Bridging Languages for Question Answering: DIOGENE at CLEF-2003

Matteo Negri, **Hristo Tanev** and **Bernardo Magnini**
ITC-Irst, Centro per la Ricerca Scientifica e Tecnologica
Via Sommarive, 38050 Povo (TN), Italy
{negri,tanev,magnini}@itc.it

Abstract

This paper presents the extension of the ITC-irst DIOGENE Question Answering system towards multilinguality. DIOGENE relies on a well tested three-components architecture built in the framework of our participation in the QA track at the Text Retrieval Conference (TREC 2002). The novelty factors are represented by the enhancement of the system with language-specific tools targeted to the Italian language (*e.g.* a module in charge of the answer-type extraction, and a named entities recognizer) and the introduction of a module for the translation of Italian queries into English queries. The overall architecture of the extended system, as well as the results obtained in the CLEF-2003 Monolingual Italian and Bilingual Italian/English QA tracks will be presented and discussed throughout the paper.

1 Introduction

Research in Question Answering (QA) has received a strong boost in recent years by the QA track organized within the TREC conferences (Voorhees and Harman, 1997), which aims at assessing the capability of systems to return exact answers to open-domain English questions. The success of the initiative has reflected on the tendency of system developers to focus their activity on issues raised by the track guidelines (*e.g.*. how to deal with particular types of questions, how to pinpoint exact answers within a text document, and how to rank them according to the system's confidence). Besides the many positive effects of the TREC experience, several aspects related to the QA problem have not been faced yet. For instance, even though multilinguality has been recognized as an important issue for the future (Burger at al., 2001), up to now little has been done to provide QA systems with the capability of handling languages other than English. However, multilinguality represents a promising direction for future developments for at least two reasons. First, allowing users to interact with machines in their native languages, it would contribute to an easier, faster, and more reliable information access. Second, cross-language capabilities enable the access to information stored in language-specific text collections, that could hardly be captured by seeking only through English documents.

In the framework of an increasing interest towards multilinguality testified by the set up of a multiple-language QA track within CLEF-2003, DIOGENE represents the first concrete attempt to develop a QA system capable of dealing with Italian and English, both in monolingual and cross-language scenarios. In the former case, the Italian version of DIOGENE has been built upon the same well tested three-components architecture of the English version (Magnini et al., 2002a). Figure 1 shows the main constituents of this common backbone: these are the *question processing* component (in charge of the linguistic analysis of the input question), the *search* component (which performs the query composition and the document retrieval), and the *answer extraction* component (which extracts the final answer from the retrieved text passages). Sharing the overall architecture, as well as a number of basic tools and resources, the two monolingual versions of DIOGENE provided us with an enough flexible and reliable infrastructure for extensions towards cross-language QA as it is concerned within the CLEF-2003 framework. In particular, in order to retrieve English answers in response to Italian questions, the Italian *question processing* component has been extended with a module for keyword translation into English (also depicted in Figure 1), and then combined with the *search* and the *answer extraction* components of the English version.

The following sections will provide a general overview of our participation to the *monolingual Italian* (M-I) and *bilingual Italian/English* (B-I/E) tasks of the multiple-language QA track at CLEF-2003. In both the tasks of this new evaluation exercise systems were presented with a set of 200 Italian questions but, while for the M-I task answers had to be sought through an Italian document collection (the 193Mb corpus of the whole 1994 year of *La*

Stampa newspaper and the 85Mb corpus of the 1994 *SDA* press agency), the target collection for the B-I/E task was composed of English texts (the 425Mb corpus of the whole year 1994 of *Los Angeles Times*). Focusing on the system's architecture, Section 2 will describe the *question processing* component, while Section 3 and 4 will describe in detail the *search*, and the *answer extraction* component respectively. Finally, Section 5 and 6 will conclude the paper presenting the results of the different runs submitted to evaluation and drawing some conclusions.

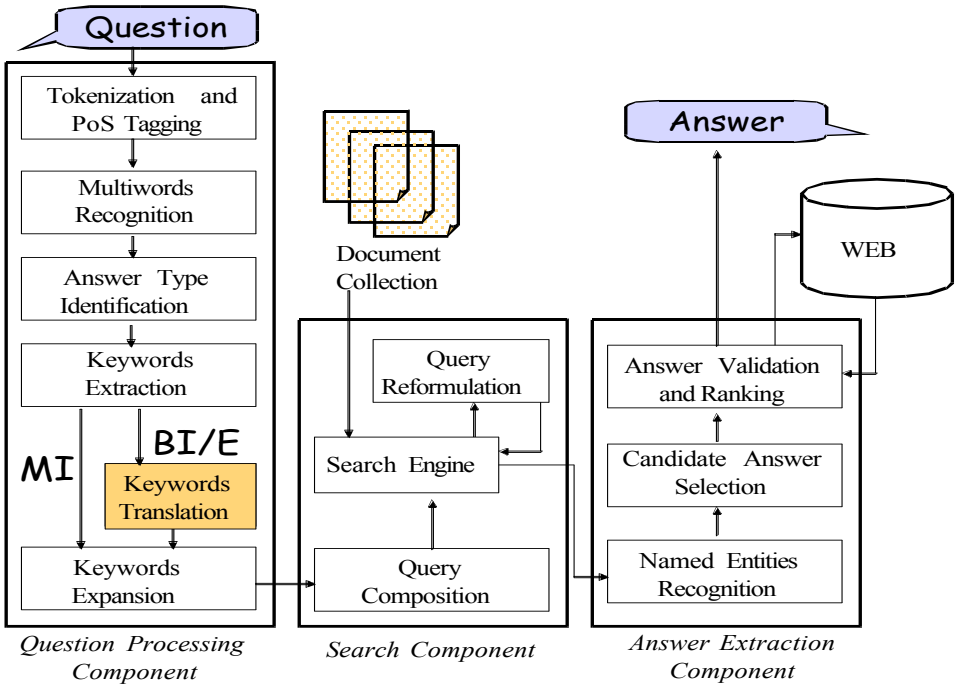

Figure1: The architecture of the DIOGENE system

2 Question Processing Component

The overall architecture set up for the ITC-irst participation to the multiple-language QA track at CLEF-2003 is an adaptation of the one described in (Magnini et al, 2002a), targeted to two specific tasks (*i.e.* M-I and B-I/E) of this new evaluation campaign. Such an adaptation was possible due to the general approach we took to develop the English version of DIOGENE, which relies on a cascade of simple, flexible and easily interchangeable modules and resources. In particular, the absence of any in-depth text analysis (most of the basic modules use a part of speech tagger as the only linguistic processor) did not require to find or develop from scratch crucial language specific tools such as, for instance, a full parser for Italian.

As for the *question processing* component, both the tasks required the substitution of the original English modules with language specific tools for dealing with the Italian language. In fact, also for the bilingual task, the addition of a module for keywords translation from Italian into English was deemed a more reliable solution than performing automatic translation of the whole question and then rely on the English version of the system.

The analysis of the input question is performed sequentially by the following modules.

- ***Tokenization and PoS tagging***. First the question is tokenized and words are disambiguated with respect to their lexical category by means of a statistical part of speech tagger developed at ITC-Irst.

- ***Multiwords recognition***. As it is done in the English version of the system, about five thousand multiwords (*i.e.* collocations, compounds and complex terms) have been automatically extracted from a monolingual Italian dictionary (Disc, 1997) and are recognized by pattern matching rules.

- ***Answer type identification***. The answer type for a question represents the entity to be searched as answer. This information will be used to select the correct answer to an input question within the documents retrieved by the search engine. In particular, knowing the category of the entity we are looking for (*e.g.* PERSON, LOCATION, DATE, etc.) we can determine if any "candidate answer" found in a document is an appropriate instantiation of that category. Answer type identification relies on a manually defined taxonomy of *answer types* (*e.g.* "LOCATION", "PERSON", "ORGANIZATION" "TIME-PERIOD", "MEASURE", "TITLE", etc.), and a set of approximately 250 rules that check different features of an input question. These rules may detect the presence of a particular word occurrence, of words of a given part of speech, and of words belonging to a given semantic category. For instance, the rule described in (1) matches any question starting with "quale" ("*what*"), whose first noun, if any, is a person.

  (1) RULENAME: QUALE-CHI
  TEST: ["quale" [¬NOUN]* [NOUN:person-p]ɹ +]
  OUTPUT: ["PERSON" J]

  Rule (1) matches questions like "*Quale Primo Ministro Britannico visitò il Sud Africa nel 1960?*" ("*What British Prime Minister visited South Africa in 1960?*", which corresponds to Q-89 of the M-I task) since the first noun encountered after "quale" (*i.e.* "Prime_Minister") satisfies the person-p constraint. The same rule does not match the question: "*Quale paese invase il Kuwait nel 1990?*" ("*What country invaded Kuwait in 1990?*", Q-50 of the M-I task), since "country" does not satisfy the person-p predicate. Following the same methodology that proved to be successful in the development of the English version of DIOGENE, semantic predicates (*e.g.* person-p, location-p, organization-p, time-p) have been defined on the MULTIWORDNET taxonomy (Pianta et *al*. 2002). Each predicate checks if the sense of a word referred in a rule is subsumed by at least one high level synset manually selected for that predicate. As an example, the predicate person-p, for which the synset persona#1 ("human being") was selected, will be satisfied by any word subsumed by this synset (*e.g.* "Prime_Minister" "philosopher" "jazzman", etc.)

- ***Keywords extraction***. At the end of the linguistic processing, a stop words filter is applied to isolate a set of "basic keywords" cutting off from the input question both non-content words and non-relevant content words.

- ***Keywords Translation (Only for B-I/E)***. Bridging languages in a cross-language QA scenario requires dealing with the big issue of looking for answers through a target document collection that is in a different language with respect to the one of the question. Such a task can be accomplished following two different approaches: the translation of the input question by means of standard machine translation techniques (Zheng 2002), or the word by word translation (Hull and Grefenstette 1996) of the question keywords. At first glance, due to the availability of a well tested English version of the system, the first solution (*i.e.* automatically translate the Italian question and then perform the whole processing as in a monolingual English scenario) seems the most promising. Besides its simplicity, automatic question translation seems favorable also because it avoids ambiguity problems inherent to the word-by-word translation, which are difficult to solve without relying on higher level linguistic analysis. However, state of the art machine translation systems still not provide good enough quality and reliability of the output. Moreover, they are not optimized for translating questions, which are usually short and do not provide sufficient contextual information for a precise translation. Therefore, some approaches (Gao et al 2001) combine dictionary-based word-by-word techniques with noun phrase extraction for a proper treatment of short questions. Adopting a similar perspective, we developed a word-by-word translation module which resorts to MULTIWORDNET and the Collins Italian/English dictionary, overcoming ambiguity difficulties by means of statistical techniques.

  Our statistical dictionary-based approach to query translation is rather similar to the one described in (Federico and Bertoldi 2002), but does not require any training. All the resources that are needed are a bilingual dictionary, a search engine, the target corpus and, if available (but not necessarily), a text corpus in the source language (Italian in our case). Due to the relatively scarce resources needed, this translation technique could be easily extended to other languages.

The starting point of our methodology is the noisy-channel model (Manning and Shutze, 1999); according to this model, if we have a text in Italian *i*, its best translation in English (*E*) is found according to the formula:

(2) $E = \arg\max_{e}(p(i \mid e).p(e))$

We applied statistical techniques to calculate the two probabilities (*p(i|e)* and *p(e)*), but further analysis of the results led us to the conclusion that *p(i|e)* has a small impact on the final results. Therefore, in order to speed up the translation process, in the final version of the system we assigned a constant value to *p(i|e)*. However here we will view how both the probabilities were estimated.

The process is carried out through the following steps.

1. First, DIOGENE extracts all the possible translations for each of the Italian keywords using the Collins Italian/English dictionary. If the translation is not found (as in the case of multiwords), MULTIWORDNET is used to translate the word. In case no translation is found and the word is capitalized, it is left as it is (this works for proper nouns, which are not listed in the dictionaries and whose coverage in MULTIWORDNET is rather limited). If the word is not capitalized and not found in Collins dictionary, nor in MULTIWORDNET, we skip it.

2. The next step is to estimate the probability of every translation in order to find the most plausible. Our algorithm has to deal with all the possible combinations of translations: even if sometimes the number of these combinations can be very large, our solution easily overcomes the problem. As for ambiguity resolution, the main resources used by the algorithm are the target English text collection (indexed at paragraph level), and the DIOGENE's search engine. Let's denote with $i=(i_1, i_2, i_3 \ldots i_n)$ a sequence of Italian keywords, and with $e=(e_{k1}, e_{k2}, \ldots e_{km})$ the set of English translations for every keyword i_k. We search in the target corpus using a Boolean query of the type:

   $(e_{11}$ OR e_{12} OR ...) AND (e_{21} OR e_{22} OR ...) AND ...AND (e_{n1} OR e_{n2} OR ...)

   This way we obtain paragraphs which contain at least one translation for every Italian keyword. If no paragraph is retrieved, a query relaxation algorithm is applied (Magnini et al 2002a) which cuts off translations for Italian keywords which are less important. The process continues until a paragraph is found which contains at least one translation for each of the remaining keywords or a half of the initial keywords have been cut off (in this case the translation process fails and NIL is returned as the final answer).

3. Then, from the paragraphs obtained, we extract translation combinations and their frequency. The probability of each combination *e* is calculated by:

   (3) $p(e = (e_{1i}, e_{2j}, e_{3k}, \ldots)) = frequency(e) / NumberParagraphsInCorpus$

   Probability $p(i|e) = p(i_1,i_2,i_3|e_1, e_2, e_3,\ldots)$ was approximated using trigram model. For example the conditional probability for five keywords has been calculated in the following way:

   $p(i_1,i_2,i_3,i_4,i_5 \mid e_1,e_2,e_3,e_4,e_5) = p(i_1 \mid e_1)\, p(i_2 \mid e_2, i_1, e_1)\, p(i_3 \mid e_3, i_1, e_1, i_2, e_2)\, p(i_4 \mid e_4, i_2, e_2, i_3, e_3)\, p(i_5 \mid e_5, i_3, e_3, i_4, e_4)$

   Our approach for calculation of the conditional probabilities relies on the following assumption: If we have a word in English *e* and a set of its possible translations in Italian denoted by *TI(e)*, we assume that every word from *TI(e)* in the Italian corpus can be translated in the word *e* when translating the corpus in English. Of course, this assumption is somehow strong and influences the accuracy of the probabilities. We assume that the appearance of *i* implies that *e* appears in the English translation of the corpus. From this assumption it follows that the formula can be rewritten in the following way:

   $p(i_1,i_2,i_3,i_4,i_5 \mid e_1,e_2,e_3,e_4,e_5) = p(i_1 \mid e_1)\, p(i_2 \mid e_2, i_1)\, p(i_3 \mid e_3, i_1, i_2)\, p(i_4 \mid e_4, i_2, i_3)\, p(i_5 \mid e_5, i_3, i_4) = p(i_1)\, p(i_2, i_1)\, p(i_3, i_1, i_2)\, p(i_4, i_2, i_3)\, p(i_5, i_3, i_4) \cdot [p(e_1) \cdot p(e_2,i_1) \cdot p(e_3,i_1,i_2) \cdot p(e_4,i_2,i_3) \cdot p(e_5,i_3,i_4)]^{-1}$

   The probabilities in the previous formula can be calculated using the Italian corpus. Probability $p(i_1, i_2, i_3)$ can be calculated counting the number of paragraphs in the Italian corpus where these words occur together. This is acquired by just one query to the index of the corpus. Probability $p(e,i_1,i_2)$ is calculated by counting the paragraphs where at least one Italian translation of *e* appears (using again the above mentioned assumption) together with i_1 and i_2.

Using just frequencies counted in two corpora (Italian and English) we find the combination of English words *E* using the noisy-channel formula (2):

However, as stated before, to speed up the translation process we simplified the calculation of *p(i|e)*. In fact, our final experiments showed that the probability *p(i|e)* has small impact on the translation quality, but significantly slows down the process. Therefore we changed the above formula into:

(4) $$E = \arg\max_{e}(p(e)) = \arg\max_{e} frequency(e)$$

This means that basically we choose combinations of English translations that have higher frequencies in the target corpus. Using the last formula, our method does not rely on any additional resource or training except for the bilingual dictionary, MULTIWORDNET (as addition to the bilingual dictionary), and the target English corpus.

As we will see in Section 5, the results achieved by DIOGENE in the B-I/E task were similar to those of the monolingual task. Assuming that the question sets in both the tasks have approximately equal level of difficulty, we may conclude that the quality of the translation achieved through this simple approach is more than satisfactory.

- ***Keyword expansion***. Both for the monolingual and the bilingual tasks, basic keywords are then passed to an expansion phase which considers both morphological derivations and synonyms.

3 Search Component

Our participation to the QA tasks within CLEF-2003, relies on the same search component developed for the English version of DIOGENE, as it is described in (Magnini et al. 2002a). The search component first combines the question keywords and their lexical expansions in a Boolean query; then performs document retrieval accessing the target document collections

The search is performed by Managing Gigabytes (MG) (Witten et al., 1999), an open-source indexing and retrieval system for text, images, and textual images covered by a GNU public license and available via ftp from *http://www.cs.mu.oz.au/mg/*. Besides the speed of the document retrieval, the advantages derived from using MG are twofold. First, it allows for the customization of the indexing procedure. As a consequence, we opted to index the document collection at the paragraph level, using the paragraph markers provided in the SGML format of the documents. This way, although no proximity operator (*e.g.* the "NEAR" operator provided by AltaVista) is implemented in MG, the paragraph index makes the "AND" Boolean operator perform proximity search. In order to divide very long paragraphs into short passages, we set 20 text lines as the limit for paragraphs' length.

The other advantage derived from using MG concerns the possibility of performing Boolean queries, thus obtaining more control over the terms that must be present in the retrieved documents. Using the Boolean query mode, at the first step of the search phase all the basic keywords are connected in a complex "AND" clause, where the term variants (morphological derivations and synonyms) are combined in an "OR" clause. As an example, given the question "*Quando morì Shapour Bakhtiar?*" ("*When did Shapour Bakhtiar die?*", Q-3 of the B-I/E task), the translated basic keywords "die", " Shapour", and " Bakhtiar" are expanded and combined into:

[**Shapour** AND **Bakhtiar** AND (**die** OR **dies** OR **died** OR **dying** OR **death** OR **deaths**)]

However, Boolean queries often tend to return too many or too few documents. To cope with this problem, we implemented a feedback loop which starts with a query containing all the relevant keywords and gradually simplifies it by ignoring some of them. Several heuristics are used by the algorithm. For example, a word is removed if the resulting query does not produce more than a fixed number of hits (this probably means that the word is significant). Other heuristics consider the capitalization of the query terms, their part of speech, their position in the question, WORDNET class, etc. (see Magnini et al., 2002c). The algorithm stops when a maximum of 50 text paragraphs has been collected or a certain percentage of the question terms has been cut off. This way, the searching algorithm builds a set of the most significant words and narrows it until enough documents are retrieved. The efficiency of these kinds of feedback loops has been recently pointed out by (Harabagiu et al., 2001).

Another problem we encountered using MG is related to the lack of language-specific stemming algorithms. Although it allows for different alternatives, derived from different combinations of case-folding and stemming modalities, English is the only language correctly handled while documents are indexed. As a consequence, as for the M-I task, the lack of an Italian stemming algorithm has reflected on a reduced precision of the document retrieval, which probably had some impact on the overall system's performance.

4 Answer Extraction Component

Once the relevant paragraphs have been retrieved, the answer extraction component first performs a rough selection of answer candidates through named entities recognition. Then, automatic answer validation procedures are applied over the selected candidates to choose and rank, according to the CLEF QA track guidelines, the three final answers to be returned by the system.

4.1 Named Entities Recognition

The named entities recognition module is in charge of identifying, within the relevant passages returned by the search engine, all the entities that match the answer type category (*e.g.* PERSON, ORGANIZATION, LOCATION, MEASURE, etc.). While the WORDNET-based named entities recognizer already developed for the English version of DIOGENE (Magnini et al. 2002d) was perfectly suitable for participating to the bilingual task, a language specific tool had to be developed from scratch to deal with Italian in the monolingual task. Also in this case, due to the availability of MULTIWORDNET and a PoS tagger for Italian, we could rely on the same approach adopted to handle English texts.

Each version of the system is based on the combination of a set of language dependent rules with a set of predicates, defined on the WORDNET/MULTIWORDNET hierarchy for the identification of both proper names (i.e. person, location and organization names, such as "Galileo Galilei", "Rome", and "Bundesbank") and *trigger words* (i.e. predicates and constructions typically associated with named entities, such as "astronomer", "capital", and "bank"). The process of recognition and identification of the named entities present in a text is carried out in three phases. The first phase (*preprocessing*) performs tokenization, PoS-tagging, and multiwords recognition of the input text. In the second phase, a set of *basic rules* (approximately 250 for the English language and 300 for Italian) is used for finding and marking with SGML tags all the possible named entities present in the text (e.g. <MEASURE><CARDINAL>*200*<\CARDINAL> *miles*<\MEASURE> *from* <LOCATION>*New York*<\LOCATION>). Finally, a set of higher level *composition rules* which is common to both the versions of the system is used to remove inclusions and overlaps among tags (e.g. <MEASURE>*200 miles*<\MEASURE> *from* <LOCATION>*New York*<\LOCATION>) as well as for co-reference resolution.

The English version of the system has been tested using the test corpora and the scoring software provided in the framework of the DARPA/NIST HUB4 evaluation exercise (Chinchor et al., 1998). Results achieved over a 365Kb test corpus of newswire texts vary among categories (ranging from an F-Measure score of 71% for the category MEASURE, to 96.5% for the category DATE), with an overall F-Measure score of 84%. As for the Italian version, experiments carried out with the same scoring software over a 77Kb text corpus[1] revealed a comparable performance, with an overall F-Measure score of 83%.

4.2 Answer Validation

The automatic answer validation module used for our participation to CLEF-2003 is the same we developed for the original English version of DIOGENE (Magnini et al. 2002a). Its reusability (which is also one of its main strengths) is due to the fully statistical approach on which the module relies, which makes it completely language-independent.

Answer validation is in charge of evaluating and scoring a maximum of 60-90 answer candidates per question in order to find the exact answer required as the final output. The top 60 (for the B-I/E task) or 90 (for the M-I task) answer candidates are selected, among the named entities matching the answer type category, on the basis of their

[1] Reference transcripts of two broadcast news shows, including a total of about 7,000 words and 322 tagged named entities, were manually produced for evaluation purposes and have been kindly provided by Marcello Federico and Vanessa Sandrini.

distance from the basic keywords and their frequency in the paragraphs retrieved by the search engine.

The basic idea behind our approach to answer validation is to identify semantic relations between concepts by mining for their tendency to co-occur in a large document collection. In this framework, considering the Web as the largest open domain text corpus containing information about almost all the different areas of the human knowledge, all the required information about the relation (if exists) between a question *q* and an answer *a* can be automatically acquired on the fly by exploiting Web data redundancy. In particular, given a question *q* and an answer *a*, it is possible to combine them in 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. Our solution to this problem relies on the definition of the more flexible idea of a *validation pattern*, in which the question and answer keywords co-occur closely. Second, we need an effective and efficient way to check the reliability of a validation pattern. With regard to this issue, our solution relies on a statistical count of Web searches. Given a question-answer pair [*q,a*] we adopted the following generic four-steps procedure for answer validation:

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 construct 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 the AltaVista search engine (http://www.altavista.com), which allows for advanced search strategies using the proximity operator "NEAR" to retrieve only Web documents where the answer and the question keywords have closer (*i.e.* within a 10 tokens window) co-occurrences. The post-processing of the results is performed by HTML parsing procedures and a simple function which calculates the *ARS* for each [*q, a*] pair by analyzing the results page returned by the search engine. The *ARS* is calculated on the basis of the number of retrieved pages by means of a statistical co-occurrence metric called *corrected conditional probability* (Magnini et al., 2002c). The formula we used is the following:

$$ARS(a) = \frac{P(Ka \mid Kq)}{P(Ka)^{2/3}} = \frac{hits(Ka\ NEAR\ Kq)}{hits(Kq) * hits(Ka)^{2/3}} * |EnglishPages| \quad (5)$$

where:

- *hits(Ka NEAR Kq)* is the number of English-language pages returned by AltaVista, where the answer keywords (*Ka*) and the question keywords (*Kq*) are in distance of no more than 10 words of each other;
- *hits(Kq)* and *hits(Ka)* are the number of English-language pages where *Kq* and *Ka* occur respectively;
- |*EnglishPages*| is the number of English pages, indexed by AltaVista.

This formula can be viewed as a modification of the Pointwise Mutual Information formula, a widely used measure that was first introduced for identifying lexical relationships (in this case the co-occurrence of *Kq* and *Ka*).

In addition to the measurement of co-occurrence frequencies, for some question types (*e.g.* Where-location or When-event) we applied patterns for answer validation. For instance, given the question "*Where is Trento?*" and the candidate answer "*Italy*", the phrases "*Trento in Italy*" and "*Trento Italy*" (which are some of the possible ways in which the answer to a question can be found in a text) are submitted to AltaVista. In these cases, the resulting number of pages is multiplied by 100 and added to the *ARS* improving its reliability.

5 Results and Discussion

The effectiveness of the extensions of DIOGENE towards multilinguality have been evaluated over four runs submitted to the monolingual Italian and the bilingual Italian/English QA tasks at CLEF-2003. The results (*strict* statistics) achieved by each run are shown in Table 1.

Run	Answers	R	W	U	X	Q with no correct A	Q with correct A	Correct NIL	**MRR**
Irstex031mi	590	132	439	6	13	103	97	2/4	0.42
Irstst032mi	**582**	**153**	**410**	**7**	**12**	**101**	**99**	**2/5**	**0.45**
Irstex031bi	492	94	378	6	14	123	77	6/49	0.32
Irstex032bi	**526**	**113**	**402**	**2**	**9**	**110**	**90**	**5/28**	**0.40**

Table 1: ITC-irst at CLEF

The first two rows of Table 1 concern the *monolingual* task, while the other two concern the *bilingual* task. Columns 2-6 show, for each submitted run, the total number of answers returned by the system (according to the CLEF QA guidelines, up to three ranked answers for each question could be output), and the number of answers that have been judged right (R), wrong (W), unsupported (U) and inexact (X). Columns 7 and 8 show the number of questions for which no correct answer was returned and the number of questions for which at least one answer was found by DIOGENE. The last two columns show the number of answers correctly marked as NIL out of the total of NIL returned, and the Mean Reciprocal Rank (MRR) obtained for each run. Differences among these figures are due to the adoption of different approaches to produce the final output. In particular, making the most of the opportunity to submit up to two runs for each task, we wanted to test the system's performance with different settings.

The monolingual version was tested in two evaluation scenarios, namely the *exact answer* task (*Irstex031mi*), where a TREC-like exact answer had to be returned for each question, and the *50 bytes* task (*Irstex032mi*), where response units with a maximal length of 50 bytes could be returned. In this second case, starting from each candidate answer, longer response units have been extracted by simply considering symmetric left and right windows covering the 50 bytes allowed. Our hypothesis was that by this way the impact of errors due to incorrect named entity tagging and, consequently, the number of inexact answers could have been reduced. Moreover, longer response units were expected to improve the quality of answers to particular classes of questions (*e.g.* non factoid questions, questions about acronyms, and those asking for entities which do not belong to the classes handled by our named entities recognizer, such as Q 127: "*Nomina un farmaco anti-malarico*" – "*Name an antimalaric drug*"), for which the system can only rely on heuristics considering the density of question keywords within the retrieved paragraphs. The evaluation of the 50 bytes run returned unexpected results, with an MRR improvement of only 3% with respect to the exact answer run. As for the number of inexact answers returned, the difference between the two runs is minimal (13 for the *exact answer* task Vs. 12 for the *50 bytes* task), showing an acceptable overall performance of the named entities tagger in the detection of entities' boundaries. Unfortunately, also the total number of questions that received at least one correct answer and the number of NIL answers are almost the same.

The two runs of the bilingual version of the system have been obtained by varying the answer validation algorithm described in the previous section, in order to test the impact of combining our statistical approach with traditional Information Retrieval metrics that take into account also the keywords density within the retrieved paragraphs. In particular, the significant improvement in the second run (*Irstex032bi*) results has been obtained by multiplying the *ARS* by a *keyword density coefficient* (Magnini et al., 2001) which considers the distance of each candidate answer from the query keywords present in a paragraph.

6 Conclusion

In this paper we overviewed the recent extensions of the DIOGENE QA system towards multilinguality. The system's backbone, built upon a cascade of simple and easily interchangeable modules, relies on a balance of shallow linguistic analysis and statistical techniques that drastically reduces the effort required to cross language barriers. Such an approach proved to be suitable both for handling languages other than English, and for bridging them in cross-language scenarios. In particular, it's worth noting how the absence of any in-depth text analysis (most of the

basic modules use a part of speech tagger as the only linguistic processor) allows for multilingual extensions without developing from scratch crucial language specific tools. Moreover, two of the main system's components, namely the answer validation module and the keyword translator set up for working in cross-language mode, rely on purely statistical approaches without requiring any language-specific knowledge. Both the modules, in fact, simply use word co-occurrences (either in a document collection or in the Web) as the main source of "knowledge".
The results achieved at the CLEF-2003 Multiple Language QA task confirm the effectiveness of our approach: both the Italian and the Italian/English version of DIOGENE performed well, producing results that are comparable to the English version of DIOGENE (the fourth ranked in the last TREC competition).

References

Burger, J., Cardie, C., Chaudhri, V., Gaizauskas, R., Harabagiu, S., Israel, D., Jacquemin, C., Lin, C.-Y., Maiorano, S., Miller, G., Moldovan, D., Ogden, B., Prager, J., Riloff, E., Singhal, A., Shrihari, R., Strzalkowski, T., Voorhees, E., Weishedel, R.: Issues Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A), (2001). URL: *http://www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc.*

Chinchor, N., Robinson, P., Brown, E.: Hub-4 Named Entity Task Definition (version 4.8). Technical Report, SAIC. http://www.nist.gov/speech/hub4_98.

Federico, M., and Bertoldi, N.: ITC-irst at CLEF 2002 Using N-best query translations for CLIR, CLEF 2002 Workshop, Rome, Italy (2002).

Gao, Jianfeng, Jian Yun Nie, Endogan Xun, Jiang Zhang, Ming Zhou, Changning Huang: Improving Query Translation for Cross-Language Information retrieval using Statistical Model. Proceedings of Conference on Research and Development in Information Retrieval (ACM SIGIR 01), New Orleans, Louisiana, USA (2001).

Harabagiu, S., Moldovan, D., Pasca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., Girjiu, R., Rus, V., Morarescu, P.: The Role of Lexico-Semantic Feedback in Open-Domain Question-Answering. Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-2001), Toulouse, France (2001).

Hull, D., and Grefenstette, G.: A dictionary-based approach to multilingual information retrieval. In Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland (1996).

Magnini, B., Negri, M., Prevete, R., Tanev, H.: Multilingual Question Answering: the DIOGENE System. Proceedings of the Tenth Text Retrieval Conference 2001 (TREC-2001), Gaithersburg, MD. (2001).

Magnini, B., Negri, M., Prevete, R., Tanev, H.: Mining Knowledge from repeated Co-occurrences: DIOGENE at TREC-2002 Proceedings of the Eleventh Text Retrieval Conference (TREC-2002), Gaithersburg, MD. (2002a).

Magnini, B., Negri, M., Prevete, R., Tanev, H.: Comparing Statistical and Content-Based Techniques for Answer Validation on the Web. Proceedings of the VIII Convegno AI*IA, Siena, Italy, (2002b).

Magnini, B., Negri, M., Prevete, R., Tanev, H.: Is It the Right Answer? Exploiting Web Redundancy for Answer Validation. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002), Philadelphia, PA. (2002c).

Magnini, B., Negri, M., Prevete, R., Tanev, H.: A WordNet-Based Approach to Named Entities Recognition. Proceedings of SemaNet02, COLING Workshop on Building and Using Semantic Networks, Taipei, Taiwan, (2002d).

Manning, C., Shutze, H.: Foundations of Statistical Natural Language Processing. MIT Press, (1999).

Pianta, E., Bentivogli, L., Girardi, C.: MultiWordNet: Developing an Aligned Multilingual Database. Proceedings of the 1st International Global WordNet Conference, Mysore, India, (2002).

Voorhees, E., Harman, D. K. Eds.: Proceedings of the Sixth Retrieval Conference (TREC-6) , Gaithersburg, MD. (1997).

Witten, I. H., Moffat, A., Bell T.: Managing Gigabytes: Compressing and Indexing Documents and Images (second ed.), Morgan Kaufmann Publishers, New York (1999).

Zhiping Zheng. AnswerBus Question Answering System. *Proceeding of HLT Human Language Technology Conference (HLT 2002).* San Diego, CA. March 24 - 27, (2002).