A Lexical–Semantic Approach to AVE

Óscar Ferrández, Rafael Muñoz, and Manuel Palomar
Natural Language Processing and Information Systems Group
Department of Computing Languages and Systems
University of Alicante
San Vicente del Raspeig, Alicante 03690, Spain
{ofe, rafael, mpalomar}@dlsi.ua.es

Abstract

This paper discusses a system capable of detecting when answers for specific questions are supported by snippets, all provided by Question Answering (QA) systems. This task is known as the Answer Validation Exercise (AVE) track within the Cross–language Evaluation Forum (CLEF). The system uses a set of regular expressions in order to join the question and the answer into an affirmative sentence and afterwards applies several lexical–semantic inferences to attempt to detect whether the meaning of this sentence can be inferred by the meaning of the supporting text. Throughout the paper we present and discuss the different system components together with the results obtained. Moreover, we want to apply special emphasis to the language–independent capabilities of some of them. As a result, we are able to apply our techniques over both Spanish and English corpora.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval

General Terms

Algorithms, Semantic Similarity, Experimentation, Measurement, Performance

Keywords

Question Answering, Answer Validation, Recognizing Textual Entailment

1 Introduction

The three–year–old Cross–Language Evaluation Forum (CLEF) track, the Answer Validation Exercise (AVE), provides an evaluation framework to consider appropriately those answers that are supported by the question and the passage from which they were extracted. This kind of inference will help Question Answering (QA) systems to increase their performance as well as humans in the assessment of QA system output.

Traditionally the approaches destined for validating the answers of QA systems have always been inspired by textual entailment recognition techniques [13, 14]. Moreover, simple techniques based on word overlapping and shallow lexical inferences (e.g. *linear distance*) have obtained competitive results [6] being considered as a suitable starting point for further research.

The system described in this paper integrates several inferences from different knowledge sources. The base of the system consists of lexical deductions without any semantic knowledge,

afterwards several modules have been added to the system in order to compute more sophisticated deductions (e.g. WordNet relations, named entities correspondences and relations between verbs).

The paper is structured as follows. The next section presents our approach for our participation in Answer Validation Exercise (AVE). Third section illustrates the experiments carried out and the results obtained. Finally, fourth section shows the conclusions and proposes future work based on our actual research.

2 Validating the Answers

Aimed at achieving an approach that obtains promising results in a short lapse of time, we built a system that uses a reduced number of external resources which would compromise the system's speed. The system is able to detect unidirectional meaning relations between affirmative sentences formed by the question and the answer and the supporting texts supplied by QA systems.

Figure 1 depicts the architecture of the system illustrating its modules and stages during the inference meanings process between the texts.

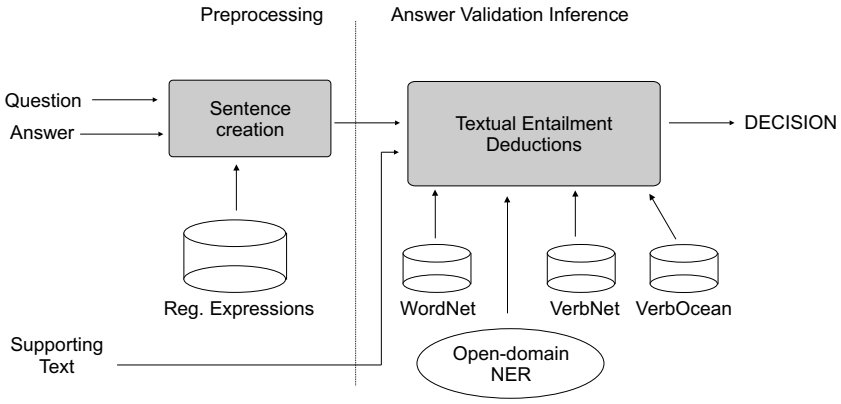

Figure 1: System's Architecture.

The process of validating the answers involves two main phases: (i) the preprocessing stage which is responsible for building an affirmative sentence merging the question and the answer by means of a set of regular expressions, and (ii) the pure textual entailment component that detects lexical–semantic inferences between a pair of texts.

2.1 Preprocessing

Each query and answer provided within both the development and test corpora were preprocessed in order to obtain an affirmative well–formed sentence (this sentence will be called *hypothesis*, or simply *H*, in order to follow the textual entailment methodology and terminology first proposed in [3]). For this purpose, an extension of the set of regular expressions proposed in our previous participation in AVE [4] was carried out. This extension was done by analysing the kinds of questions exposed in the development corpus and integrating new regular expressions capable of managing the whole set of questions. For both the development and test set every question is controlled by one regular expression, however it does not imply that the output affirmative sentence is grammatically well–formed. Obviously, it will depend on the correctness of the answer.

2.2 The Textual Entailment Component

In order to tackle the AVE task, first we have created a base system making use of well–know techniques based on lexical inferences. These techniques have already been used successfully by some research (including ourselves) in the task of recognising textual entailment relations

[5, 11, 1]. Later on adjusting the system to the idiosyncrasies of the AVE task, we have generated some constraints that the pair of texts (hypothesis–supporting text) involved within the meanings' inference must fulfil.

2.3 The Base

Its performance is supported by the computation of a wide variety of lexical measures over the lemmas of the tokens that make up the texts. Thus, prior to the calculation of the measures, all texts are tokenized, lemmatized and morphologically analysed.

From the whole set of measures, we select those that are more significant according to the information gain that they provide to a machine learning classifier. Therefore, a Bayesian Net classifier implemented in Weka [20] was used for this issue, considering each measure as a feature. Next, the group of the most meaningful measures that composes the feature set is listed[1]:

- **Levenshtein distance**: the function $match(i)$ is calculated for each item of the hypothesis (H) as:

$$match(i) = \begin{cases} 1 & \text{if } \exists j \in TLv(i,j) = 0, \\ 0.9 & \text{if } \nexists j \in TLv(i,j) = 0 \wedge \\ & \exists k \in TLv(i,k) = 1, \\ \max\left(\frac{1}{Lv(i,j)} \forall j \in T\right) & \text{otherwise.} \end{cases} \quad (1)$$

  where $Lvd(i,j)$ represents the Levenshtein distance between i and j. The cost of an insertion, deletion or substitution is equal to one, and the weight assigned to $match(i)$ when $Lvd(i,j) = 1$ has been obtained empirically.

- **Needleman-Wunsch algorithm [15]**: similar to the basic Levenshtein distance but adding a variable cost adjustment to the cost of an insertion or deletion. Some experiments were done in order to adjust the cost of a gap being a penalty of 3 the best value.

- **Smith-Waterman algorithm**: is a well–known dynamic programming algorithm for performing local sequence alignment and determining similar regions between sequences. The algorithm was first proposed by [18] and consists of two steps: (i) calculate the similarity matrix score; and (ii) according to the dynamic programming method, trace back the similarity matrix to search for the optimal alignment.

  For two sequences SQ_1 and SQ_2, the optimal alignment score of two sub–sequence $SQ_1[1] \ldots SQ_1[i]$ and $SQ_2[1] \ldots SQ_2[j]$ is the calculation of $D(i,j)$ defined as:

$$D(i,j) = max \begin{cases} 0 & \text{start over,} \\ D(i-1,j-1) - f(SQ_1[j], SQ_2[j]) & \text{substitution or copy,} \\ D(i-1,j) - GAP & \text{insertion,} \\ D(i,j-1) - GAP & \text{deletion.} \end{cases} \quad (2)$$

  It permits two adjustable parameters regarding substitutions and copies for an alphabet mapping (the f function) and also allows costs to be attributed to a GAP for insertions or deletions. In our experiments we empirically set the values 0.3, -1 and 2 for a gap, copy and substitution respectively.

[1] For some measures we use their implementation provided by the SimMetrics library (http://www.dcs.shef.ac.uk/∼sam/simmetrics.html)

- **Jaro distance**: this metric comes from the work presented in [8] and measures the similarity between two strings taking into account spelling derivations. The following equation describes the way that it obtains the similarities:

$$d_j(s_1, s_2) = \frac{m}{3 \cdot |s_1|} + \frac{m}{3 \cdot |s_2|} + \frac{m - t}{3 \cdot m} \tag{3}$$

being s_1 and s_2 the strings to be compared, $|s_1|$ and $|s_2|$ their respective lengths, m the number of matching characters considering only those are not further than $[\frac{\max(|s_1|,|s_2|)}{2}] - 1$ and t the number of transpositions computed as the number of matching (but different) characters divided by two.

- **Euclidean distance**: The traditional definition measures the distance between two points $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_n)$ in Euclidean n-space as:

$$\sqrt{(p_1 - q_1)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \tag{4}$$

With the aim of dealing with strings, we set n as the number of distinct items in any of the two strings and p_i, q_i the times that each of them appears in each string respectively.

- **Jaccard similarity coefficient**: is a statistic coefficient for comparing the similarity and diversity of sample sets. It is defined as the size of the intersection divided by the size of the union of the sample sets:

$$J(A, B) = |A \cap B| / |A \cup B| \tag{5}$$

In our case, we compute this coefficient representing each string as a Jaccard vector. This metric was first introduced and detailed in [7].

- **Dice's coefficient**: for sets X and Y of items extracted from the two strings to be processed, the coefficient is defined as:

$$D = \frac{2|X \cap Y|}{|X| + |Y|} \tag{6}$$

- **Cosine similarity**: is a common vector–based similarity. The input strings are transformed into vector space and it is computed as follows:

$$\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{||\vec{x}|| \cdot ||\vec{y}||} \tag{7}$$

- **IDF specificity**: we determine the specificity of a word using the inverse document frequency (IDF) introduced in [19], which is defined as the total number of documents in the corpus divided by the total number of documents that include that word. In our experiments, we derive the documents frequencies from the document collections used for the tracks reported within the Cross–Language Evaluation Forum (CLEF) [16], in concrete the LA Times 94 and Glasgow Herald 95 collections, which contain a total number of 169,477 documents. The IDF measure helps to the system to valuate each word regarding its specificity whereby the words with higher IDF values will be more relevant to take the entailment decision.

- **JWSL**: in order to discover word meaning relations that are not able to be detected directly from orthographic derivations we exploit the lexical–semantic resource called WordNet [12]. Relations such as synonymy, hypernyms, and semantic paths that connect two concepts can be found exploiting its taxonomy. Also, there are many implementations of similarity and relatedness measures between words based on WordNet. In our experiments, we have used the Java WordNet Similarity Library (JWSL[2]), which implements some of the most commons semantic similarity measures. This feature automatically derives a score (the maximum score obtained from all similarity measures implemented in JWSL) that shows the similarity degree between the nouns, verbs and adjectives of two texts.

Other measures that were considered but later discarded due to the fact that they introduced noise to the system were: bi– and tri–grams of letters, Block distance, SoundEx distance.

2.4 The Constraints

In addition to the aforementioned inferences, we considered very appealing the idea of integrating into the system some constraints that could support the final decision in most cases.

- **The Named Entities**: it is based on the detection, presence and absence of Named Entities (NEs). Despite the previous measures taken into account every token, even entities, these measures do not detect the importance of the presence or absence of an entity (e.g. when there is an entity in the hypothesis but the same entity is not present in the supporting text). This idea comes from the work presented in [17], where the authors successfully build their system only using the knowledge supplied by the recognition of NEs. In our case, we establish the following constraint: "*In order to be considered as a candidate entailment pair, the hypothesis' entities must also appear within the supporting text*" This constraint is prior to the launching of the similarity measures, so only pairs containing the same entities will be considered.

  In our experiments, we use NERUA system [10], an open domain NE recognizer which was trained by the corpus provided in CoNNL-2002 Share Task[3] and CoNLL-2003 Share Task[4] in order to recognise Spanish and English entities respectively.

- **The Verbs**: the other important particles in a sentence, apart from the NEs, are the verbs. Therefore, if we are able to detect whether the hypothesis' verbs are related to the supporting text's verbs, we could set another constraint showing this relatedness. To do this, we created two wrappers in Java for the VerbNet [9] and VerbOcean [2] resources. These wrappers allow us to detect semantic relationships between verbs.

  Therefore, if every verb in the hypothesis (auxiliar verbs are not considered) can be related to one or more verbs in the supporting text, the pair will successfully pass this constraint. Two verbs are related whether: (i) they have the same lemma or are synonyms considering WordNet, (ii) they belong to the same VerbNet class or a subclass of their classes, and (iii) there is a relation in VerbOcean[5] that connects them.

Consequently, if the candidate pair pass the two previous constraints, it will be processed by the measures presented in 2.3. It will be carried out for both the development and test corpora. The development corpus will be used as training for a Bayesian Net classifier.

Another way we considered in order to integrate these constraints into the system, was to add new features that indicate the matching coefficient between the entities and verbs according to the previous resources and strategy. Unfortunately, the addition of these new features into the classifier did not produce any improvement in the results. Furthermore, considering these inferences as previous constraints the corpus as well as the processing time are strongly reduced.

[2] http://grid.deis.unical.it/similarity/
[3] http://www.cnts.ua.ac.be/conll2002/ner/
[4] http://www.cnts.ua.ac.be/conll2003/ner/
[5] The VerbOcean's relations considered are: similarity, strength and happens–before.

3 Experiments and Results

We set several experiments[6] according to the inferences presented in the previous sections of the paper:

- System Base (SB): comprises the basic measures shown in 2.3 together with the JWSL inference based on WordNet.
- SB+Entities Constraint (SB+EntC): adds to SB the constraint about the detection, presence and absence of NEs.
- SB+EntC+Verbs Constraint (SB+EntC+VerbC): develops all the previous inferences including the constraint deduced by the relationships between verbs.

Table 1 shows the different experiments carried out and the results obtained by the system over the English corpora. The proposed baselines were those provided by the AVE organizers:

- Baseline100: it was generated setting all pairs as VALIDATED and randomly choosing the SELECTED values.
- Baseline50: 50% of the pairs were tagged as VALIDATED (no SELECTED values).

In order to achieve the best system training configuration, we made several combinations of the development corpora available for both this edition and previous ones. The one that reached the best results (in a 10–cross fold validation test) was joining the development corpora of the last and current edition (AVE'07 and AVE'08, respectively).

Corpus	**Run**	**Prec. YES**	**Rec. YES**	**F-measure**	**QA acc.**	**estim. QA**
Dev.	SB	0.279	0.843	0.42	–	–
	SB+EntC	0.311	0.776	0.444	–	–
	SB+EntC+VerbC	0.307	0.748	0.436	–	–
Test	Baseline100	0.08	1	0.14	0.09	0.09
	Baseline50	0.08	0.5	0.13	–	–
	SB	0.23	0.92	0.37	0.19	0.23
	SB+EntC	0.35	0.86	0.49	0.19	0.27
	SB+EntC+VerbC	0.35	0.78	0.48	0.19	0.28

Table 1: English results obtained for the AVE 2008 track.

The results point out that a significant improvement is reached when the system considers the constraint about the NEs' inference. Unfortunately, although the constraint related to the verb's relationships considerably reduced the size of the corpus and consequently the processing time, it did not report any improvement except for the *estimated QA performance*. It reveals that complex treatment of verbs should be carried out, and the coverage of the resources used should be extended by means of other complementary knowledge sources (e.g. inferences about semantic frames rather than to only consider the verbs would improve these kinds of deductions).

Although the system makes use of language–dependent resources, its base as well as the NE recognizer components are language independent. It allowed us to apply the system over the Spanish corpora. However, this time just two experiments could be done:

- Spanish System Base (SB_es): implements all measures presented in 2.3 except the one that uses WordNet[7].

[6]Some results presented in this section are not official due to the fact that some experiments were carried out after the deadline.

[7]This is owing to JWSL works with English WordNet, and at present we do not have any implementation of these measures for the Spanish WordNet.

- SB_es+Entities Constraint (SB+EntC_es): adds to SB_es the constraint about NEs, but in this case using the Spanish configuration of NERUA.

Table 2 draws the results obtained over the Spanish corpora. The system behaviour is somewhat similar. The entities constraint improves the system's performance and the system base configuration proves that it is a very good starting point for further language–independent research.

Corpus	Run	Prec. YES	Rec. YES	F-measure	QA acc.	estim. QA
Development	SB_es	0.372	0.655	0.474	–	–
	SB+EntC_es	0.418	0.603	0.494	–	–
Test	Baseline100	0.10	1	0.18	0.11	0.11
	Baseline50	0.10	0.5	0.17	–	–
	SB	0.26	0.76	0.38	0.32	0.37
	SB+EntC	0.32	0.67	0.44	0.27	0.33

Table 2: Spanish results obtained for the AVE 2008 track.

Finally, we should like to mention how the system establishes the SELECTED value. Since our system returns a numeric value to determine the validation of the answers, we decided to mark as SELECTED the pair with the highest positive score among all pairs that belong to the same question. In the event that two or more pairs have the highest score, then one of them is randomly chosen and tagged as SELECTED value.

4 Conclusions and Future Work

This paper describes a system capable of validating the answer for a given question according to a snippet that supposedly supports the answer. Moreover, we present a basic configuration of the system and afterwards we add some constraints in order to enrich the knowledge and improve the results of the system. Also, the language–independent capabilities of some system's components are clearly exposed with the application of them over Spanish and English.

Future work can be related to the improvement in the treatment of verbs as well as the detection of NEs. For instance, some heuristics regarding semantic verb frames could help the system to extend the coverage of verb's relationships. Regarding the NE recognizer, currently we only detect a strict matching between the hypothesis and supporting text entities and whether an entity is contained by another. However, there are pairs in the corpora that contain the same entity expressed in different manners/words, and when it occurs the NE recognizer is unable to detect an inference between them (e.g. when an entity is inferred by its acronym). Therefore, subsequent work will be characterized by identifying deeper inference relations between entities such as acronyms, date expansion, etc.

Acknowledgements

This research has been partially funded by the QALL-ME consortium, which is a 6^{th} Framework Research Programme of the European Union (EU), contract number FP6-IST-033860 and by the Spanish Government under the project CICyT number TIN2006-1526-C06-01.

References

[1] Rod Adams, Gabriel Nicolae, Cristina Nicolae, and Sanda Harabagiu. Textual entailment through extended lexical overlap and lexico-semantic matching. In *Proceedings of the ACL-*

PASCAL Workshop on Textual Entailment and Paraphrasing, pages 119–124, Prague, June 2007. Association for Computational Linguistics.

[2] Timothy Chklovski and Patrick Pantel. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, Barcelona, Spain, 2004.

[3] Ido Dagan and Oren Glickman. Probabilistic textual entailment: Generic appied modelling of language variability. In *Proceedings of the PASCAL Workshop on Learning Methods for Text Understanding and Mining*, Grenoble, France, 2004.

[4] Óscar Ferrández, Daniel Micol, Rafael Muñoz, and Manuel Palomar. The contribution of the university of alicante to ave 2007. In *Working Notes of the CLEF 2007 Workshop*, Budapest, Hungary, September 2007.

[5] Óscar Ferrández, Daniel Micol, Rafael Muñoz, and Manuel Palomar. A perspective-based approach for solving textual entailment recognition. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 66–71, Prague, June 2007. Association for Computational Linguistics.

[6] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[7] Paul Jaccard. The distribution of the flora in the alpine zone. *New Phytologist*, 11(2):37–50, 1912.

[8] Matthew A. Jaro. Probabilistic linkage of large public health data file. *Statistics in Medicine*, 14:491–498, 1995.

[9] Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. Extending verbnet with novel verb classes. In *Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, Genova, Italy, June 2006.

[10] Z. Kozareva, Ó. Ferrández, A. Montoyo, and R. Muñoz. Combining data-driven systems for improving named entity recognition. *Data and Knowledge Engineering*, 61(3):449–466, 2007.

[11] Prodromos Malakasiotis and Ion Androutsopoulos. Learning textual entailment using svms and string similarity measures. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 42–47, Prague, June 2007. Association for Computational Linguistics.

[12] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J. Miller. Introduction to WordNet: An On-line Lexical Database. *International Journal of Lexicography*, 3(4):235–244, 1990.

[13] Anselmo Pe nas, Álvaro Rodrigo, and Felisa Verdejo. Overview of the answer validation exercise 2006. In C. Peters et al., editor, *CLEF 2006, Lecture Notes in Computer Science LNCS 4730*, Alicante, Spain, September 2006.

[14] Anselmo Pe nas, Álvaro Rodrigo, and Felisa Verdejo. Overview of the answer validation exercise 2007. In C. Peters et al., editor, *CLEF 2007, Lecture Notes in Computer Science LNCS 5152*, Budapest, Hungary, September 2007.

[15] Saul Needleman and Christian Wunsch. A general method applicable to the search for similarities in amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, 1970.

[16] Carol Peters. What happened in clef 2007? introduction to the working notes. In *Working Notes for the 8th Workshop of the Cross-Language Evaluation Forum, CLEF*, Budapest, Hungary, September 2007.

[17] Álvaro Rodrigo, Anselmo Peñas, and Felisa Verdejo. UNED at Answer Validation Exercise 2007. In *Working Notes of the CLEF 2007 Workshop*, Budapest, Hungary, September 2007.

[18] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[19] Karen Sparck-Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

[20] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools and techniques*. 2nd Edition, Morgan Kaufmann, San Francisco, 2005.