Combining Lexical Information with Machine Learning for Answer Validation at QA@CLEF 2007

M.Á. García-Cumbreras, J.M. Perea-Ortega, F. Martínez-Santiago, and L. Alfonso Ureña-López

SINAI Research Group, Computer Science Department, University of Jaén, Spain
{magc,jmperea,dofer,laurena}@ujaen.es

Abstract. This document contains the description of the experiments carried out by the SINAI group. We have developed an approach based on several lexical measures integrated by means of different machine learning models. Based on lexical features it obtains a 41% of accuracy in answer validation for the Question-Answering task.

1 Introduction

This document contains the description of the experiments carried out by the SINAI group[1] at the AVE subtask of QA@CLEF 2007 [1], using English as target language. We have developed an approach based on several lexical measures integrated by means of different machine learning models. More precisely, we have evaluated three features based on lexical similarity. In order to calculate the semantic distance between a pair of tokens (stems), we have tried several measures based on Lin's similarity measure [2]. In spite of the relatively straightforward approach we have obtained a remarkable accuracy.

2 Approach Description

We have developed a system based on Machine Learning (ML) methods, which makes use of a binary classifier to solve the answer validation. We can distinguish between two processes: training and classification.

The data was given by triples (question, exact answer and supporting text passage). We used the question and the exact answer in our experiments.

In the training process we have extracted several features for each training collection[2]. Previous results have been evaluated using the existing entailment judgements of these collections, and Machine Learning parameters have been adjusted.

[1] http://sinai.ujaen.es

[2] Answer Validation Exercise training collection and Third Recognizing Textual Entailment Challenge (RTE3) training and set collections.

C. Peters et al. (Eds.): CLEF 2007, LNCS 5152, pp. 381–386, 2008.

We have trained the classifier obtaining a *learned model* which will be used later in the classification process.

In the classification process we extract the same features used in the training process for each pair question-answer. The classification algorithm uses these features and the *learned model* obtained in the training process and returns a boolean value (*correct* or *incorrect*) for each pair question-answer. Figure 1 describes the system architecture.

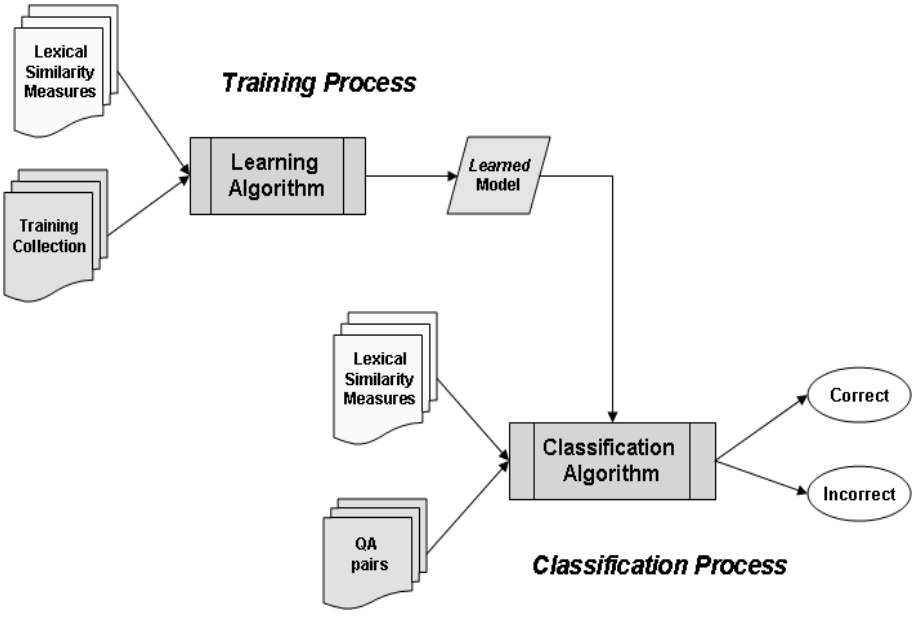

Fig. 1. System architecture

The extracted features are related to the lexical similarity. In our experiments we have applied four different lexical similarity measures, which are explained below.

2.1 Lexical Similarity

This experiment approaches the answer validation task, based on the extraction of a set of lexical measures, that check the existing similarity between the hypothesis-text pairs. Our approach is similar to [3] but the matching between pairs of words is relaxed by using the Lin's similarity measure [2] through Wordnet hierarchy. More concisely, we have applied simple matching, Binary Matching and Consecutive Subsequence Matching. In this task we have considered the answers as hypotheses and questions as texts.

Before the calculation of the different measures, the first step was to preprocess the pairs using the English *stopwords* list and the Porter stemmer available in

GATE[3]. In this step we also obtain the Part Of Speech (POS) of each token using GATE.

After that, we have applied four different measures or techniques:

- **Simple Matching:** this technique calculates the semantic distance between the stems of each question and its answer. If the distance exceeds a threshold, both stems are considered similar and the similarity weight value increases in one. The accumulated weight is normalized dividing it by the number of terms of the answer (hypothesis). In this experiment we have considered the threshold 0.5. The values of semantic distance measure range from 0 to 1. In order to calculate the semantic distance between two stems, we have tried several measures based on WordNet [4]. **Lin's similarity measure** [2] was shown to be best overall measures. It uses the notion of information content and the same elements as Jiang and Conrath's approach [5] but in a different fashion:

$$sim_L(c_1, c_2) = \frac{2 \times \log p(lso(c_1, c_2))}{\log p(c_1) + \log p(c_2)}$$

  where c_1 and c_2 are *synsets*, *lso(c_1,c_2)* is the information content of their lowest super-ordinate (most specific common subsumer) and *p(c)* is the probability of encountering an instance of a *synset* c in a specific corpus like the Brown Corpus of American English[6].

  The Simple Matching technique is defined in the following equation:

$$SIM_{matching} = \frac{\sum_{i \in H} similarity(i)}{|H|}$$

  where H is the set that contains the elements of the answer (hypothesis) and *similarity(i)* is defined like:

$$similarity(i) = \begin{cases} 1 \text{ if } \exists j \in T sim_L(i,j) > 0.5 \\ 0 \text{ otherwise} \end{cases}$$

- **Binary Matching:** this measure is the same that the previous one but modifying the *similarity* function:

$$similarity(i) = \begin{cases} 1 & \text{if } \exists j \in T \ i = j \\ 0 & \text{otherwise} \end{cases}$$

- **Consecutive Subsequence Matching:** this technique relies on forming subsequences of consecutive stems in the answer (hypothesis) and matching them in the question (text). The minimal size of the consecutive subsequences is two, and the maximum is the maximum size of the answer. Every correct matching increases in one the final weight. The sum of the obtained weights of the matching between subsequences of a certain size or length is normalized by the number of sets of consecutive subsequences of the answer created for this length. These weights are accumulated and normalized

[3] http://gate.ac.uk/

by the size of the answer less one. The Consecutive Subsequence Matching technique is defined in the following equations:

$$CSS_{matching} = \frac{\sum_{i=2}^{|H|} f(SH_i)}{|H| - 1}$$

where SH_i is the set that contains the subsequences of the answer with i size or length and *f(SH$_i$)* is defined like:

$$f(SH_i) = \frac{\sum_{j \in SH_i} matching(j)}{|H| - i + 1}$$

where

$$matching(i) = \begin{cases} 1 & \text{if } \exists k \in ST_i \ k = j \\ 0 & \text{otherwise} \end{cases}$$

where ST_i represents the set that contains the subsequences with i size from question (text).

– **Trigrams:** this technique relies on forming trigrams of words in the answer and matching them in the question. If a answer trigram matches in question, then the similarity weight value increases in one. The accumulated weight is normalized dividing it by the number of trigrams of the answer.

In order to obtain the results of our experiments we have used two CPAN[4] Perl modules: the *Wordnet::Similarity* and the *Wordnet::QueryData*. We have employed the *Wordnet::QueryData* Perl module for getting the *synsets* of each *(stem,POS)* pair from the text and the hypothesis. Then, we have used the *Wordnet::Similarity* module for computing the semantic relatedness of two word senses, using the information content based measure described by Lin[2].

3 Experiments and Results

The algorithms used in the experiments as binary classifiers are two, namely, Bayesian Logistic Regression (BBR)[7] and TiMBL [8]. Both algorithms have been trained with the development data provided by the organization of the Pascal challenge (RTE-3) and the AVE task of CLEF.

As it has been explained in previous sections, a model is generated via the supervised learning process. This model is used by the classification algorithm, which will decide whether an answer is entailed by the given snippet or not.

Table 1 shows two official results and two non official, where:

– **Exp1** uses three lexical similarities (*SIMmatching* + *CSSmatching* + *Trigrams*). The model has been trained using the development data provided by the organization of the Pascal challenge, RTE-3. The ML method used was BBR.

[4] http://www.cpan.org

- **Exp2** uses the same three features. The model has been trained using the development data provided by the organization of the Answer Validation Exercise task, AVE-2007, and the development data provided by the organization of the Pascal challenge, RTE-3. The ML method used was TiMBL.
- **Exp3** (non-official) uses the same three features. The model has been trained using the development data provided by the organization of the Answer Validation Exercise task, AVE-2007, and the development data provided by the organization of the Pascal challenge, RTE-3. The ML method used was BBR.
- **Exp4** (non-official) uses the same three features. The model has been trained using the development data provided by the organization of the Pascal challenge, RTE-3. The ML method used was TiMBL.

Table 1. Results with TiMBL and BBR classifiers

Experiment	Classifier	Train Data	F measure	Qa accuracy
Exp1	BBR	RTE-3	0.19	0.08
Exp2	TimBL	RTE-3 and AVE-2007	0.37	0.41
Exp3 (non-official)	BBR	RTE-3 and AVE-2007	0.17	0.08
Exp4 (non-official)	TimBL	RTE-3	0.25	0.32

As we expected, the best result is obtained by means of the use of both development collections, RTE-3 and AVE-2007, and the ML method TiMBL. TiMBL has been used in some classification experiments, obtaining better results than BBR [9].

4 Conclusions and Future Work

In spite of the simplicity of the approach, we have obtained remarkable results: each set of features has reported relevant information, concerning the entailment judgement determination. Our experiments approach the textual entailment task being based on the extraction of a set of lexical measures which show the existing similarity between the hypothesis-text pairs.

We have applied Simple Matching, Binary Matching, Consecutive Subsequence Matching and Trigrams, but the matching between pairs of words is relaxed by using the Lin's similarity measure through Wordnet hierarchy.

Finally, we want to implement a hierarchical architecture based on constraint satisfaction networks. The constraints will be given by the set of available features and the maintenance of the integrity according to the semantic of the phrase.

Acknowledgments

This work has been partially supported by a grant from the Spanish Government, project TIMOM (TIN2006-15265-C06-03), and the RFC/PP2006/Id_514 granted by the University of Jaén.

References

1. Di Nunzio, G.M., Ferro, N., Mandl, T., Peters, C.: CLEF 2007 Ad Hoc Track Overview. In: Peters, C., et al. (eds.) CLEF 2007. LNCS, vol. 5152. Springer, Heidelberg (2007)
2. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning (1998)
3. Ferrandez., O., Micolo, D., Muñoz, R., Palomar, M.: Técnicas léxico-sintácticas para reconocimiento de implicación textual. Tecnologías de la Información Multilingüe y Multimodal (in press, 2007)
4. Budanitsky, A., Hirst, G.: Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures (2001)
5. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: Proceedings of International Conference on Research in Computational Linguistics (1997)
6. Resnik, P.: Using information content to evaluate semantic similarity. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (1995)
7. Genkin, A., Lewis, D.D., Madigan, D.: BBR: Bayesian logistic regression software. Center for Discrete Mathematics and Theoretical Computer Science, Rutgers University (2005), `http://www.stat.rutgers.edu/~madigan/bbr/`
8. Daelemans, W., Zavrel, J., van der Sloot, K., van den Bosch, A.: TiMBL: Tilburg Memory Based Learner, version 1.0, Reference Guide. ILK Technical Report (1998)
9. García Cumbreras, M.A., Ureña López, A., Martínez Santiago, F.: BRUJA: Question Classification for Spanish. Using Machine Translation and an English Classifier. In: Proceedings of the MLQA 2006 (2006)