. Their work directs us in the future toward the incorporation of knowledge rich information sources.

We present some textual entailment sentences that our system was able to identify correctly. Each $(T, H)$ pair belongs to one NLP task.

- CD_task number 1808:
  **T:** Vanunu converted to Christianity while in prison, and has been living in an anglican cathedral in Jerusalem since his release on April 21.
  **H:** A convert to Christianity, Vanunu has sequestered himself at a Jerusalem church since he was freed on April 21.
- IE_task number 1871:
  **T:** The third African Union summit opened in the Ethiopia's capital of Addis Ababa on Tuesday, June 29.
  **H:** The third African Union summit is held in Addis Ababa.
- MT_task number 1301:
  **T:** The former wife of the South African president did not ask for amnesty, and her activities were not listed in the political reports submitted by the African National Congress to the Truth and Reconciliation Commission in 1996 and 1997.
  **H:** Winny Mandela, the President's ex-wife, is requesting amnesty.
- QA_task number 1498:
  **T:** If Russia is excluded from NATO membership while Poland and Hungary are on the path to becoming NATO allies, how can Moscow avoid the conclusion that a renewed and enlarged NATO retains its traditional objective of confronting Russia?
  **H:** Moscow is the capital of Russia.
- RC_task number 1112:
  **T:** Every year, 1.2 million people in America have a new or repeat heart attack.
  **H:** Every year, 1.2 million Americans have a heart attack.
- MT_task number 1301:
  **T:** The former wife of the South African president did not ask for amnesty, and her activities were not listed in the political reports submitted by the African National Congress to the Truth and Reconciliation Commission in 1996 and 1997.
  **H:** Winny Mandela, the President's;s ex-wife, is requesting amnesty.
- IR_task Number 967:
  **T:** Chadrick Fulks escaped from a Kentucky jail.
  **H:** Chadrick Fulks gets the death penalty.

The n-gram features we modelled are precise with short sentences. These measures have the tendency to punish large text by considering the final outcome of the ratio of the overlapping words as false e.g. the textual entailment between the sentences does not hold.

Other errors that occurred in the presented approach concern the number and the time mismatching. For sentences like "Bill met Mary in 1998" and "Bill met Mary in 1999", the number of common words is high, however we need a time matching attribute to determine that the entailment relation does not hold

because 1998 and 1999 are not the same time period. Our system should be able to reason for the two sentences "I woke up before 10" and "I woke up at 9" that "before 10" indicates "at 9".

We failed in the recognition of other examples where Named Entity Recognition module was needed. We need to identify that "Mexican" and "Mexico" are similar, that "John Parker" and "Parker Ltd" are not the same as one is a name of a person and the other is a name of an organization.

The negation attribute handled surface negations, where the particle "not" was present. However, this attribute lacks in defining that "He is a bad boy" and "He is a good boy" do not infer the same meaning. In order to resolve this problem, antonyms are needed.

Finally, but not on a last place, text similarity module as proposed by [11] and [5] is needed in order to establish the semantic relatedness of the words, e.g. "apple" is a type of "fruit". The presently modelled attributes fail in matching that there is a synonym, hyponym or hypernym relation among the words. The incorporation of semantic information will relate the words and establish the textual entailment relation with better precision.

As can be seen from the conducted experiment and the results of other systems, the resolution of textual entailment is a very difficult, but challenging task. Present systems using sophisticated probabilistic models [7] and various information sources [21],[5] achieve as a maximum 62% accuracy. To our knowledge, we are the first entirely based machine-learning entailment system functioning with word overlaps. Our system can be easily adapted to languages other than English, because counting words between two sentences is not a language dependent task.

## 5 Conclusions and work in progress

In this paper we discussed the impact and the role of textual entailment for various Natural Language Processing applications. In order to handle the language variability matter, we designed and developed a completely automatic and language independent machine learning system. This system was evaluated on several NLP applications such as Information Extraction, Information Retrieval, Question Answering, Comparable Documents, Paraphrase Acquisition, Machine Translation and Reading Comprehension. The overall accuracy achieved by our system is 54.13%. The majority of the correctly resolved entailment relations were found for Comparable Documents and Paraphrase recognition. In a comparative study with other entailment systems evaluated on the same data sets, the results show that our system achieved the highest score for Paraphrase Acquisition and yields comparable results to the other systems.

The attributes we worked with are based on common words and sequences of word overlaps, which makes our system easy to be adapted and incorporated for languages other than English. In our approach we did not need to develop a hand-made threshold through which the system should decide if an entailment relation holds or not. This process was completely automatic, handled by the

machine learning algorithm. Besides its facility of language independence, we claim that our system is also resource independent. Tools as WordNet, syntactic, semantic or part-of-speech tagging were neither needed nor utilized. The system we developed is extremely usefull and practical for many NLP tasks and different languages.

In the future, we will tune and specialize the described entailment recognition system, for a crosslingual Question Answering and Information Retrieval needs. We are interested in the exploration and combination of probabilistic models and information from the web as described in [22]. To improve the negation attribute, we will include the antonym relation from WordNet. We will examine the robustness of our entailment system, with the participation in the Second Textual Entailment Challenge[6].

## Acknowledgements

This research has been partially funded by the Spanish Government under project CICyT number TIC2003-07158-C04-01 and PROFIT number FIT-340100-2004-14 and by the Valencia Government under project numbers GV04B-276.

## References

1. Elena Akhmatova. Textual entailment resolution via atomic propositions. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 61–64.
2. Alina Andreevska, Zhuoyan Li, and Sabine Bergler. Can shallow predicate argument structure determine entailment? In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 45–48.
3. Regina Barzilay and Kathleen McKeown. Extracting paraphrases from a parallel corpus. In *ACL, 2001.*, pages 50–57.
4. Regina Barzilay and Kathleen McKeown. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *HTLT-NAACL, 2003.*, pages 16–23.
5. Courtney Corley and Rada Mihalcea. Measures of text semantic similarity. In *Proceedings of the ACL workshop on Empirical Modeling of Semantic Equivalence.*, 2005.
6. Walter Daelemans, Jakub Zavrel, Ko van der Sloot, and Antal van den Bosch. Timbl: Tilburg memory-based learner. Technical Report ILK 03-10, Tilburg University, November 2003.
7. Ido Dagan and Oren Glickman. Probabilistic textual entailment: Generic applied modeling of language variability. In *PASCAL workshop on Text Understanding and Mining, 2004.*
8. Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*

[6] www.pascal-network.org/Challenges/RTE2/

9. Thomas G. Dietterich. Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, 1998.
10. Oren Glickman. *Applied Textual Entailment.* PhD thesis, Bar Ilan University, 2005.
11. Valentin Jijkoun and Maarten de Rijke. Recognizing textual entailment using lexical similarity. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 73–76.
12. Milen Kouylekov and Bernardo Magnini. Recignizing textual entailment with tree edit distance algorithm. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 17–20.
13. Chin-Yew Lin and Franz Josef Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-birgam statistics. In *Proceedings of ACL-2004*. Barcelona, Spain, 2004.
14. Dekang Lin and Patrik Pantel. Discovery of inference rules for question answering. *Natural Language Engineering, 4(7)*, pages 343–360.
15. Dan Moldovan and Vasile Rus. Logic form transformation of wordnet and its applicability to question answering. In *ACL*, pages 394–401, 2001.
16. Christof Monz and Maarten de Rijke. Lightweight entailment checking for computational semantics. In *ICoS-3*.
17. Eamonn Newman, Nicola Stokes, John Dunnion, and Joe Carthy. Ucd iirg approach to the textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 53–56.
18. Maria Teresa Pazienza, Marco Pannacchiotti, and Fabio Massimo Zanzotto. Textual entailment as syntactic graph distance: a rule based and svm based approach. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 25–28.
19. Diana Pérez and Enrique Alfonseca. Application of the bleu algorithm for recognising textual entailments. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 9–12.
20. Dragomir Radev. A common theory of information fusion from multiple text sources. In *Proceedings of the First SIGdial Workshop on Discourse and Dialogue*, pages 74–83, 2000.
21. Rajat Raina, Aria Haghighi, Christopher Cox, Jenny Finkel, Jeff Michels, Krsitina Toutanova, Bill MacCartney, Marie-Catherine de Marneffe, Christopher D. Manning, and Andrew Y. Ng. Robust textual inference using diverse knowledge sources. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 57–60.
22. Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. Scaling web-based acquisition of entailment relations. In *Proceedings of Empirical Methods in Natural Language Processing*, 2004.
23. Dekai Wu. Textual entailment recognition based on inversion transduction grammars. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment, 2005.*, pages 37–40.