Comparing Statistical and Content-Based Techniques for Answer Validation on the Web

Bernardo Magnini, Matteo Negri, Roberto Prevete, and Hristo Tanev
ITC-irst, Centro per la Ricerca Scientifica e Tecnologica,
Via Sommarive, 3850 POVO (TN), Italy

Abstract. Answer Validation is an emerging topic in Question Answering, where open domain systems are often required to rank huge amounts of candidate answers. We present a novel approach to answer validation based on the intuition that the amount of implicit knowledge which connects an answer to a question can be estimated by exploiting the redundancy of Web information. Two techniques are considered in this paper: a statistical approach, which uses the Web to obtain a large amount of pages, and a content-based approach, which analyses text snippets retrieved by the search engine. Both the approaches do not require to download the documents. Experiments carried out on the TREC-2001 judged-answer collection show that a combination of the two approaches achieves a high level of performance (*i.e.* about 88% success rate). The simplicity and the efficiency of these Web-based techniques make them suitable to be used as a module in Question Answering systems.

1 Introduction

Open domain question-answering (QA) systems search for answers to a natural language question either on the Web or in a local document collection. Different techniques, varying from surface patterns [6] to deep semantic analysis [8], are used to extract the text fragments containing candidate answers. Several systems apply answer validation techniques with the goal of filtering out improper candidates by checking how adequate a candidate answer is with respect to a given question. These approaches rely on discovering semantic relations between the question and the answer. As an example, [3] describes answer validation as an abductive inference process, where an answer is valid with respect to a question if an explanation for it, based on background knowledge, can be found. Although theoretically well motivated, the use of semantic techniques on open domain tasks is quite expensive both in terms of the involved linguistic resources and in terms of computational complexity, thus motivating a research on alternative solutions to the problem.

This paper presents a novel approach to answer validation based on the intuition that the amount of implicit knowledge which connects an answer to a question can be estimated by exploiting the redundancy of Web information. Our hypothesis is that a Web search of documents in which the question and the answer co-occur can provide all the information needed to accomplish the answer validation task. Documents are searched in the Web by means of *validation patterns*, which are derived from a linguistic processing of the question and the answer. The retrieved documents are then used for determine an *answer validation*

score as the composition of two alternative answer validation techniques, namely a *statistical approach* and a *content-based approach*.

The statistical approach relies on the hypothesis that the number of documents retrieved from the Web in which the question and the answer co-occur can be considered a significant clue to the validity of the answer. This approach has been addressed in [5]. The content-based approach takes advantage of text passages (i.e. snippets) returned by some search engines and exploit the presence of relevant keywords within such passages. In both cases the Web documents are not downloaded, which makes the algorithms fast. The two approaches has been integrated in a composite algorithm and experimented on a data set derived from the TREC-2001 conference.

The paper is organized as follows. Section 2 presents the main features of the approach. Section 3 describes how validation patterns are extracted from a question-answer pair by means of specific question answering techniques. Section 4 explains the statistical approach. Section 5 give details about the content-based methodology. Section 6 gives the results of a number of experiments and discusses them. Finally, Section 7 outlines the future directions.

2 Overall-Methodology

Given a question q and a candidate answer a the answer validation task is defined as the capability to assess the relevance of a with respect to q. We assume open domain questions and that both answers and questions are texts composed of few tokens (usually less than 100). This is compatible with the TREC-2001 data, that will be used as examples throughout this paper. We also assume the availability of the Web, considered to be the largest open domain text corpus containing information about almost all the different areas of the human knowledge.

The intuition underlying our approach to answer validation is that, given a question-answer pair [q,a], it is possible to formulate a set of *validation statements* whose truthfulness is equivalent to the degree of relevance of a with respect to q. For instance, given the question "What is the capital of the USA?", the problem of validating the answer "Washington" is equivalent to estimating the truthfulness of the validation statement "The capital of the USA is Washington". Therefore, the answer validation task could be reformulated as a problem of statement reliability. There are two issues to be addressed in order to make this intuition effective. First, the idea of a validation statement is still insufficient to catch the richness of implicit knowledge that may connect an answer to a question: we will attack this problem defining the more flexible idea of a *validation pattern*. Second, we have to design an effective and efficient way to check the reliability of a validation pattern: we propose two solutions relying on a statistical count of Web searches and on document content analysis respectively.

Table 1: Web search for validation fragments

1. Capital Region USA: Fly-Drive Holidays in and Around Washington D.C.
2. the Insider's Guide to the Capital Area Music Scene (Washington D.C., USA).
3. The Capital Tangueros (Washington, DC Area, USA)
4. I live in the Nation's Capital, Washington Metropolitan Area (USA).
5. in 1790 Capital (also USA's capital): Washington D.C. Area: 179 square km

Answers may occur in text passages with low similarity with respect to the question.

Passages telling facts may use different syntactic constructions, sometimes are spread in more than one sentence, may reflect opinions and personal attitudes, and often use ellipsis and anaphora. For instance, if the validation statement is "The capital of USA is Washington", we have Web documents containing passages like those reported in Table 1, which can not be found with a simple search of the statement, but that nevertheless contain a significant amount of knowledge about the relations between the question and the answer. We will refer to these text fragments as *validation fragments*.

A common feature in the above examples is the co-occurrence of a certain subset of words (*i.e.* "capital","USA" and "Washington"). We will make use of *validation patterns* that cover a larger portion of text fragments, including those lexically similar to the question and the answer (*e.g.* fragments 4 and 5 in Table 1) and also those that are not similar (*e.g.* fragment 2 in Table 1). In the case of our example a set of validation statements can be generalized by the validation pattern:

[capital <*text*> USA <*text*> Washington]

where <*text*> is a place holder for any portion of text with a fixed maximal length.

2.1 General answer validation algorithm

Starting from the above considerations and given a question-answer pair $[q, a]$, we propose a generic scheme for answer validation. Both the statistical and the content-based approach perform four basic steps:

1. Compute the set of representative keywords Kq and Ka both from q and from a; this step is carried out using linguistic techniques, such as answer type identification (from the question) and named entities recognition (from the answer);
2. From the extracted keywords compute the validation pattern for the pair $[q, a]$;
3. Submit the validation pattern to a search engine;
4. Estimate an *Answer Relevance Score (ARS)* considering the results returned by the search engine.

The retrieval on the Web is delegated to a public available search engine. The post-processing of the results is performed by HTML parsing procedures and simple functions which calculates the *Answer Relevance Score (ARS)* for every $[q, a]$ pair by analysing the result page returned by the search engine.

3 Extracting Validation Patterns

In our approach a validation pattern consists of two components: a question sub-pattern (*Qsp*) and an answer sub-pattern (*Asp*). This is discussed in more details in [5]

Building the *Qsp*. A *Qsp* is derived from the input question cutting off non-content words with a stop-words filter. The remaining words are expanded with both synonyms and morphological forms in order to maximize the recall of retrieved documents. Synonyms are automatically extracted from the most frequent sense of the word in WordNet [2], which considerably reduces the risk of adding disturbing elements. As for morphology, verbs are expanded with all their tense forms (*i.e.* present, present continuous, past tense and past participle). Synonyms and morphological forms are added to the *Qsp* and composed in an `OR` clause.

The following example illustrates how the *Qsp* is constructed. Given the TREC-2001 question "When did Elvis Presley die?", the stop-words filter removes "When" and "did" from the input. Then synonyms of the first sense of "die" (*i.e.* "decease", "perish", etc.) are extracted from WordNet. Finally, morphological forms for all the corresponding verb tenses are added to the *Qsp*. The resultant *Qsp* will be the following:

[Elvis <*text*> Presley <*text*> (die `OR` died `OR` dying `OR` perish `OR` ...)]

Building the *Asp*. An *Asp* is constructed in two steps. First, the *answer type* of the question is identified considering both morpho-syntactic (a part of speech tagger is used to process the question) and semantic features (by means of semantic predicates defined on the WordNet taxonomy; see [4] for details). In general possible answer types are: DATE, MEASURE, PERSON, LOCATION, ORGANIZATION, DEFINITION and GENERIC. DEFINITION is the answer type peculiar to questions like "What is an atom?" which represent a considerable part (around 25%) of the TREC-2001 corpus. The answer type GENERIC is used for non definition questions asking for entities that can not be classified as named entities (*e.g.* the questions: "Material called linen is made from what plant?" or "What mineral helps prevent osteoporosis?")

In the second step, a rule-based named entities recognition module identifies in the answer string all the named entities matching the answer type category. We excluded categories GENERIC and DEFINITION from our experiment because for these categories it is difficult to extract the exact answer from the answer string. An *Asp* for each selected named entity is created. In addition, in order to maximize the recall of retrieved documents, the *Asp* is expanded with verb tenses. The following example shows how the *Asp* is created. Given the TREC question "When did Elvis Presley die?" and the candidate answer "though died in 1977 of course some fans maintain", since the answer type category is DATE the named entities recognition module will select [1977] as an answer sub-pattern.

4 Statistical approach

For the statistical approach we used AltaVista (http://www.altavista.com), because this is one of the search engines with largest indices and provides advanced search through a set of boolean operators. The most interesting among them is the proximity operator `NEAR` which plays an important role in our approach.

4.1 Mining the Web for co-occurrences using the NEAR *operator*

Our statistical algorithm considers the number of pages where both the question keywords and the answer keywords are found in proximity to each other. We use the NEAR logical operator to combine Qsp and Asp into a validation pattern[1]. Such operator searches for pages where the keywords are found in a distance of no more than 10 tokens from each other.

For every question-answer pair the answer validation module submits three searches to the search engine: the sub-patterns [*Qsp*] and [*Asp*] and the validation pattern [*QAp*] built as the composition of the two sub-patterns using the AltaVista NEAR operator. Afterwards, a statistical algorithm considers the output of the Web search for estimating the consistency of the patterns.

Several pattern relaxation heuristics have been defined in order to gradually increase the number of retrieved documents. If the question sub-pattern Qsp does not return any document or returns less than a certain threshold (experimentally set to 7) the question pattern is relaxed by cutting one word; in this way a new query is formulated and submitted to the search engine. This is repeated until no more words can be cut or the returned number of documents becomes higher than the threshold.

4.2 Computing answer relevance

As a result of the Web search with patterns, the search engine returns three numbers: $hits(Qsp)$, $hits(Asp)$ and $hits(Qsp$ NEAR $Asp)$. The probability $P(A)$ of a pattern A in the Web is calculated by:

$$P(A) = \frac{hits(A)}{MaxPages}$$

where $hits(A)$ is the number of pages in the Web where A appears and $MaxPages$ is the maximum number of pages that can be returned by the search engine. We set this constant experimentally. However, in the formula we use, $MaxPages$ may be ignored.

The joint probability P(*Qsp,Asp*) is calculated by means of the validation pattern probability:

$$P(QAp) = P(Qsp \text{ NEAR } Asp)$$

In order to estimate the degree of relevance of Web searches we use a variant of Conditional Probability (i.e. *Corrected Conditional Probability*) which considers the asymmetry of the question-answer relation. In contrast with other measures widely used to find co-occurrence in large corpora (*e.g.* Pointwise Mutual Information [7]), Corrected Conditional Probability (CCP) is not symmetric (*e.g.* generally $CCP(Qsp, Asp) \neq CCP(Asp, Qsp)$). This is based on the fact that we search for the occurrence of the answer pattern Asp only in the cases when *Qsp* is present. The statistical evidence for this can be measured through $P(Asp|Qsp)$, however this value is corrected with $P(Asp)^{2/3}$ in the denominator, to avoid

[1] In general these patterns consist of keywords, but we made experiments with phrase search and weaker question patterns (for details see [5]).

the cases when high-frequency words and patterns are taken as relevant answers. This measure provides an answer relevance score (ARS) for any candidate answer: high values are interpreted as strong evidence that the validation pattern is consistent. This is a clue to the fact that the Web pages where this pattern appears contain validation fragments, which imply answer accuracy.

$$ARS(a) = CCP(Qsp, Asp) = \frac{P(Asp|Qsp)}{P(Asp)^{2/3}}$$

For CCP we obtain:

$$\frac{hits(Qsp\ \texttt{NEAR}\ Asp)}{hits(Qsp) * hits(Asp)^{2/3}} * MaxPages^{2/3}$$

4.3 An example

Consider an example taken from the question answer corpus of the main task of TREC-2001: "Which river in US is known as Big Muddy?". The question keywords are: "river", "US", "known", "Big", "Muddy". The search of the pattern [river NEAR US NEAR (known OR know OR...) NEAR Big NEAR Muddy] returns 0 pages, so the algorithm relaxes the pattern by cutting the initial noun "river", according to the heuristic for discarding a noun if it is the first keyword of the question. The second pattern [US NEAR (known OR know OR...) NEAR Big NEAR Muddy] also returns 0 pages, so we apply the heuristic for ignoring verbs like "know", "call" and abstract nouns like "name". The third pattern [US NEAR Big NEAR Muddy] returns 28 pages, which is over the experimentally set threshold of seven pages.

One of the 50 byte candidate answers from the TREC-2001 answer collection is "recover Mississippi River". Taking into account the answer type LOCATION, the algorithm considers only the named entity: "Mississippi River". To calculate answer validity score (in this example PMI) for [Mississippi River], the procedure constructs the validation pattern: [US NEAR Big NEAR Muddy NEAR Mississippi River] with the answer sub-pattern [Mississippi River]. These two patterns are passed to the search engine, and the returned numbers of pages are substituted in the expression at the places of $hits(Qsp\ \texttt{NEAR}\ Asp)$ and $hits(Asp)$ respectively; the previously obtained number (*i.e.* 28) is substituted at the place of $hits(Qsp)$. In this way an answer validity score of 55.5 is calculated. It turns out that this value is the maximal validity score for all the answers of this question. Other correct answers from the TREC-2001 collection contain as name entity "Mississippi". Their answer validity score is 11.8, which is greater than 1.2 and also greater than $0.2 * Maximal\_Validity\_Score$ $(= 11.1)$. This score (*i.e.* 11.8) classifies them as relevant answers. On the other hand, all the wrong answers has validity score below 1 and as a result all of them are classified as irrelevant answer candidates.

5 Content-based approach

We have chosen Google (http://www.google.com) to back up our content based algorithm. It has the largest document index and, at the same time, it is very fast. Moreover, it has a number of features, such as the support of boolean expressions and co-occurrence snippets extraction, which are a prerequisite for fast and successful retrieval of validation fragments.

5.1 Using text snippets for co-occurrence mining in the Web

Google gives highest ranking for documents where the query terms co-occur closely [1], which allows to analyse only the first result page. When a query is submitted to Google, in the returned pages the search engine provides surrounding context for the first occurrences of a query term. When two or more keywords co-occur near to each other, they are included in one co-occurrence fragment. Our experiments with the search engine show that Google prefers to extract the snippets where close co-occurrences take place ignoring the passages where only one keyword appears. The validation algorithm makes intensive use of these context snippets to identify co-occurrences between the answer and the question keywords.

For every question-answer pair the answer validation module submits one query to the search engine, i.e. the validation pattern [Qsp AND Asp] built from the question and candidate answer keywords (see Section 3). This query aims at the pages where keywords both from Qsp and Asp appear. Moreover, as Google gives the highest ranks to the documents where the keywords appear close to each other (see [1]), if there is co-occurrence tendency of Qsp and Asp, it will be shown in the top ranked hits.

During query submission we set the number of results per page to 100, i.e. we consider the top one hundred documents. Each document is provided with list of document snippets where query terms appear. An example of text snippets is presented in Table 2

Table 2: Snippets for the top three ranked documents obtained by the query [*Idaho (become* OR *became) state 1890*]

1.	...This, too, was controversial and was redrawn several times. Nevertheless, it was used until **Idaho became** a **state** in **1890**. STATE SEAL NOW IN USE. ...
2.	... **Idaho State** Symbols. ... Entered Union: July 3, **1890**, 43rd **state** to ... the University of **Idaho**, wrote the verse which became the chorus of the **Idaho State** ...
3.	... Legislature on January 11, 1866. This, too, was controversial and was redrawn several times. Nevertheless, it was used until **Idaho became** a **state** in **1890**. ...

All the snippets are separated by an ellipsis symbol "..." and each snippet contains at least one query term. How it was stated before, if the keywords are found in proximity to each other, they are provided with common context. This way we obtain a list of contexts where the answer and question keywords appear together.

Our hypothesis is that the closer the distance between the answer a and the question keywords, the stronger their relation is. We consider strong relation between a and the question keywords as a clue to the validation pattern consistency and answer relevance. For example, the first snippet in Table 2 includes both the answer and the question keywords and the keyword density is very high. In fact, if we take out the stop words, the distance between the keywords will become zero.

5.2 Computing answer relevance

Co-occurrence weight. The validation procedure identifies the context snippets and evaluates them to obtain an answer relevance score. Every appearance of a candidate answer a in a context snippet is evaluated by calculating the number of the question keywords and their distance from a. We call this score *Co-occurrence Weight* (CW). If we have co-occurrence of

the answer a and a set of question keywords $QK = \{qk_1, qk_2 \ldots\}$, $CW(a, QK)$ is calculated by means of the following formula:

$$CW(a, QK) = \prod_i w(qk_i)^{(\|qk_i\, a\|+1)^{-1}}$$

Where $w(qk_i)$ is the weight of the question keyword qk_i. In general $w(qk_i)$ can be calculated from the keyword frequency. However in our experiment we used equal weights for all the words. We denote the distance between the answer a and the closest appearance of qk_i by $\|qk_i\, a\|$. The distance is measured with the number of non-stop non-keywords between a and qk_i. As many different question keywords occur in proximity to a as higher their CW is. The CW also depends on the distance between a and the question terms.

If we denote with S_a the set of text snippets where a appears in the first one hundred documents, the Answer Relevance Score $ARS(a)$ is calculated in the following way:

$$ARS(a) = \sum_{QK \in S_a} CW(a, QK)$$

By using a sum of the $[q, a]$ co-occurrence weights we stress on the number of co-occurrences. This way we consider more important the patterns which appear with higher frequency in the top 100 most relevant documents retrieved by Google.

Answers which obtain *ARS* lower than a certain threshold are discarded. The rest are sorted by *ARS* and the answer with maximal score is judged correct along with all the answers which have similar score.

5.3 An example

As an example of content based answer validation consider the question-answer pair:

Question: *When did Idaho become a state?*
Answer: *1890*

First, the keywords are extracted from the question, which results in the keyword list $Idaho$, $become$ and $state$. Next, the past form of the verb $become$ is added to the keyword list and the question pattern is transformed into the Google query:

[*Idaho (become* OR *became) state*]

Next we add the candidate answer *1890* and the query becomes:

[*Idaho (become* OR *became) state 1890*]

This query is equivalent to the boolean expression:

[*Idaho* AND *(become* OR *became)* AND *state* AND *1890*]

The search for this expression in the Web catches all the pages where the words $Idaho$, $state$ and one of the forms of $become$ are present.

Table 3: Performance of the answer validation algorithm on different types of factoid questions

Q. type	Test set		Baseline		Content-based			Statistical		
	# q	#qa	P	SR	P	R	SR	P	R	SR
DATE	52	282	60.0	70.2	91.3	92.0	92.6	90.2	95.2	93.3
MEASURE	61	299	61.7	76.1	82.3	56.0	78.3	86.1	53.4	78.6
PERSON	56	323	52.5	56.7	91.9	80.0	87.0	86.3	89.7	88.2
LOCATION	80	447	52.4	57.9	87.3	82.6	86.4	81.6	89.9	85.9
total	249	1351	55.5	64.2	88.6	78.8	86.0	85.4	83.9	86.4

For the above mentioned example Google returns about 11,100 hits. The snippet lists for the first three hits are presented in Table 2. In the first text snippet the distance between the candidate answer "1890" and all the question keywords ("Idaho", "became" and "state") is 0, since only stop-words are present between the candidate answer and the question keywords. We set experimentally a constant weight of 2 for any question keyword. By substituting these values in the formula for co-occurrence weight, we obtain a weight of 8 for the first co-occurrence snippet. In this way we assign a weight to all the snippets in the top 100 hits. Finally, we sum these weights and obtain the final ARS. In case there are other candidate answers for the question an ARS is calculated for each of them and the candidate with the higher value is selected.

6 Experiments and Discussion

A number of experiments have been carried out in order to check the validity of the proposed answer validation techniques. As a main data set, the 249 factoid questions[2] of the TREC-2001 database have been used. For each question, at most three correct answers and three wrong answers have been randomly selected from the TREC-2001 participants' submissions, resulting in a corpus of 1351 question-answer pairs (some question have less than three positive answers in the corpus). Additionally, we tested the statistical approach on the full set of the 492 TREC-2001 questions.

We wanted to check performance variation based on different types of TREC-2001 questions. We carried out five evaluations: for questions which ask for DATE, MEASURE, PERSON, or LOCATION and the total performance on the full set of named entity questions. The number of questions for each type is reported in Table 3.

The baseline model. A baseline for the answer validation experiment was defined by judging correct all the answer strings which include at least one named entity belonging to the type of the question. Baseline results are also reported in Table 3.

Results. For each question type we report in Table 3 precision (P), recall (R) and success rate (SR) for both the content-based and the statistical approach. Success rate best represents the performance of the system, being the percent of $[q, a]$ pairs where the result given by the system is the same as the TREC judges' opinion. Precision is the percent of $[q, a]$ pairs estimated by the algorithm as relevant, for which the opinion of TREC judges was the same. Recall shows the percent of the relevant answers which the system also evaluates as relevant.

[2]Factoid questions are questions which require named entity as an answer. For example "How tall is the Sears Building?"

The recall of the baseline is 100% because all the correct answers contain at least one named entity of the question type. Therefore in Table 3 for the baseline algorithm we report only precision and success rate.

The overall results in the last row of Table 3 show a success rate of 86% for the content-based approach and 86.4% for the statistical. Both these results are about 22% over the baseline overall success rate. The highest performances we obtain for the category DATE - 92.6% for the content-based and 93.3% for the statistical approach. However the baseline success rate is also high - 70.2%, therefore the improvement with respect to the baseline is 22% and 23% respectively.

For PERSON and LOCATION the system demonstrates good performances (87% and 86.4% for content-based approach and 88.2%, 85.9% for statistical). Even more, the performance of both validation algorithms for these named entities exceed the baseline by about 28-30%. This is the highest increase from the baseline, therefore we may conclude that both algorithms validate best answers which belong to category PERSON or LOCATION. The MEASURE category shows the lowest success rate for both methodologies (78.3% for content-based and 78.6% for the statistical). Besides, these numbers are is only 2.2-2.5% above the baseline corresponding success rate. There are different hindrances in this kind of answers. Often measures are given in different units, to make the things more difficult, the different texts treat the numbers with different precision. These obstacles can be solved by comparing numbers and measure units with more intelligent algorithms than the simple string match.

We also carried out an experiment checking the validity of the statistical approach over the full set of 492 TREC-2001 questions. Such an experiment resulted in 81.25% success rate.

Both algorithms show similar performances. Not only the overall results are similar but also the results for the specific named entity types. We calculated that for 1164 (86.2%) [q,a] pairs both algorithms vote equally (both algorithms accept or reject the candidate answer). Moreover, for these 1164 pairs the common success rate is 92%.

Considering these figures and the negligible distance in the success rates of the two approaches (statistical prevails with only 0.4%) we may suppose that in the data-driven approaches for answer validation the data is more significant than the methodology. Although the two algorithms used two different search engines and the overlap between search engine indices is generally considered to be relatively small [3], we think that the answers of general questions can be found on the most important Internet sites, such as government or educational institution sites, Internet portals and directories, home pages of famous people, etc. Those pages should be indexed both by Google and AltaVista.

Combining approaches. The combination of the two approaches is an interesting issue. How it was stated before when both algorithms judge equally a [q,a] pair, in 92% of the cases these judgments are correct. Therefore, we created a combined approach which accepts the common judgment when the two approaches vote equally for a [q,a] pair. When the two approaches diverge in their judgment we combine their score. In this way, a small improvement of the success rate (1.4% with respect to the statistical approach) was achieved , i.e. 87.8%. We made another experiment by judging correct all the [q,a] pairs which are judged correct by one of the search engines. This algorithm obtained nearly the same success rate of 87.3%.

[3] see http://www.searchengineshowdown.com/stats/overlap.shtml for latest statistics

Although we did not perform a thoroughfully result analysis, we suppose that the two approaches diverge in their judgment when the data about [q,a] relation is is insufficient or ambiguous. For example, the question "What hemisphere is the Philippines in?" requires the answer "Eastern" and one of the wrong candidates is "Central". AltaVista obtains many pages where "hemisphere", "Philippines" and "Central" are in proximity to each other because these words often appear together in geographic texts. However by exploring Google snippets it becomes evident that "Central" appears in a great distance from "hemisphere". Whereupon, the content-based approach discards the wrong answer "Central" and the statistical approach accepts it.

7 Conclusion and Future Work

We have presented approaches to answer validation based on the intuition that the amount of implicit knowledge which connects an answer to a question can be estimated by exploiting the redundancy of Web information. Results obtained on the TREC-2001 QA corpus correlate well with the human assessment of answers' correctness and confirm that a Web-based algorithm provides a workable solution for answer validation.

Several activities are planned in the near future. First, a generate and test module based on the validation algorithm presented in this paper will be integrated in the architecture of the DIOGENE QA system under development at Irst. In order to exploit the efficiency and the reliability of the algorithm, such system will be designed trying to maximize the recall of retrieved candidate answers. Instead of performing a deep linguistic analysis of these passages, the system will delegate to the evaluation component the selection of the right answer.

We also consider the possibility combine the two approaches in more effective way.

References

[1] S. Brin and L. Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Proceedings of the 7th International World Wide Web Conference*, Brisbane, Australia, April 1998.

[2] C. Fellbaum. *WordNet, An Electronic Lexical Database.* The MIT Press, 1998.

[3] S. Harabagiu and S. Maiorano. Finding Answers in Large Collections of Texts: Paragraph Indexing + Abductive Inference. In *Proceedings of the AAAI Fall Symposium on Question Answering Systems*, pages 63–71, November 1999.

[4] B. Magnini, M. Negri, R. Prevete, and H. Tanev. Multilingual Question/Answering: the DIOGENE System. In *Proceeding of the TREC-10 Conference*, pages 335–344, Gaithesburg, MD, 2001.

[5] B. Magnini, M. Negri, R. Prevete, and H. Tanev. Is It the Right Answer? Exploiting Web Redundancy for Answer Validation. *To be published in Proceeding of ACL-02*, 2002.

[6] M. Subbotin and S. Subbotin. Patterns of Potential Answer Expressions as Clues to the Right Answers. In *Proceeding of the TREC-10 Conference*, pages 175–182, Gaithesburg, MD, 2001.

[7] P.D. Turney. Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of ECML2001*, pages 491–502, Freiburg, Germany, 2001.

[8] R. Zajac. Towards Ontological Question Answering. In *Proceedings of the ACL-2001 Workshop on Open-Domain Question Answering*, Toulouse, France, July 2001.