INAOE at QA@CLEF 2008: Evaluating Answer Validation in Spanish Question Answering

Alberto Téllez-Valero, Antonio Juárez-González, Manuel Montes-y-Gómez, Luis Villaseñor-Pineda
Laboratorio de Tecnologías del Lenguaje
Instituto Nacional de Astrofísica, Óptica y Electrónica
Luis Enrrique Erro no. 1, Sta. María Tonantzintla, Pue.; 72840; Mexico
{albertotellezv, antjug, mmontesg, villasen}@inaoep.mx

Abstract

This paper introduces the new INAOE's answer validation method. This method is based on supervised learning approach that uses a set of attributes that capture some lexical-syntactic relations among the question, the answer and the given support text. In addition, the paper describes the evaluation of the proposed method at both the Spanish Answer validation Exercise (AVE 2008) and the Spanish Question Answering Main Task (QA 2008). The evaluation objectives were twofold. One the one hand, evaluate the ability of our answer validation method to discriminate correct from incorrect answers, and on the other hand, measure the impact of including an answer validation module in our QA system. The evaluation results were encouraging; the proposed method achieved a 0.39 F-measure in the detection of correct answers, outperforming the baseline result of the AVE 2008 task by more than 100%. It also enhanced the performance of our QA system, showing a gain in accuracy of 22% for answering factoid questions. Furthermore, when there were evaluated three candidate answers per question, the answer validation method allowed increasing the MRR of our QA system by 40%, reaching a MRR of 0.28.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [**Database Managment**]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Question Answering, Answer Validation, Textual Entailment, Machine Learning

1 Introduction

Latest evaluations of question answering (QA) systems evidenced two important facts about the state of the art of this field. First, they indicated that it already does not exist any system capable of answering all types of questions with similar precision rates, and second, they revealed

that most current QA systems are complementary (see for instance the Spanish QA evaluation overview at CLEF 2005 [1]). These two facts have triggered the development of answer validation (AV) methods, which allow determining if a specified answer is correct and supported [2].

In line with these recent efforts, in this paper we describe a new AV method. This method, similar to our previous year work [3], is based on a supervised learning approach for recognizing the textual entailment. It mainly uses a set of attributes that capture some simple relations among the question, the answer and the given supported text. In particular, it considers some novel attributes that characterize: (*i*) the compatibility between question and answer types; (*ii*) the redundancy of answers across streams; and (*iii*) the overlap (as well as the non-overlap) between the question-answer pair and the core fragment of the support text.

In order to evaluate the proposed method we considered two different scenarios: the Answer Validation Exercise (AVE 2008) and the Question Answering Main Task (QA 2008). The objective of the first scenario was to evaluate the ability of our AV method to discriminate correct from incorrect answers as well as its capacity to combine the answers from several QA systems. In contrast, the goal of the second evaluation scenario was to measure the impact of including an answer validation module in our QA system [4].

The evaluation results were encouraging; the proposed method achieved a 0.39 F-measure in the detection of correct answers, outperforming the baseline result of the AVE 2008 task by more than 100%. It also enhanced the performance of our QA system, producing a gain in accuracy of 22% for answering factoid questions. Furthermore, when there were evaluated three candidate answers per question, the answer validation method allowed increasing the MRR of our QA system by 40%, reaching a MRR of 0.28.

The rest of the paper is organized as follows. Section 2 describes our answer validation method. Section 3 presents the evaluation results of the proposed method in both the Answer Validation Exercise and Spanish QA Main Task. Finally, Section 4 exposes our conclusions and outlines some future work directions.

2 The Answer Validation Method

Given a question (Q), a candidate answer (A) and a support text (S), this method returns a confidence value (β) that allows deciding whether to accept or reject a candidate answer. In other words, it helps to determine if the specified answer is correct and if it can be deduced from the given support text.

Like other previous answer validation methods evaluated in the last AVE [5, 6], and following the original idea proposed in the first RTE challenge at PASCAL [7], our method is mainly based on recognizing the textual entailment (RTE) between the support text (T) and an affirmative sentence (H) called hypothesis, created from the combination of the question and the answer[1].

The returned confidence value β is generated by means of a supervised learning approach that considers three main processes: preprocessing, attribute extraction and answer classification. The following sections describe each of these processes.

2.1 Preprocessing

The objective of this process is to extract the main content elements from the question, answer and support text, which will be subsequently used for deciding about the correctness of the answer. This process considers two basic tasks: on the one hand, the identification of the main constituents from the question-answer pair, and on the other hand, the detection of the core fragment of the support text as well as the consequent elimination of the unnecessary information.

[1] The entailment between the pair (T, H) occurs when the meaning of H can be inferred from the meaning of T.

2.1.1 Constituent Identification

We detect three basic constituents from the questions: its main action, the action actors, and if exist, the action restriction. As an example, consider the question in Table 1. In this case, the action is represented by the verb `invade`, its actors are the syntagms `Which country` and `Iraq`, and the action restriction is described by the propositional syntagma `in 1990`.

Table 1: Example of excessive support text to accept or reject an answer

Question:	`Which country did Iraq invade in 1990?`
Candidate answer:	`Kuwait`
Support text:	`Kuwait was a close ally of Iraq during the Iraq-Iran war and functioned as the country's major port once Basra was shut down by the fighting. However, after the war ended, the friendly relations between the two neighboring Arab countries turned sour due to several economic and diplomatic reasons which finally culminated in an Iraqi invasion of Kuwait.`

In order to detect the question constituents we firstly apply a shallow parsing to the given question[2]. Then, from the resulting syntactic tree (Q_{parsed}), we construct a new representation of the question (called *Q'*) by detecting and tagging the following elements:

1. *The action constituent.* It corresponds to the syntagm in Q_{parsed} that includes the main verb.
2. *The restriction constituent.* It is represented by the prepositional syntagm in Q_{parsed} having at least one explicit time expression (e.g., `in 1990`), or including a preposition such as `after` or `before`.
3. *The actors constituents.* These constituents are formed by the rest of the elements in Q_{parsed}. It is commonly divided in two parts. The first one, henceforth called *hidden actor constituent*, corresponds to the syntagm that includes the interrogative word and it is generally located at the left of the action constituent. The second part, which we call the *visible actor constituent*, is formed by the rest of the syntagms, generally located at the right of the action constituent.

Finally, we also consider an *answer constituent*, which is simply the lemmatized candidate answer (denoted by *A'*).

2.1.2 Support Text's Core Fragment Detection

Commonly, the support text is a short paragraph —of maximum 700 bytes according to CLEF evaluations— which provides the context necessary to support the correctness of a given answer. However, in many cases, it contains more information than required, damaging the performance of RTE methods based on lexical-syntactic overlaps. For instance, the example of Table 1 shows that only a part of the last sentence in the support text (i.e., `Iraqi invasion of Kuwait`) is useful for validating the given answer, whereas the rest of the text only contribute to produce an irrelevant overlap (e.g., `Kuwait was a close ally of Iraq`).

In order to reduce the support text to the minimum useful text fragment according to the candidate answer validation, we proceed as follows:

- First, we apply a shallow parsing to the support text, obtaining the syntactic tree (S_{parsed}).

[2]In all the text processing used in our method (i.e., lemmatization, part of speech tag, named entities recognition and classification, and shallow parsing), we employed the open source tool called Freeling (http://garraf.epsevg.upc.es/freeling/).

- Second, we match the content terms (nouns, verbs, adjectives and adverbs) from the question constituents against the terms from S_{parsed}. In order to avoid some minimal writing differences of the same concept not solved by the morphological analysis (e.g., `Iraq` against `Irak` or `Iraqi`), we compare the terms using the Levenshtein edition distance[3]. Mainly, we consider that two different words are equal if their distance value is less than 0.4.

- Third, based on the number of matched terms, we align the question constituents with the syntagms from the support text.

- Forth, we match the answer constituent against the syntactic tree (S_{parsed}). The idea is to find all occurrences of the answer in the given support text.

- Fifth, we determine the minimum context of the answer in the support text that contains all matched syntagms. This minimum context (represented by a sequence of words around the answer) is what we call the *core fragment* (denoted by *T'*). In the case that the support text includes several occurrences of the answer, we select the one with the smallest context.

Applying the procedure described above we determine that the core fragment of the support text showed at Table 1 is `in an Iraqi invasion of Kuwait`.

2.2 Attribute Extraction

This stage gathers a set of processes that allow extracting several attributes from the question, the answer and the support text. These attributes can be categorized in two different groups: the attributes that indicate the relation between the question and the answer, and the attributes that measure the entailment relation between the question-answer pair and the support text.

The following sections describe both kinds of attributes and explain the way they are calculated from *Q'*, *A'* and *T'*.

2.2.1 Attributes about the Question-Answer Relation

Question Characteristics

We consider four different attributes from the question: the question word (what, how, where, etc.), the question category (factoid or definition), the expected answer type (date, quantity, name or other), and the type of question restriction (date, period, event, or none).

The question word, question category, and the expected answer type are determined using a set of simple lexical patterns. Some of these patterns are showed below. It can be observed that each of them includes information about the question category and the expected answer type.

(`WHAT` *OR* `WHO`) `is` [*whatever*]	→	DEFINITION – OTHER
`HOW many` [*whatever*]	→	FACTOID – QUANTITY
`WHEN` [*whatever*]	→	FACTOID – DATE

On the other hand, the value of the question restriction (date, period, event or none) depends on the form of the restriction constituent. If this constituent contains only one time expression, then this value is set to "date". In the case the restriction constituent includes two time expressions, it is set to "period". If the restriction constituent does not include any time expression, then the question restriction is defined as "event". Finally, when the question does not have any restriction constituent, the value of the question restriction is set to "none".

[3]The Levenshtein edition distance has been previously used in other works related to answer validation in Spanish language, see for instance [8].

Question-Answer Compatibility

This attribute indicates if the question and answer types are compatible. The idea of this attribute is to capture the situation where the semantic class of the evaluated answer does not correspond to the expected answer type. For instance, having the answer `yesterday` for the question `How many inhabitants are there in Longyearbyen?`.

This is a binary attribute: it is equal to 1 when the answer corresponds to the expected answer type, and it is equal to 0 if this correspondence does not exist.

Answer Redundancy

Taking into account the idea of "considering candidates as allies rather than competitors" [9], we decided to include an attribute related to the occurrence of the answers across the pool of candidate answers.

Different from other redundancy methods (like the one present in [10]) that directly uses the frequency of occurrence of the answers, the proposed attribute indicates the sum of the edition distances between the actual evaluated answer to each one of the rest of the candidate answers.

The edition distance strategy allows dealing with the great language variability and also with the presence of some typing errors. In this way, an answer `X` contributes to the redundancy rate of another answer `Y` and vice versa, even though `X` and `Y` are not exactly the same (e.g. `spacial telescope` and `Hubble telescope`).

2.2.2 Attributes related to the Textual Entailment Recognition

The attributes of this category are of two main types: (*i*) attributes that measure the overlap between the support text and the hypothesis (an affirmative sentence formed by combining the question and the answer); and (*ii*) attributes that denote the differences (non-overlap) between these two components.

It is important to explain that, different from other RTE methods, we do not use the complete support text, instead we only use its core fragment *T'*. In addition, we neither need to construct an hypothesis text, instead we use as hypothesis the set of question-answer constituents (the union of *Q'* and *A'*, which we call *H'*).

Overlap Characteristics

These attributes express the degree of overlap —in number of words— between *T'* and *H'*. In particular, we compute an overlap attribute for each one of the fourth types of content terms (nouns, verbs, adjectives and adverbs) as well as for each one of the six types of named entities (names of persons, places, organizations, and other things, as well as dates and quantities). We generate these ten different overlap attributes for each one of the five constituents in *H'* (the action constituent, the restriction constituent, the hidden actor constituent, the visible actor constituent, and the answer constituent). In this way, we get a total of fifty attributes that represent the overlap characteristics.

Similar to the calculation of the answer redundancy attribute, in this case we also apply the edition distance to evaluate the overlap between the terms of *H'* and *T'*.

Non-Overlap Characteristics

These attributes indicate the number of non-overlapped terms from the core fragment of the support text, that is, they indicate the number of terms from *T'* that are not present in any of the detected constituents. Mainly, we measure the non-overlap between the answer constituent and each one of the other constituents, and compute this non-overlap for each type of content term as well as for each type of named entity. In total we generate forty different non-overlap attributes.

2.3 Answer Classification

This final process generates the answer validation decision by means of a supervised learning approach. In particular, it applies a boosting ensemble formed by ten decision tree classifiers[4].

The constructed classifier decides whether to accept or reject the candidate answer based on the ninety-six attributes described in the previous section. In addition, it also generates a validation confidence (β) that indicates how reliable is the given answer in accordance to the support text.

3 Experimental Evaluation

As we previously mentioned, we evaluated the proposed AV method in two different scenarios: the Answer Validation Exercise (AVE 2008) and the Question Answering Main Task (QA 2008). The objective of the first scenario was to evaluate the ability of the AV method to discriminate correct from incorrect answers as well as its capacity to combine the answers from several QA systems. In contrast, the goal of the second evaluation scenario was to measure the impact of including an answer validation module in our QA system [4]. The following sections present the results from both scenarios.

3.1 Training and Test Sets

Table 2 resumes the used training and test sets. The training set combines the instances from the training set of the AVE 2006 and the instances from the test sets of the AVE 2006 and 2007. In total, it contains 574 questions with 2905 answers (the first row of the table details this set).

On the other hand, we consider two different test sets, one for the Spanish Answer Validation Exercise (AVE) and other for the Spanish Question Answering Main Task (QA). The AVE test set consists of 1528 answers; these answers correspond to 136 different questions and were generated by all participating systems at the 2008 Spanish QA task. Whereas, the QA test set includes 1152 answers returned by our QA system. These answers correspond to 164 questions that our system catalogue as not NIL from the entire 2008 test set.

Table 2: Training and Test Sets (VALIDATED are the answers judged as *right*, REJECTED are the answers judged as *wrong* or *unsupported*, and UNKNOWN are the answers judged as *inexact*)

	VALIDATED	REJECTED	UNKNOWN
Training set	1436 (25%)	4306 (75%)	0
AVE test set	153 (10%)	1354 (89%)	21 (1%)
QA test set	74 (6%)	1066 (93%)	12 (1%)

3.2 Results

3.2.1 Spanish Answer Validation Exercise

This evaluation exercise focuses on analyzing two different aspects of the AV methods: on the one hand, their ability to discriminate correct from incorrect answers (*answer validation evaluation*), and on the other hand, their capacity to select the correct answer from a pool of candidate answers returned by diverse QA systems (*stream fusion evaluation*).

From previous experiments [11], we noticed that the best method for AV (for discriminating correct from incorrect answers) is not necessarily the best option for a QA stream fusion, it taking into account that the actual AV methods are far away of the perfect validation. Based on this evidence, we decided to evaluate two different runs obtained by applying two different acceptance thresholds over the confidence value (β). The first run (RUN 1) aimed to increase the recall by

[4]We used the Weka implementations for the boosting (AdaBoostM1) and C4.5 (J48) algorithms (http://www.cs.waikato.ac.nz/ml/weka/).

reducing in 10% the default acceptance threshold, whereas the second run (RUN 2) maintained the default threshold ($\beta = 0.5$).

Table 3 shows the answer validation results corresponding to our two submitted runs. It also shows (in the last row) the results for a 100% VALIDATED baseline (i.e., an answer validation system that accepted all given answers). The results indicate that reducing the acceptance threshold (RUN 1) our method achieved a high recall but a low precision, which means that it correctly accepts must correct answers (there are a few false negatives), but it also incorrectly accepts many wrong responses (there are several false positives). In contrast, the second run (RUN 2) got a worst recall, but achieved a major precision and F-measure, outperforming the baseline result in more than 100%.

Table 3: Results for the answer validation evaluation

	Precision	Recall	F-measure
RUN 1	0.13	0.86	0.23
RUN 2	0.30	0.59	0.39
100% VALIDATED	0.10	1.0	0.18

Complementary to the previous data, Table 4 shows the evaluation results for the QA stream fusion. These results indicate that the QA-accuracy of RUN 1 is 19% better than the accuracy of RUN 2. Given that RUN 2 clearly outperformed the answer validation result of RUN 1 (see Table 3), these results confirm our observation that the best answer validation method not necessary produces the best QA stream fusion performance.

Besides the traditional QA-accuracy measure, this year the AVE organizers included a new evaluation measure called QA-performance. This measure allows evaluating the influence into the question answering task of not only correctly accept those right answers but also to correctly reject those wrong ones. The results in Table 4 indicate that, because of the better capacity of RUN 2 to reject wrong answers, the QA-performance of both runs were very similar.

Table 4: Results for the QA stream fusion evaluation

	QA-accuracy	QA-performance
RUN 1	0.32	0.34
RUN 2	0.27	0.33
PERFECT FUSION	0.62	0.85

Finally, it is important to comment that, in order to understand the behavior of the proposed method, we carried out a deep analysis of the usefulness of each one of the used characteristics. Table 5 resumes the information gain values of the main kinds of attributes used by our supervised AV method. Surprisingly, these values show that the proposed non-overlap characteristics are more discriminative than the traditional overlap features.

Table 5: Information gain of the used characteristics (For the overlap and non-overlap characteristics, the showed value indicates the average of the complete subset of attributes)

Answer redundancy	Overlap characteristics	Non-overlap characteristics
0.023	0.002	0.012

3.2.2 Spanish QA Main Task

In order to evaluate the effect of including an AV module in a QA system, this year we submitted two different runs at the Main QA task. The first run (inao081eses) was the original output of our

QA system (refer to [4] for details), whereas the second run (inao082eses) was the result of applied the AV method over the set of candidate answers generated by the first run. Table 6 shows the evaluation results of both runs as well as a baseline result corresponding to a perfect validation of the output of our QA system.

Table 6: Results of the QA main task (Also the Accuracy (Ac) the table presents, by question type (factoid and definition), the number of questions answered right (R), wrong (W), inexact (X), and unsupported (U))

	Factoid questions				Definition questions				
	R	W	X	U	R	W	X	U	Ac
inaoe081eses (original QA system)	23	156	1	1	19	0	0	0	0.21
inaoe082eses (QA system with an AV module)	28	149	3	1	16	3	0	0	0.22
PERFECT VALIDATION	30	147	3	1	19	0	0	0	0.25

Results from Table 6 indicate that the AV module helped increasing the number of right answers for factoid questions, improving the accuracy of our QA system by 22%. In contrast, the AV module damaged the treatment of definition question since it incorrectly rejected three right answers. In this case, taking into account that our QA system is very accurate for answering definition questions, our conclusion is that it is better not to include the AV module.

Given that this year was allowed to deliver three candidate answers per question, we not only evaluated the effect of the AV module in the answer accuracy but also in rank of the correct answers. For achieving this objective, we included more that one answer for some questions; the first run (inaoe081eses) consisted of 422 answers, whereas the second run (inaoe082eses) included a total of 343 answers. The difference in the number of answers between both runs was caused because many candidate answers were rejected in the second run during the validation process. It is important to mention that the AV module not only eliminated some *tentative incorrect* answers, but also modified their final order. In particular, the rank of the answers in the second run was determined by means of their confidence values. The evaluation results indicated that the second run outperformed by 40% (with a Mean Reciprocal Rank of 0.28) the output of the first run (with a Mean Reciprocal Rank of 0.20).

4 Conclusions

This paper presented a new AV method based on a supervised textual entailment approach. This method mainly differs from previous ones in the kind of used attributes. In particular, it considers some *novel* attributes that characterize: (*i*) the compatibility between question and answer types; (*ii*) the redundancy of answers across streams; and (*iii*) the overlap as well as the non-overlap between the question-answer pair and the core fragment of the support text. Regarding these attributes, it is important to mention that an analysis about their usefulness showed that the proposed non-overlap characteristics are more discriminative than the traditional overlap features.

The proposed method was evaluated in two different scenarios: the Spanish Answer Validation Exercise (AVE 2008) and the Spanish QA Main Task (QA 2008). The objective of the first scenario was to evaluate the ability of our AV method to discriminate correct from incorrect answers as well as its capacity to combine the answers from several QA systems. In contrast, the goal of the second evaluation scenario was to measure the impact of including an answer validation module in a QA system. The evaluation results were encouraging; the proposed method achieved a 0.39 F-measure in the detection of correct answers, outperforming the baseline result of the AVE 2008 task by more than 100%. It also enhanced the performance of our Spanish QA system, producing a gain in accuracy of 22% for factoid questions. Furthermore, when there were evaluated three candidate answers per question, the AV method allowed increasing the MRR of our QA system by 40%, reaching a MRR of 0.28.

Finally, it is important to comment that this year our best results in the AVE (a F-measure of 0.39 and a qa-accuracy of 0.32) were very distant from those corresponding to a perfect validation.

We presume that this situation was caused by the decreasing number of right answers together with the increasing number of relevant support passages related to the wrong answers. In order to tackle these problems, and based on the fact that non-overlap attributes were the most discriminative, we plan to include more elements (such as prepositions, conjunctions, and some punctuation marks) for their computation.

Acknowledgments

This work was done under partial support of CONACYT (project grants 43990 and 61335, and scholarships 171610 and 165499). We also want to thank CLEF organizers for the provided resources.

References

[1] Vallin, A., Magnini, B., Giampiccolo, D., Aunimo, L., Ayache, C., Osenova, P., Peñas, A., de Rijke, M., Sacaleanu, B., Santos, D., Sutcliffe, R.F.E.: Overview of the clef 2005 multilingual question answering track. In Peters, C., Gey, F.C., Gonzalo, J., Müller, H., Jones, G.J.F., Kluck, M., Magnini, B., de Rijke, M., eds.: CLEF. Volume 4022 of Lecture Notes in Computer Science., Springer (2005) 307–331

[2] Peñas, A., Rodrigo, Á., Sama, V., Verdejo, F.: Testing the reasoning for question answering validation. Journal of Logic and Computation (3) (December 2007)

[3] Téllez-Valero, A., Montes-y-Gómez, M., Villaseñor-Pineda, L.: INAOE at AVE 2007: Experiments in spanish answer validation. In: Working Notes for the CLEF 2007 Workshop, Budapest, Hungary (September 2007)

[4] Téllez, A., Juárez, A., Hernández, G., Denicia, C., Villatoro, E., Montes, M., Villaseñor, L.: INAOE's participation at QA@CLEF 2007. In: Working Notes for the CLEF 2007, Budapest, Hungary (September 2007)

[5] Peñas, A., Rodrigo, Á., Sama, V., Verdejo, F.: Overview of the answer validation exercise 2006. [12] 257–264

[6] Peñas, A., Rodrigo, Á., Verdejo, F.: Overview of the answer validation exercise 2007. In: Working Notes for the CLEF 2007 Workshop, Budapest, Hungary (September 2007)

[7] Dagan, I., Magnini, B., Glickman, O.: The PASCAL recognising textual entailment challenge. In: Proceedings of Pascal Challenge Workshop on Recognizing Textual Entailment, Southampton, UK (April 2005) 1–8

[8] Rodrigo, Á., Peñas, A., Herrera, J., Verdejo, F.: The effect of entity recognition on answer validation. [12] 483–489

[9] Dalmas, T., Webber, B.L.: Answer comparison in automated question answering. J. Applied Logic **5**(1) (2007) 104–120

[10] Roussinov, D., Chau, M., Filatova, E., Robles-Flores, J.A.: Building on redundancy: Factoid question answering, robust retrieval and the "other". In: Proceedings of the Thirteenth Text REtreival Conference (TREC 2005). (2005) 15–18

[11] Téllez-Valero, A., Montes-y-Gómez, M., Villaseñor-Pineda, L., Peñas, A.: Improving question answering by combining multiple systems via answer validation. In Gelbukh, A.F., ed.: CICLing. Volume 4919 of Lecture Notes in Computer Science., Springer (2008) 544–554

[12] Peters, C., Clough, P., Gey, F.C., Karlgren, J., Magnini, B., Oard, D.W., de Rijke, M., Stempfhuber, M., eds.: Evaluation of Multilingual and Multi-modal Information Retrieval, 7th Workshop of the Cross-Language Evaluation Forum, CLEF 2006, Alicante, Spain, September 20-22, 2006, Revised Selected Papers. In Peters, C., Clough, P., Gey, F.C., Karlgren, J., Magnini, B., Oard, D.W., de Rijke, M., Stempfhuber, M., eds.: CLEF. Volume 4730 of Lecture Notes in Computer Science., Springer (2007)