INESC-ID at ASSIN: measuring semantic similarity and recognizing textual entailment

INESC-ID at ASSIN: medidor de similaridade semântica e classificador de inferência textual

Pedro Fialho
Universidade de Évora, INESC-ID
pedro.fialho@l2f.inesc-id.pt

Ricardo Marques
IST/UTL
ricardo.sa.marques@tecnico.ulisboa.pt

Bruno Martins
IST/UTL, INESC-ID
bruno.g.martins@tecnico.ulisboa.pt

Luísa Coheur
IST/UTL, INESC-ID
luisa.coheur@l2f.inesc-id.pt

Paulo Quaresma
Universidade de Évora, INESC-ID
pq@di.uevora.pt

Abstract

In this work we present INESC-ID@ASSIN, the system from INESC-ID that competed in the 2016 joint evaluation effort entitled "Avaliação de Similaridade Semântica e Inferência Textual" (ASSIN) in the tasks of semantic similarity and textual entailment recognition. INESC-ID@ASSIN addresses the problem of detecting sentence similarity as a regression task, and it addresses textual entailment as a classification task. Although INESC-ID@ASSIN relies mainly on simple lexical features for detecting paraphrases and recognizing textual entailment, promising results were achieved.

Keywords

supervised learning, regression, classification

Resumo

Neste artigo apresentamos o sistema INESC-ID@ASSIN do INESC-ID, o qual competiu no evento "Avaliação de Similaridade Semântica e Inferência Textual" (ASSIN) de 2016, nas tarefas de similaridade semântica e reconhecimento de inferência textual. O sistema INESC-ID@ASSIN aborda o problema de detectar similaridade entre frases como uma tarefa de regressão e aborda a inferência textual como uma tarefa de classificação. Embora o INESC-ID@ASSIN seja baseado essencialmente em características lexicais simples para detecção de paráfrases e reconhecimento de inferência textual, foram obtidos resultados promissores.

Palavras chave

aprendizagem supervisionada, regressão, classificação

1 Introduction

Detecting the amount and type of equivalence between two sentences is a complex Natural Language Understanding (NLU) task, mostly due to the lexical and syntactic variability allowed by human natural languages. Detecting sentence equivalence may comprise semantic similarity, which supports entailment and paraphrase identification.

Entailment can be defined as a relationship between two natural language units (e.g., between two sentences) where the truth of one requires the truth of the other. We can say that a sentence A entails a sentence B if and only if whenever A is true, B is also true.

Paraphrases are a special type of entailment, namely bidirectional entailment. A paraphrase is a kind of semantic equivalence responsible for the interconnection of statements, by replacing grammatical classes and variables unchanged between lexical and syntactic structures.

Recognizing Textual Entailment (RTE) and semantic similarity calculation have many practical applications, such as question answering, information extraction, summarization and Machine Translation (MT).

In this paper we present INESC-ID@ASSIN, a system that detects paraphrases and textual entailment, based on supervised machine learning and that leverages lexical features denoting relatedness among two sentences. Detecting the amount of similarity is achieved with a regression model while the type of entailment is predicted with a classifier.

We evaluated our approach on ASSIN

(Avaliação de Similaridade Semântica e Inferência Textual), a shared task from PROPOR (International Conference on the Computational Processing of Portuguese). ASSIN provides datasets with examples in European (PT-PT) and Brazilian (PT-BR) Portuguese.

The rest of this paper is organized as follows: Section 2 presents related work, Section 3 presents the INESC-ID@ASSIN system and Section 4 details the evaluation procedure and results. Section 5 concludes and points to future work.

2 Related work

The availability of shared tasks focused on the problem of RTE has fostered the experimentation with a number of data-driven approaches applied to semantics (Dagan et al., 2009; Dagan et al., 2013; Zhao, Zhu, and Lan, 2014; Bjerva et al., 2014). Specifically, the availability of RTE datasets for supervised training made it possible to formulate the problem as a classification task, where features are extracted from the training examples and then used by machine learning algorithms in order to build a classifier, which is finally applied to the test data to classify each pair of sentences/phrases as either entailed or not.

Most recent approaches to RTE or paraphrase identification use machine learning algorithms (e.g., linear classifiers) with a variety of features, including lexical, syntactic and semantic matching features, based on document co-occurrence counts, first-order syntactic rewrite rules, and based on extracting the information gain provided by lexical measures.

Different approaches have been produced along the years with some sort of combination of the features described above. A simple approach is the bag-of-words strategy, in which the comparison of a given sentence pair is calculated using a cosine similarity score. If the score is greater than a threshold value (determined manually or learned through a supervised training data), the sentences are classified as paraphrases.

Zhang and Patrick (2005) proposed a classification method where the sentence pair is simplified into canonical forms (through a set of rules) such as changing sentences from passive to active voice. Using a decision tree learning method, the authors exploit lexical matching features such as the edit distance between the tokens.

In addition to lexical matching features, authors like Kozareva and Montoyo (2006) or Ul-Qayyum and Wasif (2012) proposed classification approaches using a combination of lexical and semantic features and heuristics (e.g., negation patterns) to aid in the detection of false paraphrases.

Methods used in most previous approaches work at the sentence level, but as paraphrases regularly involve synonyms or other forms of word relatedness, authors like Mihalcea, Corley, and Strapparava (2006) or Fernando and Stevenson (2008) developed word-level similarity methods to determine if a sentence is paraphrase of another sentence. These methods are based in word-to-word similarity measures (e.g., knowledge-based metrics which use WordNet). Methods based in alignments (such as summarization and MT metrics) are also commonly used.

Madnani, Tetreault, and Chodorow (2012) proposed an approach based on string alignment metrics from the field of MT. Although the use of MT metrics for the task of paraphrase identification is not novel (Finch, Hwang, and Sumita, 2005), the authors merit from a thorough reassessment of these metrics conjointly with the creation of new metrics in order to achieve the best results so far on the well-known Microsoft Research Paraphrase Corpus (Dolan, Quirk, and Brockett, 2004).

Pakray, Bandyopadhyay, and Gelbukh (2011) describe a lexical and syntactic approach for solving the RTE problem. This method results from the composition of several modules, namely a pre-processing module, a lexical similarity module and a syntactic similarity module.

Tsuchida and Ishikawa (2011) proposed an RTE system that uses machine learning methods with features based on lexical and predicate-argument structure level information. The underlying idea is to identify the text-hypothesis pairs that have a high entailment score but are in fact not entailed, i.e., false-positive pairs classified by the system's lexical-level module can latter be rejected by the sentence-level module.

It is important to notice that previous work typically takes advantage of methods that are language independent using simple strategies such as counting n-grams. Most of the described RTE approaches also conclude that the lexical modules achieve better results than syntactic and sentence structure modules.

3 INESC-ID@ASSIN

The models created by INESC-ID@ASSIN were based on multiple similarity metrics. Several previous studies, within the area of Natural Language Processing (NLP) and also in other fields, have already used similar methods for combin-

ing multiple similarity metrics in the context of accessing the similarity between objects (Martins, 2011; Madnani, Tetreault, and Chodorow, 2012). The features used in INESC-ID@ASSIN are explained in the following sections and further detailed in (Marques, 2015). A Support Vector Machine (SVM) was used for classification (RTE and paraphrase identification) and a Kernel Ridge Regression (KRR) was used for deriving continuous values (similarity grading). We used the SVM/KRR implementation from the scikit-learn Python toolkit [1].

3.1 String similarity

The string similarity features considered in INESC-ID@ASSIN are:

1. **Longest Common Subsequence**. The size of the Longest Common Subsequence (LCS) between the text and the hypothesis. The value is clamped between 0 and 1 by dividing the size of the LCS by the size of the sentence with the longer length.
2. **Edit Distance**. The minimum edit distance between the tokens/words from the text and the hypothesis.
3. **Length**. The absolute difference in length (number of tokens/words) between the text and the hypothesis. Also the maximum and minimum length are considered separately as features.
4. **Cosine Similarity**. The cosine similarity between the text and hypothesis, with basis on the number of occurrences of each word in the text/hypothesis (the term frequency representation). The cosine formula is shown in Equation 1.

$$\text{sim}(d_1, d_2) = \frac{\overrightarrow{V}(d_1) \cdot \overrightarrow{V}(d_2)}{||\overrightarrow{V}(d_1)|| \times ||\overrightarrow{V}(d_2)||} \quad (1)$$

   The returned value is a real number between 0 and 1. The higher the value, the more identical is the text-hypothesis pair.
5. **Jaccard Similarity**. The Jaccard similarity between the text and the hypothesis. The returned value is a real number between 0 and 1, where 1 means equal, and 0 totally different. The Jaccard similarity coefficient is used for comparing the similarity and diversity of sample sets. It measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. The Jaccard similarity between two sets of words A and B is thus defined as follows:

$$\text{J(A, B)} = \frac{|\text{A} \cap \text{B}|}{|\text{A} \cup \text{B}|} \quad (2)$$

6. **Soft TF-IDF**. The Soft-TF-IDF similarity metric measures similarity between vector-based representations of the sentences, but considering an internal similarity metric for finding equivalent words. The Jaro-Winkler similarity metric between words with a threshold of 0.9, is used as the internal similarity metric. The Jaro distance d_j of two given strings s_1 and s_2 is:

$$d_j = \begin{cases} 0 & \text{if m} = 0 \\ \frac{1}{3} \times (\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m}) & \text{otherwise} \end{cases} \quad (3)$$

In the equation, m is the number of matching characters and t is half the number of transpositions. The Jaro-Winkler measure modifies the Jaro measure by adding more weight to a common prefix. This enhancement introduces 2 parameters: (1) PL, expressed as the length of the longest common prefix between the two strings, and (2) PW, the weight to give the prefix.

$$\text{JaroWinkler}(x, y) = \\ (1 - PL \times PW) \times \text{jaro}(x, y) + PL \times PW \quad (4)$$

3.2 RTE features

The features inspired on previous RTE studies are:

1. **NE Overlap**. The Jaccard similarity taking into consideration only named entities. For simplicity, named entities are all words containing capital letters.
2. **NEG Overlap**. The Jaccard similarity taking into consideration only negative words. The negative words are: *não*, *nunca*, *jamais*, *nada*, *nenhum*, *ninguém*.
3. **MODAL Overlap**. The Jaccard similarity taking into consideration only modal words. The modal words are: *podia*, *poderia*, *dever*, *deve*, *devia*, *deverá*, *deveria*, *faria*, *possivel*, *possibilidade*, *possa*.

[1] `http://scikit-learn.org/`

3.3 Paraphrase features

The features inspired on previous studies focusing on paraphrase identification are:

1. **BLEU**. This MT metric is computed as the amount of n-gram overlap, for different values of n, between two sentences, tempered by a length penalty (Papineni et al., 2002). We employed a maximum n gram order of 3, for coverage of short sentences, as its suggested in (Papineni et al., 2002) that it yields similar performance compared to the classic 4-gram (BLEU-4).
2. **METEOR**. This metric is a variation of BLEU based on the harmonic mean of unigram precision and recall, with recall weighted higher than precision (Banerjee and Lavie, 2005).
3. **TER**. The Translation Error Rate is an extension of the Word Error Rate (WER), which is a simple metric based on dynamic programming that is defined as the number of edits needed to transform one string into another. TER includes a heuristic algorithm to deal with shifts in addition to insertions, deletions and substitutions (Snover et al., 2006).
4. **NCD**. The Normalized Compression Distance is a general way of measuring the similarity between two objects (Li et al., 2004). The underlying idea is that if you compress two strings s_1 and s_2 only the overlapping information is extracted.
5. **ROUGE-N**. N-gram overlap based on co-occurrence statistics (Lin and Hovy, 2003).
6. **ROUGE-L**. A variation of the ROUGE metric based on the length of longest common subsequence (Lin and Och, 2004).
7. **ROUGE-S**. A variation of the ROUGE metric based on skip-bigrams (i.e., bi-grams of word tokens, allowing for in-between words) (Lin and Och, 2004).

3.4 Numeric features

The idea behind the numeric features is simple: sentences that refer to the same entities but with different numbers are likely to be contradictory. We used a simple numeric feature that results from the multiplication of 2 Jaccard similarities. One between the numeric characters in the pair text-hypothesis, and a second between the surrounding words of such numeric characters. The result is a real value between 0 and 1, where 0 indicates a contradictory statement.

3.5 Text representations

The previously described features are applied to different representations of the sentences. We specifically considered the following representations:

1. **Original tokens**.
2. **Lowercased tokens**.
3. **Stemmed of lowercase tokens**.
4. **Word clusters**. The Brown word clustering algorithm is an agglomerative bottom-up method that aggregates words in a binary tree of classes (Turian, Ratinov, and Bengio, 2010), through a criterion based on the log-probability of a text under a class-based language model. Much like a Hidden Markov Model, given a cluster membership indicators t_i for the tokens w_i in a text, the probability of the w_i given w_{i-1} is given by Equation 5.

$$\begin{aligned} &\text{argmax}_{T\in\gamma}\text{P}(W|T)\times\text{P}(T)\approx \\ &\text{argmax}_{T\in\gamma}\prod_{i=1}^{n}\text{P}(w_i|t_i)\text{P}(t_i|t_{i-1}) \end{aligned} \quad (5)$$

   The Brown clustering procedure was applied to a collection of newswire documents from a Portuguese newspaper called *Público*, which resulted in 1001 clusters.
5. **Double Metaphone**. We used a well known algorithm to phonetically encode the words in the sentences, reducing words to a combination of 12 consonant sounds. The Double Metaphone algorithm (Philips, 1990) is however based on English pronunciation, being more adequate to encode English words and foreign words often heard in the United States.
6. **Character trigrams**. Trigrams are a special case of the concept of n-gram, where n is 3. The character trigrams are used as key terms in a representation of the phrase much as words are used as key terms to represent a document.

Our model combines features along with these different representations giving a total of 96 features. Notice that some features are not suitable to be combined with some representations, such

Feature	O	L	S	C	DM	T
LCS	X	X	X	X	X	
Edit Distance	X	X	X	X	X	
Cosine Similarity	X	X	X	X	X	X
Abs Length	X	X	X	X	X	
Max Length	X	X	X	X	X	
Min Length	X	X	X	X	X	
Jaccard	X	X	X	X	X	X
Soft TF-IDF	X	X	X			
NE Overlap	X	X	X	X	X	X
NEG Overlap	X	X	X	X	X	X
Modal Overlap	X	X	X	X	X	X
BLEU-3	X	X	X	X	X	
METEOR	X	X	X	X	X	
ROUGE N	X	X	X	X	X	
ROUGE L	X	X	X	X	X	
ROUGE S	X	X	X	X	X	
TER	X	X	X	X	X	
NCD	X	X	X	X	X	
Numeric	X	X	X			

Table 1: Combination of features with representations, where O, L, S, C, DM and T correspond to Original, Lowercased, Stemmed, Cluster, Double Metaphone and Trigrams, respectively.

as the numeric feature with the double metaphone representation. The combinations can be seen in Table 1.

4 Evaluation

INESC-ID@ASSIN was evaluated on the ASSIN dataset to assess its performance on the task of automatically measuring semantic relatedness and type of textual entailment.

We report results from 2 distinct setups, one with SVM and KRR configured to use a polynomial kernel and the other with SVM and Ridge Regression configured to use a linear kernel. For the linear models, the most informative features are also reported.

Each experiment includes results for 3 different runs, on both tasks and for Portuguese and Brazilian test data.

Additionally, we also measured the performance when training our algorithm with one Portuguese variety and testing with the other.

All the runs employ the same algorithm on distinct datasets. One of such datasets corresponds to expanding the ASSIN dataset with the results of using MT over the SICK corpus (Marelli et al., 2014), while the remaining runs use partitions of the original ASSIN dataset.

4.1 Task description

The ASSIN dataset contains 10000 sentence pairs collected from Google News, split into training and test sets with an equal number of Portuguese and Brazilian examples in each set. Each pair is annotated for both semantic relatedness and textual entailment.

Semantic relatedness is a continuous value from 1 to 5, according to the following guidelines:

1. Completely different sentences, on different subjects;
2. Sentences are not related, but are roughly on the same subject;
3. Sentences are somewhat related. They may describe different facts but share some details;
4. Sentences are strongly related, but some details differ;
5. Sentences mean essentially the same thing.

Textual entailment is a categorical assignment to the classes of entailment, paraphrase or none.

Within ASSIN, 2 tasks are available to automatically calculate semantic relatedness and label textual entailment. Performance is also measured separately for Portuguese and Brazilian.

4.2 Training with more data

We experimented with the urge of MT methods for expanding the original ASSIN dataset with new sentences taken from an existing English dataset, as more data typically lead to better results.

The SICK dataset (Marelli et al., 2014) is very similar to that from ASSIN, in size and annotated information. However, it is based on image and video captions obtained by crowdsourcing, therefore featuring less language variability but more similarity among pairs.

SICK was translated to Portuguese, using a Python wrapper over the Microsoft Bing translation service, and merged with the European and Brazilian train datasets. We added 9191 examples from SICK to the 6000 examples from the ASSIN training set, for one of our runs.

4.3 Results

Our approach to the ASSIN task was evaluated with the Pearson coefficient and Mean Squared Error (MSE) metrics for semantic similarity, and with Accuracy and F1 for RTE.

We considered 3 different runs of our approach, which differ in the employed amount of training data, namely:

1. PT-PT or PT-BR: train only with the same Portuguese sample (European or Brazilian, respectively) of the test (3000 samples).
2. PT: merge datasets of both languages for training, regardless of the intended test (6000 samples).
3. PT+BingSICK: use the full Portuguese dataset and the translated SICK dataset for training (15191 samples, 9191 from SICK).

These runs were evaluated over the European and Brazilian test sets, although on the official submission we only reported assessments for the European test set. On the official submission, PT with polynomial kernels was our best run (on the European test set). However, due to a software problem (now solved) the officially reported values were lower than those shown in Table 2.

Results for our approach to the ASSIN task with polynomial kernels are shown in Tables 2 and 3.

	Similarity		RTE	
Training	Pearson	MSE	Accuracy	F1
PT-PT	0.74	0.60	83.55%	0.68
PT	0.74	0.60	83.95%	0.69
PT+BingSICK	0.72	0.68	80.70%	0.59

Table 2: Evaluation results, with a polynomial kernel and all features — European test set.

	Similarity		RTE	
Training	Pearson	MSE	Accuracy	F1
PT-BR	0.73	0.36	85.45%	0.64
PT	0.73	0.36	85.70%	0.66
PT+BingSICK	0.70	0.40	84.30%	0.58

Table 3: Evaluation results, with a polynomial kernel and all features — Brazilian test set.

Results for our approach to the ASSIN task with linear kernels are shown in Tables 4 and 5.

Here, performance with linear kernels is similar to that of polynomial kernels, but the advantage of the higher dimensionality of the polynomial kernel space is highlighted when more data exists, as can be seen in the higher performance drop of the linear models when using the expanded MT dataset (particularly in MSE and F1) and compared to the polynomial results.

	Similarity		RTE	
Training	Pearson	MSE	Accuracy	F1
PT-PT	0.73	0.62	84.90%	0.71
PT	0.74	0.61	84.05%	0.68
PT+BingSICK	0.70	0.73	77.10%	0.47

Table 4: Evaluation results, with a linear kernel and all features — European test set.

	Similarity		RTE	
Training	Pearson	MSE	Accuracy	F1
PT-BR	0.73	0.36	85.35%	0.55
PT	0.73	0.36	85.85%	0.66
PT+BingSICK	0.70	0.42	82.60%	0.46

Table 5: Evaluation results, with a linear kernel and all features — Brazilian test set.

From these results, we conclude that adding more quality training data (hand selected and verified) may slightly improve performance, while adding more training data unfiltered (repetitive and with lexical or syntactic errors from MT) yields a lower performance.

Comparing the results by table, the configuration that most consistently yielded the best results is PT, both for RTE and similarity grading. Considering all tables, our system performs better on Brazilian inputs.

We have also conducted experiments to understand the performance of the models trained with one Portuguese variety while predicting on the other Portuguese variety. As shown in Table 6, understanding a Portuguese variety while only knowing the other is better than using the Bing translated SICK dataset. For simplicity, only is shown the experiment for polynomial kernels, but linear kernels were also evaluated and yielded similar results.

	Similarity		RTE	
Training	Pearson	MSE	Accuracy	F1
PT-BR	0.73	0.63	82.70%	0.64
PT-PT	0.72	0.37	84.30%	0.66

Table 6: Varying the train set employed for testing with the other/remaining Portuguese variety, with a polynomial kernel and all features.

4.4 Analysis of best features

We employed the Recursive Feature Elimination method, as implemented in scikit-learn, for re-

trieval of the best 10 features on the PT setup (which yielded the best results), for each task (RTE and similarity grading).

This is a greedy method for feature selection based on feature weights. As scikit-learn only provides feature weights from linear models, and we only employ feature selection in our linear models.

The top 10 features (order is irrelevant) for RTE are:

- Soft TF-IDF, on original tokens
- Jaccard, on Double Metaphone
- Jaccard, on stemmed of lowercase tokens
- Absolute Length, on Double Metaphone
- LCS, on stemmed of lowercase tokens
- Numeric, on original tokens
- NE Overlap, on Double Metaphone
- ROUGE-N, on original tokens
- ROUGE-L, on stemmed of lowercase tokens
- TER, on stemmed of lowercase tokens

And the top 10 for similarity grading are:

- Cosine Similarity, on original tokens
- Soft TF-IDF, on original tokens
- Jaccard, on Double Metaphone
- Jaccard, on stemmed of lowercase tokens
- Jaccard, on character trigrams
- Numeric, on stemmed of lowercase tokens
- NE Overlap, on Double Metaphone
- ROUGE-N, on original tokens
- ROUGE-N, on word clusters
- ROUGE-S, on stemmed of lowercase tokens

5 Conclusion and future work

This work addressed RTE and similarity grading by applying several features based on previous work for RTE and paraphrase identification - mostly machine translation and summarization metrics. These features, together with string similarity and numeric features, represent a novel approach that shies away from the most recent line of work of the area, which mostly focused on building systems based on semantic alignment and binary relations matching.

In future work, we will compare INESC-ID@ASSIN's performance with that of the same machine learning algorithms applied to complex features based on rich syntactic/semantic structures and knowledge sources.

Acknowledgments

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) through the project with reference UID/CEC/50021/2013, through the international project RAGE with reference H2020-ICT-2014-1/644187 and through the project LAW-TRAIN with reference H2020-EU.3.7. – 653587.

References

Banerjee, Satanjeev and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*.

Bjerva, Johannes, Johan Bos, Rob van der Goot, and Malvina Nissim. 2014. The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2014)*, pages 642–646, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Dagan, I., B. Dolan, B. Magnini, and D. Roth. 2009. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(04).

Dagan, Ido, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. 2013. Recognizing textual entailment: Models and applications. *Synthesis Lectures on Human Language Technologies*, 6(4):1–220.

Dolan, Bill, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the International Conference on Computational Linguistics*.

Fernando, Samuel and Mark Stevenson. 2008. A semantic similarity approach to paraphrase

detection. In *Proceedings of the Annual Research Colloquium on Computational Linguistics in the UK*.

Finch, Andrew, Young-Sook Hwang, and Eiichiro Sumita. 2005. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Proceedings of the International Workshop on Paraphrasing*.

Kozareva, Zornitsa and Andres Montoyo. 2006. Paraphrase identification on the basis of supervised machine learning techniques. In *Proceedings of the International Conference on Advances in Natural Language Processing*.

Li, Ming, Xin Chen, Xin Li, Bin Ma, and Paul Vitányi. 2004. The similarity metric. *Information Theory, IEEE Transactions on*, 50(12).

Lin, Chin-Yew and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*.

Lin, Chin-Yew and Franz Josef Och. 2004. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*.

Madnani, Nitin, Joel Tetreault, and Martin Chodorow. 2012. Re-examining machine translation metrics for paraphrase identification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

Marelli, Marco, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC-2014), Reykjavik, Iceland, May 26-31, 2014.* (Marelli et al., 2014), pages 216–223.

Marques, Ricardo. 2015. Detecting contradictions in news quotations. Master's thesis, IST, University of Lisbon, November.

Martins, Bruno. 2011. A supervised machine learning approach for duplicate detection over gazetteer records. pages 34–51, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mihalcea, Rada, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the National Conference on Artificial Intelligence*.

Pakray, Partha, Sivaji Bandyopadhyay, and Alexander Gelbukh. 2011. Textual entailment using lexical and syntatic similarity. *Internacional Journal of Artificial Intelligence and Applications*, 2(1).

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*.

Philips, L. 1990. Hanging on the metaphone. *Computer Language Magazine*, 7(12).

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the Conference of the Association for Machine Translation in the Americas*.

Tsuchida, Masaaki and Kai Ishikawa. 2011. A method for recognizing textual entailment using lexical-level and sentence structure-level features. *In Proceedings of the Text Analysis Conference*.

Turian, Joseph, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Ul-Qayyum, Zia and Altaf Wasif. 2012. Paraphrase identification using semantic heuristic features. *Research Journal of Applied Sciences, Engineering and Tecnhology*, 4(22).

Zhang, Yitao and Jon Patrick. 2005. Paraphrase identification by text canonicalization. In *Proceedings of the Australasian Language Technology Workshop*.

Zhao, Jiang, Tiantian Zhu, and Man Lan. 2014. Ecnu: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2014)*, pages 271–277, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.