Towards Multi-Stream Question Answering Using Answer Validation

Alberto Téllez-Valero, 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
E-mail: {albertotellezv,mmontesg,villasen}@inaoep.mx

Anselmo Peñas-Padilla
Depto. Lenguajes y Sistemas Informáticos
Universidad Nacional de Educación a Distancia
Juan del Rosal, 16; 28040 Madrid; Spain
E-mail: anselmo@lsi.uned.es

Keywords: question answering, information fusion, answer validation, textual entailment

Received: January 29, 2009

Motivated by the continuous growth of the Web in the number of sites and users, several search engines attempt to extend their traditional functionality by incorporating question answering (QA) facilities. This extension seems natural but it is not straightforward since current QA systems still achieve poor performance rates for languages other than English. Based on the fact that retrieval effectiveness has been previously improved by combining evidence from multiple search engines, in this paper we propose a method that allows taking advantage of the outputs of several QA systems. This method is based on an answer validation approach that decides about the correctness of answers based on their entailment with a support text, and therefore, that reduces the influence of the answer redundancies and the system confidences. Experimental results on Spanish are encouraging; evaluated over a set of 190 questions from the CLEF 2006 collection, our method responded correctly 63% of the questions, outperforming the best QA participating system (53%) by a relative increase of 19%. In addition, when they were considered five answers per question, our method could obtain the correct answer for 73% of the questions. In this case, it outperformed traditional multi-stream techniques by generating a better ranking of the set of answers presented to the users.

Povzetek: Metoda temelji na kombiniranju odgovorov več sistemov za QA.

1 Introduction

In the last two decades the discipline of Automatic Text Processing has showed an impressive progress. It has found itself at the center of the information revolution triggered by the emergence of Internet. In particular, the research in information retrieval (IR) has led to a new generation of tools and products for searching and navigating the Web. The major examples of these tools are search engines such as Google[1] and Yahoo[2]. This kind of tools allows users to specify their information needs by short queries (expressed by a set of keywords), and responds to them with a ranked list of documents.

At present, fostered by diverse evaluation forums (TREC[3], CLEF[4], and NTCIIR[5]), there are important efforts to extend the functionality of existing search engines. Some of these efforts are directed towards the development of question answering (QA) systems, which are a new kind of retrieval tools capable of answering concrete questions. Examples of pioneering Web-based QA systems are START[6] and DFKI[7].

Regardless of all these efforts, the presence of QA systems in the Web is still too small compared with traditional search engines. One of the reasons of this situation is that QA technology, in contrast to traditional IR methods, is not equally mature for all languages. For instance, in the TREC 2004, the best QA system for English achieved an accuracy of 77% for factoid questions[8] (Voorhees, 2004), whereas, two years later in the CLEF 2006, the best QA system for Spanish could only obtain an accuracy of 55% for the same kind of questions (Magnini et al, 2006). Taking into account that Spanish is the third language with more presence in the Web[9], and that it is the second language used

[1] http://www.google.com
[2] http://www.yahoo.com
[3] The Text REtrieval Conference. http://trec.nist.gov/
[4] The Cross Language Evaluation Forum. http://www.clef-campaign.org/
[5] The NTCIR Project. http://research.nii.ac.jp/ntcir/
[6] http://start.csail.mit.edu
[7] http://experimental-quetal.dfki.de
[8] Questions that asks for short, fact-based answers such as the name of a person or location, the date of an event, the value of something, etc.
[9] Internet World Stats (November 2007).

for searching it (de Sousa, 2007), these results clearly show the necessity of improving current accuracy of Spanish QA systems.

Recently, an alternative approach known as a multi-stream QA has emerged. In this approach the idea is to combine the output of different QA systems (streams) in order to obtain a better answer accuracy. This is an ideal solution due to the evidence that a perfect combination of the correct answers from several Spanish QA systems could improve by 31.5% the best individual result (Vallin et al, 2005).

In line with these efforts, in this paper we propose a new *multi-stream approach for QA*. Different to most previous methods, the proposed approach is specially suited to work with poor performance QA systems, representing the real situation in most non-English languages. In particular, it is based on an answer validation method that decides about the correctness of answers based on their entailment with a given support text. In this way the method does not rely on the stream's confidences, nor depend on the redundancy of the answers across the systems.

Our experimental results in a set of 190 questions from the CLEF 2006 collection demonstrate the appropriateness of the proposed method for combining the output of several (including poor performance) QA systems. It could correctly respond 63% of the questions, outperforming the best QA participating system (53%) by a relative increase of 19%. In addition, when we considered a set of five answers per question, our method could obtain the correct answer for 73% of the questions. In this case, it outperformed other multi-stream techniques by generating a better ranking of the set of answers presented to the users. This last characteristic is of great relevance for Web applications, where users hope to get the requested information as direct as possible.

The rest of the paper is organized as follows. Section 2 organizes the previous work in multi-stream QA. Section 3 and 4 describe our proposal for a multi-stream QA method based on an answer validation approach. Then, Section 5 presents the evaluation results of the proposed method in a set of 190 questions in Spanish language. Finally, Section 6 exposes our conclusions and outlines some future work directions.

2 Related work

Typically, QA systems consist of a single processing stream that performs three components in a sequential fashion: question analysis, document/passage retrieval, and answer selection (see e.g., (Hovy et al, 2000)). In this single-stream approach a kind of information combination is often performed within its last component. The goal of the answer selection component is to evaluate multiple candidate answers in order to choose from them the most likely answer for the question. There are several approaches for answer selection, ranging from those based on lexical overlaps and answer redundancies (see e.g., (Xu et al, 2002)) to those based on knowledge intensive methods (see e.g., (Moldovan et al, 2007)).

Recently, an alternative approach known as a multi-stream QA has emerged. In this approach the idea is to combine different QA strategies in order to increase the number of correctly answered questions. Mainly, multi-stream QA systems are of two types: internal and external.

Internal multi-stream systems use more than one stream (in this case, more than one strategy) at each particular component. For instance, Pizzato and Mollá-Aliod (2005) describes a QA architecture that uses several document retrieval methods, and Chu-Carroll et al (2003) presents a QA system that applies two different methods at each system component.

On the other hand, *external multi-stream systems* directly combine the output of different QA systems. They employ different strategies to take advantage of the information coming from several streams. Following we describe the main strategies used in external multi-stream QA systems. It is important to mention that most of these strategies are adaptations of well-known information fusion techniques from IR. Based on this fact, we propose organizing them into five general categories taking into consideration some ideas proposed elsewhere (Diamond, 1996; Vogt and Cottrell, 1999).

Skimming Approach. The answers retrieved by different streams are interleaved according to their original ranks. In other words, this method takes one answer in turn from each individual QA system and alternates them in order to construct the final combined answer list. This approach has two main variants. In the first one, that we called *Naïve Skimming Approach*, the streams are selected randomly. Whereas, in the second variant, which we called *Ordered Skimming Approach*, streams are ordered by their general confidence. In other words, QA systems are ordered by their global answer accuracy estimated from a reference question set. Some examples of QA systems that use this approach are described in (Clarke et al, 2002) and (Jijkoun and de Rijke, 2004).

Chorus Approach. This approach relies on the answer redundancies. Basically, it ranks the answers in accordance to their repetition across different streams. Some systems based on this approach are described in (de Chalendar et al, 2002), (Burger et al, 2002), (Jijkoun and de Rijke, 2004), (Roussinov et al, 2005), and (Rotaru and Litman, 2005).

Dark Horse Approach. This approach can be considered as an extension of the Ordered Skimming Approach. It also considers the confidence of streams, however, in this case, these confidences are computed separately for each different answer type. That is, using this approach, a QA system will have different confidence values associated to factoid, definition and list questions. A QA system based on this strategy is described in (Jijkoun and de Rijke, 2004).

Web Chorus Approach. This approach uses the Web information to evaluate the relevance of answers. It basically

http://www.internetworldstats.com/stats7.htm

ranks the answers based on the number of Web pages containing the answer terms along with the question terms. It was proposed by Magnini et al (2001), and subsequently it was also evaluated in (Jijkoun and de Rijke, 2004).

Answer Validation Approach. In this approach the decision about the correctness of an answer is based on its entailment with a given support text. This way of answer selection not only allows assuring the rightness of answers but also their consistency with the snippets that will be showed to the users. This approach was suggested by Peñas et al (2007), and has been implemented by Glöckner et al (2007)[10].

In addition, it has also been used a combination of different approaches. For instance, Jijkoun and de Rijke (2004) describes a QA architecture that combines a chorus-based method with the dark horse approach. Its evaluation results indicate that this *hybrid approach* outperformed the results obtained by systems based on one single multi-stream strategy[11].

In this paper we propose a new multi-stream QA method based on the answer validation approach. We decide using this approach because it does not consider any confidence about the input streams as well as it does not exclusively depend on the answer redundancies. These characteristics make this approach very appropriate for working with poor performance QA systems such as those currently available for most languages except for English.

Our method distinguishes from existing answer-validation multi-stream methods (Glöckner, 2006; Tatu et al, 2006) in the following two concerns. First, it is the only one specially suited for Spanish, and second, whereas the other two methods are based on a deep semantic analysis of texts, ours is only based on a lexical-syntactic analysis of documents. We consider this last difference very important for constructing Web applications since it makes our method more easily portable across languages.

In particular, the proposed answer validation method is based on a supervised learning approach that considers a combination of two kinds of attributes. On the one hand, some attributes that indicate the compatibility between the question and the answer, and on the other hand, some attributes that allow evaluating the textual entailment between the question-answer pair and the given support text. The first kind of attributes has been previously used in traditional single-stream QA systems (e.g., (Vicedo, 2001)), whereas the second group of attributes is commonly used by answer validation (AV) and textual entailment recognition (RTE) systems (e.g., (Kozareva et al, 2006; Jijkoun and de Rijke, 2005)). In this case, our method not only considers attributes that indicate the overlap between the question-answer pair and the support text, but also includes some attributes that evaluates the non-overlapped information. In some sense, these new attributes allow analyzing the situations where exists a high overlap but not necessarily an entailment relation between these two elements.

The following section describes in detail the proposed method.

3 A multi-stream QA system based on answer validation

Figure 1 shows the general scheme of the proposed external multi-stream QA System. It uses an *answer validation module* to superficially combine the outputs (answers) from several streams (QA systems).

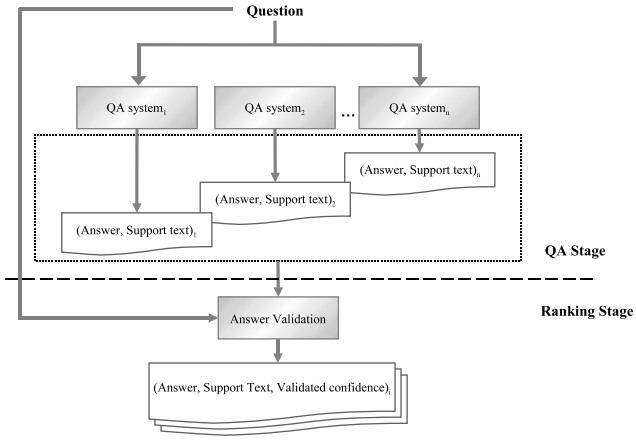

Figure 1: Multi-stream QA System based on Answer Validation

Mainly, the proposed multi-stream QA system consists of two main stages. In the first stage (called QA stage), several different QA systems extract — in parallel — a candidate answer and its corresponding support text for a given question. Then, in the second stage (called ranking stage), an answer validation module evaluates — one by one — the candidate answers and assigns them a confidence value from 0 to 1. A confidence value equal to 0 indicates that the answer is totally rejected, whereas a confidence equal to 1 indicates that the answer is completely accepted. At the end, answers are ranked in line with their confidence values.

The following section describes in detail the answer validation method. This method is an extension of our previous work described in Téllez-Valero et al (2007). In particular, it includes a novel set of attributes for answer validation which allow to increase our previous results by 14% as well as to outperform all results reported in the 2006 Spanish Answer Validation Exercise (Peñas et al, 2006).

4 The answer validation module

Given a question (Q), a candidate answer (A) and a support text (S), the answer validation module returns a confidence value (β) that allows deciding whether to accept or reject

[10](Harabagiu and Hickl, 2006) also describes an answer validation approach for multi-stream QA; nevertheless, it is an internal approach.

[11]They compared their hybrid method against all other approaches except the answer validation.

the 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.

Our answer validation module is mainly based on the idea of 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. The entailment between the pair (*T, H*) occurs when the meaning of *H* can be inferred from the meaning of *T* (Dagan et al, 2005).

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

4.1 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.

4.1.1 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`.

In order to detect the question constituents we firstly apply a shallow parsing to the given question. 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.

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.`
Relevant support text	`Iraqi invasion of Kuwait`

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

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

4.1.2 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 the last sentence (a smaller text fragment) is useful for validating the given answer, whereas the rest of the text only contribute to produce an irrelevant overlap.

In order to reduce the support text to the minimum useful text fragment we proceed as follows:

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

- Second, we match the content terms (nouns, verbs, adjectives and adverbs) from question constituents against the terms from S_{parsed}. In order to capture the morphological inflections of words we compare them using the Levenshtein edition distance [12]. Mainly, we

[12]The Levenshtein edition distance has been previously used in other works related to answer validation in Spanish language, see for instance (Rodrigo et al, 2006).

consider that two different words are equal if its distance value is greater than 0.6.

- 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*. 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`

4.2 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'*.

4.2.1 Attributes about the question-answer relation

Question characteristics

We consider three different attributes from the question: 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 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 is` [*whatever*] → DEFINITION-OTHER
`Who is` [*whatever*] → DEFINITION-PERSON
`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".

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" (Dalmas and Webber, 2007), we decided to include an attribute related to the occurrence of the answers across streams.

Different from the Chorus Method (refer to Section 2) that directly uses the frequency of occurrence of the answers across streams, the proposed attribute indicates the average similarity of the candidate answer with the rest of stream answers (it takes values from 0 to 1).

In order to deal with the great language variability and also with the presence of some typing errors, we decide using the Levenshtein edition distance to measure the similarity between answers. Using this strategy, the answer `X` contributes to the redundancy rate of the answer `Y` and vice versa.

4.2.2 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 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'*. On the other hand, 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 the overlap for each type of content term (nouns, verbs, adjectives and adverbs) as well as for each type of named entity (names of persons, places, organizations, and other

things, as well as dates and quantities). In total we generate 10 different overlap attributes.

Non-overlap characteristics

These attributes indicate the number of non-overlapped terms from the core fragment of the support text, that is, the number of terms from *T'* that are not present in *H'*.

Similar to the previous kind of attributes, for this case we also compute the non-overlap for each type of content term (nouns, verbs, adjectives and adverbs) as well as for each type of named entity (names of persons, places, organizations, and other things, as well as dates and quantities). In total we generate 10 different non-overlap attributes.

4.2.3 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[13].

The constructed classifier decides whether to accept or reject the candidate answer based on the twenty-five 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.

5 Experimental results

5.1 Experimental setup

5.1.1 Training and test data

As we describe in section 3, the core component of the proposed multi-stream QA method is the answer validation module, which relies on a *supervised learning* approach.

In order to train this module we used the SPARTE corpus. This corpus was build from the Spanish corpora used at CLEF for evaluating QA systems from 2003 to 2005. It contains 2962 training instances represented by the tuple ($\langle$question$\rangle$, $\langle$answer$\rangle$, $\langle$support-text$\rangle$, $\langle$entailment-value$\rangle$), where $\langle$entailment-value$\rangle$ is a binary variable indicating whether the support text entails or not the question-answer pair.

One important fact about this corpus is that it is very unbalanced: 77% of the training instances are negative (their entailment value is FALSE), whereas just 695 instances (the rest 23%) correspond to positive entailment examples.

On the other hand, for evaluating the proposed method, we used a set of 190 questions and the answers from 17 different QA systems (i.e., 17 different streams). In total, we considered 2286 candidate answers with their corresponding support texts.

[13]We used the Weka implementations for the AdaBoot and ADTree algorithms (Witten and Frank, 1999).

The used test set gathers the outputs from all QA systems participating at the QA track of CLEF 2006 (Magnini et al, 2006), and it was employed at the first Spanish Answer Validation Exercise (Peñas et al, 2006).

5.1.2 Evaluation measure

The evaluation measure most commonly used in QA is the *accuracy*, i.e., the percentage of correctly answered questions. Following the CLEF evaluation criteria, this measure is calculated as the fraction of correct answers and correct nil-answers[14] with respect to the total number of questions (see formula 1).

$$Accuracy = \frac{|\text{right\_answers}| + |\text{right\_}\textit{nil's}|}{|\text{questions}|} \quad (1)$$

In particular, in our experiments we used an evaluation measure called *accuracy@N*, which basically indicates the accuracy of a QA system when considering *N* candidate answers for each question. In this case, an answer is evaluated as correct if it occurs in the list of *N* candidate answers, independently of its position.

5.1.3 Results from the input streams

Table 2 shows some data from the input streams. It mainly presents their number of right and wrong answers as well as their accuracy for each different type of question. From this table, it is noticeable that most QA systems (streams) have a very poor performance level, having an average accuracy of 26%.

5.2 Experiments

5.2.1 First experiment: general evaluation of the proposed method

The objective of an external multi-stream QA method is to combine the responses from different QA systems in order to increase the final answer accuracy. In other words, its goal is to obtain a better result than that from the best input stream.

In a first experiment, we attempted to evaluate the fulfillment of this objective. We compared the results obtained by our method with the accuracy from the *best input stream* (53%). In addition, we also compared our method against other multi-stream approaches (refer to Section 2). In particular, we implemented some methods from these approaches based on the following criteria:

- *The Naïve Skimming Method*. In this case, streams maintain the order showed in Table 2.

[14]Nil questions do not have an answer in the target document collection, or even worst, they do not have any possible answer. As an example consider the question `What is the capital of Neverland?`. For these questions give no answer is considered as a correct response.

Stream	Right answers	*nil's*	Wrong answers	Accuracy
1	25	16	77	0.22
2	48	17	46	0.34
3	49	7	113	0.30
4	34	10	92	0.23
5	10	1	179	0.06
6	24	5	142	0.15
7	16	3	138	0.10
8	88	12	69	**0.53**
9	31	7	125	0.20
10	26	10	125	0.19
11	15	11	78	0.14
12	85	12	63	0.51
13	33	10	88	0.23
14	21	18	34	0.21
15	57	13	89	0.37
16	45	12	102	0.30
17	64	16	55	0.42

Table 2: Results from the input streams

- *The Ordered Skimming Method*. It ranks the answers in accordance to the stream's overall accuracies (refer to the last column of Table 2).
- *The Chorus Method*. It ranks the answers based on their repetitions across different streams.
- *The Dark Horse Method*. It uses the factoid accuracies to rank answers corresponding to factoid questions and the definition accuracies for ranking the answers for definition questions (refer to the antepenultimate and penultimate columns of Table 2.
- *The Web Chorus Method*. It ranks the answers based on the number of Web pages that contain the terms of the question (without the question word) along with the terms of the answer.

Table 3 shows the results from the first experiment. This table also includes the accuracy corresponding to a *perfect combination* of the correct answers from all streams (87%). This value indicates the maximum reachable accuracy for a multi-stream approach in this data set.

The results from this first experiment show that our method was the only multi-stream approach that could improve the result from the best input stream; it responded correctly 58% of the questions outperforming the best individual result (53%) by a relative increase of 9%. Considering a list of five candidate answers (which is the typical configuration of existing online QA systems) our method outperformed the accuracy from the best input stream by 11%, a relative improvement of 18%.

The methods that rank answers based on the stream confidences, namely the Ordered Skimming Method and the Dark Horse Method, also obtained relevant results. However, it is necessary to mention that – in our implementations – these methods made use of a *perfect estimation* of these confidences[15]. For that reason, and given that in a real scenario it is practically impossible to obtain these perfect estimations, we consider that our proposal is more robust than these two methods.

The results from Table 3 also give evidence that the presence of several deficient streams (which generate a lot of incorrect answers) seriously affects the performance of the Naïve Skimming Method. This phenomenon also had an important effect over the Chorus Method, which normally is reported as one of the best multi-stream approaches.

Finally, it is important to comment that we attribute the poor results achieved by the Web Chorus Method to the quantity of online information for Spanish (which it is considerably less than that for English). In order to obtain better results it is necessary to apply some question/answer expansions, using for instance synonyms and hyperonyms.

5.2.2 Second experiment: the impact of rejecting less reliable answers

Taking into account that the accuracy of QA systems not only depends on the number of correctly answered questions, but also on the number of correctly *unanswered* nil questions, we decided to modify the basic multi-stream methods (including ours) in order to allow them rejecting some answers. The idea was to incorporate some filtering conditions that obligate the methods to eliminate the less reliable answers. In the cases that no answer could satisfy these conditions, the answer was set to nil. Following we describe the modifications incorporated to each one of the methods.

- *Ordered Skimming Method**. It only considers answers from the best five streams (i.e., it only returns answers coming from the streams with the five highest global accuracies).
- *Chorus Method**. It only considers answers recommended by two or more streams.
- *Dark Horse Method**. It only returns answers coming from the best five streams for each question type. In this case there were selected the best five streams for answering factoid questions and the best five for answering definition questions.
- *Our Method**. It only returns answers with a validation confidence greater than 0.5.

Table 4 shows the results from this second experiment. It is interesting to notice that all methods improved their results when they rejected some answers. The explanation of this behavior is that with these modifications all methods

[15]The confidences were calculated directly from the test set (refer to Table 2). It was so because there is no correspondence between the systems that were used to generate the train and test sets.

	Number of answers by question				
	1	2	3	4	5
Naïve Skimming Method	0.25	0.44	0.50	0.55	0.61
Ordered Skimming Method	0.52	0.63	0.66	0.68	0.71
Chorus Method	0.53	0.61	0.66	0.68	0.68
Dark Horse Method	0.52	0.61	0.67	0.68	0.72
Web Chorus Method	0.17	0.28	0.37	0.46	0.56
Our Method	**0.58**	**0.65**	**0.69**	**0.71**	**0.73**
Best Input Stream	0.53	0.56	0.58	0.62	0.62
Perfect Combination	0.87	-	-	-	-

Table 3: Results from the first experiment: general evaluation of the proposed method

could answer some nil questions. In particular, our method correctly respond 63% of the questions outperforming the best input stream by a relative increase of 19%.

This experiment also helped to reveal another important characteristic of our method. It could correctly reject several answers without using any information about the confidence of streams and without considering any restriction on the answer frequencies.

5.2.3 Third experiment: combination of our method with other approaches

In (Jijkoun and de Rijke, 2004), Jijkoun and de Rijke describe a multi-stream QA architecture that combines the Chorus and the Dark Horse Methods. Its evaluation results indicate that this combination outperformed the results obtained by other systems based on one single multi-stream strategy[16].

Motivated by this result, we designed a third experiment which considered the combination of our method with other confidence-based methods, in particular, the Dark Horse Method and the Ordered Skimming Method. The combination of our method with these two other approaches was performed as follows. In a first stage, our method selected a set of candidate answers, then, in a second stage, a confidence-based method ordered the candidate answers in accordance to their own ranking criteria[17].

Table 5 shows the results from the combination of these methods. On the one hand, these results confirm the conclusions of Jijkoun and de Rijke since they also indicate that the combination of methods outperformed the results obtained by individual approaches. On the other hand, and most important, these results demonstrate the competence of our method since they show that its individual result outperformed that from the combination of the Chorus Method with the Dark Horse Method (stated by Jijkoun and de Rijke as the best configuration for a multi-stream QA system).

[16]In their experiments, as mentioned in Section 2, they did not consider the answer validation approach.

[17]Given that we use the same implementations for the confidence-based methods that those described in the first experiment, in this case we also used a perfect estimation of the streams confidences.

6 Conclusions and future work

In this paper we proposed a new external multi-stream QA method. This method is founded on the idea of combining the output of different QA systems (streams) in order to obtain a better answer accuracy.

The proposed method is based on an answer validation approach. This way, it decides about the correctness of the answers based on their entailment with a support text, and does not exclusively rely on answer redundancies nor on the stream confidences. In addition, this method only considers lexical-syntactic information and does not make use of a deep semantic analysis of texts. All these features together make our method appropriate for dealing with poor performance QA systems which represent the current state for most non-English languages. In particular, we have evaluated our method in Spanish, where current average answer accuracy is of 26% (please refer to Table 2).

The core component of the proposed multi-stream method is the answer validation module. This module applies a supervised 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 support text. At this point, it is important to comment that an evaluation of the proposed attributes during the development phase — using the information gain algorithm — showed us that the non-overlap and answer-redundancy attributes were the most discriminative.

From the evaluation results achieved on a test set of 190 Spanish questions from the CLEF-2006 QA collection, we could observe the following:

- The proposed method significantly enhanced the accuracy from the best individual stream. It correctly responded to 63% of the questions, outperforming the best QA participating system (53%) by a relative increase of 19%.
- Although our method also takes advantage of the re-

	Number of answers by question				
	1	2	3	4	5
Ordered Skimming Method*	0.55	0.66	0.69	0.70	0.71
Chorus Method*	0.58	0.64	0.67	0.67	0.67
Dark Horse Method*	0.55	0.64	0.68	0.69	0.69
Our Method*	**0.63**	**0.69**	**0.72**	**0.73**	**0.73**
Best input stream	0.53	0.56	0.58	0.62	0.62
Perfect Combination	0.87	-	-	-	-

Table 4: Results from the second experiment: the impact of rejecting less reliable answers

	Number of answers by question				
	1	2	3	4	5
Chorus Method* + Ordered Skimming Method	0.63	0.65	0.66	0.67	0.67
Chorus Method* + Dark Horse Method	0.62	0.65	0.66	0.67	0.67
Our Method* + Ordered Skimming Method	**0.64**	**0.71**	**0.72**	**0.73**	**0.73**
Our Method* + Dark Horse Method	0.63	0.69	**0.72**	**0.73**	**0.73**
Best Input Stream	0.53	0.56	0.58	0.62	0.62
Perfect Combination	0.87	-	-	-	-

Table 5: Results from the third experiment: combination of our method with other approaches

dundancy of answers across streams, it turned out to be less sensible to their low frequency than other approaches. For instance, it outperformed the Chorus Method by 5%.

– The proposed method allowed to significantly reduce the number of wrong answers presented to the user. In relation to this aspect, our method was especially adequate to deal with nil questions. It correctly responded 65% of the nil questions, outperforming the best input stream by a relative increase of 8%.

– The combination of our method with the Dark Horse approach only produced a slightly improvement of 1%. This fact indicates that our method does not require knowing the input stream confidences.

Finally, it is clear that any improvement in the answer validation module will directly impact the performance of the proposed multi-stream method. Hence, our future work will be mainly focused on enhancing this module by: (i) considering some new features in the entailment recognition process, (ii) including a process for treatment of temporal restrictions, and (iii) using Wordnet in order to consider synonyms and hyperonyms for computing the term and structure overlaps.

Acknowledgement

This work was done under partial support of Conacyt (scholarship 171610). We also thank CLEF organizers for the provided resources.

References

Burger JD, Ferro L, Greiff WR, Henderson JC, Mardis S, Morgan A, Light M (2002) MITRE's qanda at TREC-11. In: TREC

Chu-Carroll J, Czuba K, Prager JM, Ittycheriah A (2003) In question answering, two heads are better than one. In: HLT-NAACL

Clarke CLA, Cormack GV, Kemkes G, Laszlo M, Lynam TR, Terra EL, Tilker PL (2002) Statistical selection of exact answers (multitext experiments for TREC 2002). In: TREC

Dagan I, Glickman O, Magnini B (2005) The PASCAL recognising textual entailment challenge. In: Quiñonero J, Dagan I, Magnini B, d'Alché Buc F (eds) MLCW, Springer, Lecture Notes in Computer Science, vol 3944, pp 177–190

Dalmas T, Webber BL (2007) Answer comparison in automated question answering. J Applied Logic 5(1):104–120

de Chalendar G, Dalmas T, Elkateb-Gara F, Ferret O, Grau B, Hurault-Plantet M, Illouz G, Monceaux L, Robba I, Vilnat A (2002) The question answering system QALC at LIMSI, experiments in using web and wordnet. In: TREC

de Sousa P (2007) El español es el segundo idioma más usado en las búsquedas a través de google (english translated: The spanish is the second language most used in the google's searches). CDT internet.net, URL `http://www.cdtinternet.net/`

Diamond T (1996) Information retrieval using dynamic evidence combination, PhD Dissertation Proposal, School of Information Studies, Syracuse University

Glöckner I (2006) Answer validation through robust logical inference. In: Peters et al (2007), pp 518–521

Glöckner I, Sven H, Johannes L (2007) Logical validation, answer merging and witness selection - a case study in multi-stream question answering. In: RIAO 2007, Large-Scale Semantic Access to Content, Pittsburgh, USA

Harabagiu SM, Hickl A (2006) Methods for using textual entailment in open-domain question answering. In: ACL, The Association for Computer Linguistics

Hovy EH, Gerber L, Hermjakob U, Junk M, Lin CY (2000) Question answering in webclopedia. In: TREC

Jijkoun V, de Rijke M (2004) Answer selection in a multi-stream open domain question answering system. In: McDonald S, Tait J (eds) ECIR, Springer, Lecture Notes in Computer Science, vol 2997, pp 99–111

Jijkoun V, de Rijke M (2005) Recognizing textual entailment: Is word similarity enough? In: Candela JQ, Dagan I, Magnini B, d'Alché Buc F (eds) MLCW, Springer, Lecture Notes in Computer Science, vol 3944, pp 449–460

Kozareva Z, Vázquez S, Montoyo A (2006) University of alicante at qa@clef2006: Answer validation exercise. In: Peters et al (2007), pp 522–525

Magnini B, Negri M, Prevete R, Tanev H (2001) Is it the right answer?: exploiting web redundancy for answer validation. In: ACL, The Association for Computational Linguistics, Morristown, NJ, USA, pp 425–432

Magnini B, Giampiccolo D, Forner P, Ayache C, Jijkoun V, Osenova P, Peñas A, Rocha P, Sacaleanu B, Sutcliffe RFE (2006) Overview of the clef 2006 multilingual question answering track. In: Peters et al (2007), pp 223–256

Moldovan DI, Clark C, Harabagiu SM, Hodges D (2007) Cogex: A semantically and contextually enriched logic prover for question answering. J Applied Logic 5(1):49–69

Peñas A, Rodrigo Á, Sama V, Verdejo F (2006) Overview of the answer validation exercise 2006. In: Peters et al (2007), pp 257–264

Peñas A, Rodrigo Á, Sama V, Verdejo F (2007) Testing the reasoning for question answering validation. Journal of Logic and Computation (3), DOI 10.1093/logcom/exm072

Peters C, Clough P, Gey FC, Karlgren J, Magnini B, Oard DW, de Rijke M, Stempfhuber M (eds) (2007) 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, LNCS, vol 4730, Springer

Pizzato LAS, Mollá-Aliod D (2005) Extracting exact answers using a meta question answering system. In: Proceedings of the Australasian Language Technology Workshop, Sydney, Australia, pp 105–112

Rodrigo Á, Peñas A, Herrera J, Verdejo F (2006) The effect of entity recognition on answer validation. In: Peters et al (2007), pp 483–489

Rotaru M, Litman DJ (2005) Improving question answering for reading comprehension tests by combining multiple systems. In: Proceedings of the American Association for Artificial Intelligence (AAAI) 2005 Workshop on Question Answering in Restricted Domains, Pittsburgh, PA.

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

Tatu M, Iles B, Moldovan DI (2006) Automatic answer validation using cogex. In: Peters et al (2007), pp 494–501

Téllez-Valero A, Montes-y-Gómez M, Villaseñor-Pineda L (2007) A supervised learning approach to spanish answer validation. In: Peters C, Jijkoun V, Mandl T, Müller H, Oard DW, Peñas A, Petras V, Santos D (eds) CLEF, Springer, Lecture Notes in Computer Science, vol 5152, pp 391–394

Vallin A, Magnini B, Giampiccolo D, Aunimo L, Ayache C, Osenova P, Peñas A, de Rijke M, Sacaleanu B, Santos D, Sutcliffe RFE (2005) Overview of the clef 2005 multilingual question answering track. In: Peters C, Gey FC, Gonzalo J, Müller H, Jones GJF, Kluck M, Magnini B, de Rijke M (eds) CLEF, Springer, LNCS, vol 4022, pp 307–331

Vicedo JL (2001) Using semantics for paragraph selection in question answering systems. In: SPIRE, pp 220–227

Vogt CC, Cottrell GW (1999) Fusion via a linear combination of scores. Information Retrieval 1(3):151–173

Voorhees EM (2004) Overview of the TREC 2004 question answering track. In: Proceedings of the Thirteenth Text REtreival Conference (TREC 2004), pp 52–62

Witten IH, Frank E (1999) Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann

Xu J, Licuanan A, May J, Miller S, Weischedel R (2002) TREC 2002 QA at BBN: Answer selection and confidence estimation. In: TREC