QA at ILC-UniPI: Description of the Prototype[*]

Francesca Bertagna[▪], Luminita Chiran[▪▪] and Maria Simi[▪▪▪]

[▪] Istituto di Linguistica Computazionale (Consiglio Nazionale delle Ricerche), Via Moruzzi 1, 56100 Pisa, Italy. francesca.bertagna@ilc.cnr.it,

[▪▪] Universita' "A. I. Cuza", Str. General Berthelot 16, 700483, Iasi, Romania. luminitachiran@yahoo.com

[▪▪▪] Dipartimento di Informatica (Università di Pisa), Via Buonarroti 2, 56100 Pisa, Italy. simi@di.unipi.it

Abstract

This paper introduces the general architecture of a prototype for monolingual Italian QA. The adopted strategies, the tools and resources for the linguistic processing are presented, together with the system results and a discussion about current limits and future directions of our work.

1. Introduction

This is the first time the Istituto di Linguistica Computazionale of the Italian National Council of Research and the Department of Computer Science at the University of Pisa take part in the QA track at CLEF. The participation at CLEF was an important occasion to finalize a first version of a prototype for Italian QA, working on a controlled set of questions and answer pairs and on a common reference corpus of news and articles. The CLEF QA track represented an important exercise to individuate the most important problems, to discuss and study possible solutions and also to share our first results in a collaborative and experimental environment. The experience gained will surely be of great importance in the further development of our work. Aim of this paper is thus twofold: on one hand we want to describe the QA prototype and its modules of analysis, on the other we would like to present the most important problems emerged and discuss possible ways to overcome them.

2. General Architecture

The system described in Fig 1. is heavily inspired by the FALCON (Harabagiu et al., 2000, Paşca, 2003) and by the PIQASso (Attardi et al., 2001) applications and it is organized following the classic three-modules architecture consisting in the question analysis, the search engine and the answer extraction modules.

In what follows we will describe in detail each of these steps, focussing in the adopted solutions and in the analysis of the encountered problems. Some important, even crucial, external modules are missing (a Named Entity Recognizer and modules for WSD and multiword recognition). We will consider this first release of the prototype as a starting point and a first assembly of different modules and resources, hoping to be able to add what is missing in the next future.

The system is organized as follows:

- in the first module, a detailed analysis of the question is performed in order to extract the information that will be of use in the QA downstream, i.e.: i) the list of the question keywords that will be used in the IR module, ii) the Question Stem and Answer Type Term, iii) the dependency representation of the question that will be compared against the dependency representation of the candidate answer, iv) the Question Focus notion that defines the type of expected answer and provides the "semantic" type of the expected answer element.

[*] We would like to thank Simone Pecunia and Giuseppe Attardi for their indispensable help and Nicoletta Calzolari and Irina Prodanof for their comments and suggestions. We also thank Roberto Bartolini, Alessandro Lenci, Simonetta Montemagni and Vito Pirrelli for the kind concession of text analysis tools.

- The second module consists of a document indexing and retrieval sub-system that takes in input the keywords of the query and provides in output a list of paragraphs matching the query .

- The last module represents the place where all the information collected during the first phase of question analysis should be used. In the future we would like to use a system of filters to rule out candidate paragraphs not satisfying a certain set of constraints (in particular semantic constrains based on the expected answer types). For the moment, only a preliminary module exploiting the dependency structure of the question and of the candidate answer has been implemented, together with the exploitation of few named entity types that can be individuated by means of simple pattern matching rules.

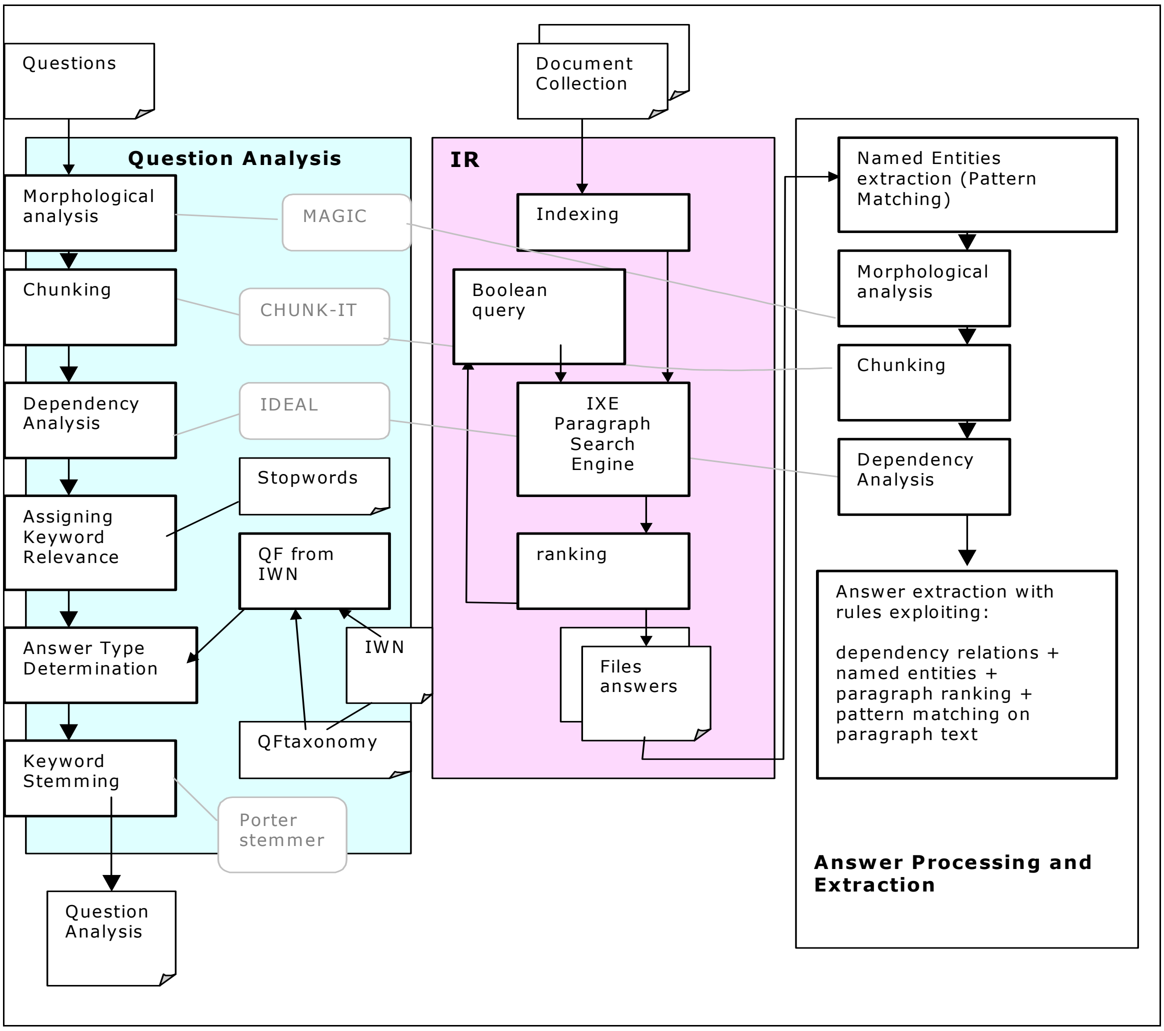


Fig 1. Prototype General Architecture

3. Question Analysis Module

In this module the system performs a multi-layered analysis of the question:

- first of all, a sequence of steps leads to the linguistic representation of the question: each word of the question is isolated, morphologically analysed and associated to one or more lemmas. Then a two-stages (chunking and dependency) syntactic analysis is performed, allowing the system to: i) segment the question into syntactically organized text units, ii) perform POS-tagging of the words in the question, iii) identify grammatical functions;

- the system applies a set of rules in order to assign to each word in the question a specific weight in term of its relevance as a keyword of the query ;
- the system extracts from the question the Question Stem (the interrogative element usually introducing the sentence) and, where needed, the Answer Type Term (Paşca, 2003);
- the Question Focus (i.e. the expected answer type) is individuated, by merely relying on the Question Stem type or by recurring, via the Answer Type Term and via the a Question Focus Taxonomy, to the information stored in the ItalWordNet database;
- a stemmer is used on some of the keywords of the query.

The next paragraphs will describe more in detail each of these steps.

3.1. Linguistic Analysis

First of all, the question goes through a chain of tools for the analysis of Italian language developed at ILC-CNR by (Bartolini et al., 2002). The analysis chain includes[1]:

- morphological analyser
- chunker
- dependency analyser

The morphological analysis is performed by Magic (Battista and Pirrelli, 1999). Magic produces, for each word form of the question, all its possible lemmas together with their morpho-syntactic features. Magic also recognizes the capitalization of the word, a small set of basic multi-word expressions (such as *al di là*[2] but also some proper names like *San Vittore* in question#3) and analyses verbs containing clitic pronouns.

The chunker, CHUNK-IT (Lenci et al., 2001), first performs the morpho-syntactic disambiguation of the question and then segments it into an unstructured sequence of syntactically organized text units (the *chunks*). We will see how also this initial, flat and *linguistically poor* syntactic representation can be exploited to extract information crucial for the task of question classification on the basis of the type of expected answer (i.e. what the user is looking for with his/her question). These information are the Question Stem (QS) and the Answer Type Term (ATT).

The chunked file is the input of IDEAL (Italian DEpendency AnaLyzer) that builds a representation of the sentence using binary, asymmetric relations (modifier, object, subject, complement etc.) between a head and a dependent based on the FAME annotation schema (Lenci et al., 2000). The success of a QA application highly depends on the quality of the parser output and very important is efficiently parsing interrogatives forms and extracting the syntactic relations that allows the system to recognize information such as direct object, subject etc. that have such an importance in the semantic interpretation of the sentence. In order to reach this goal, a specific set of rules has been written, starting with an analysis of a corpus of Italian interrogative forms.

Also the paragraphs returned by the Search Engine and candidate to be identified as answers will be subjected to these same linguistic analysis and tools.

3.2. Determining the Question Focus

The Question Stem is the interrogative element (adjective, pronoun, adverb) we find in the first chunk of the sentence (*Cosa*, *Chi*, *Quando*, etc.. [3]), while the Answer Type Term is the element modified by the QS (*Quale animale tuba?*[4] or *Quale casa automobilistica produce il "Maggiolone"?*[5]). The convergence between these two information allows us to get closer to the expected answer type and to the text portion plausibly containing the answer. Some QSs, for example *Quando* (*When*) and *Dove* (*Where*), reveal which kind of answer we can expect to receive and a set of simple rules was encoded in order to allow the system to establish univocal correspondences between them and specific QFs. Other QSs are, on the contrary, completely ambiguous: *Che* and *Quale*, being interrogative adjective, do not provide any clues about the semantic category of the expected

[1] We only mention here the tokenisation phase i.e. the pre-processing step needed to map the input sentences onto the format required by the morphological analyser.

[2] *Beyond*.

[3] *What, Who, When* etc..

[4] *What animal coos?*

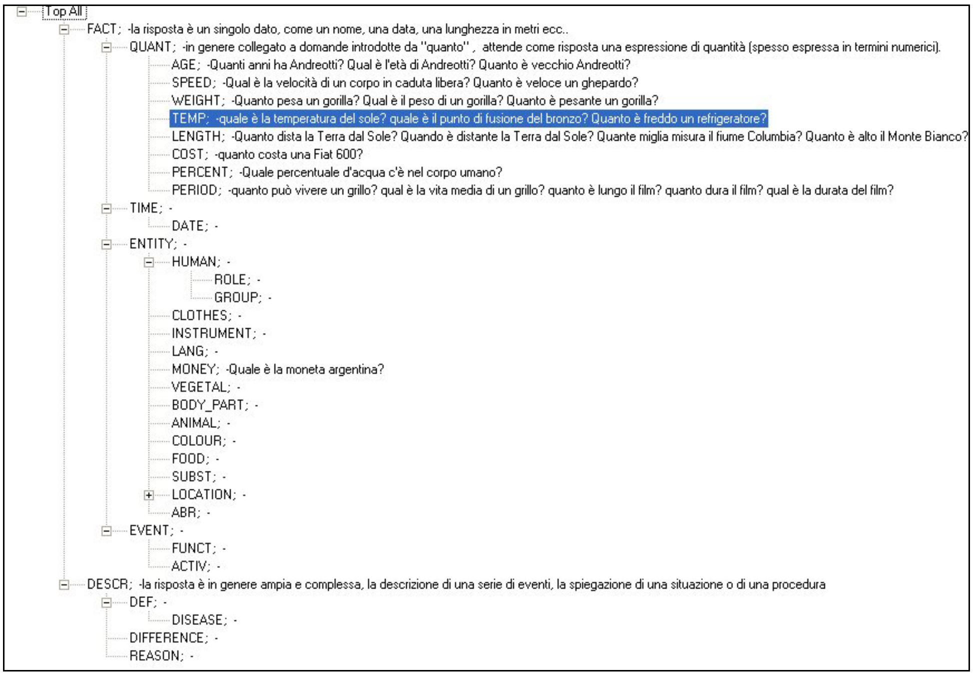
[5] *What car company produces "the Beetle"?*

answer. In these cases, to obtain the expected answer type (to individuate what we call the Question Focus) the system has to analyse the noun modified by *Che* and *Quale* and resort to their representation in the source of lexical-semantic knowledge, ItalWordNet.

ItalWordNet (IWN) (Roventini et al., 2003) is the extension of the Italian component of the EuroWordNet database (Vossen, 1999). IWN follows the linguistic design of EuroWordNet (with which shares the Interlingual Index and the Top Ontology as well as the large set of semantic relation[6]) and consists now of about 70,000 word senses organized in 50,000 *synsets*. In order to better exploit the information available in ItalWordNet, a Question Focus Taxonomy has been created and connected to ItalWordNet, allowing the system to go from the Answer Type Term to the Question Focus via the ItalWordNet hyperonymical links.

3.2.1 Question Focus Taxonomy

The Question Focus Taxonomy has been defined analysing about 500 questions obtained translating into Italian the English question collection of the QA track of the tenth Text Retrieval Conference and downloading Italian factoid questions from web sites dedicated to on-line quizes. Two disjoint types of expected answer can be identified: the first type consists of the answers referring to a single factual information (a person's name, a specific location, a length expressed in meters etc.); the second type refers to more complex answers, describing series of events, explanation, reasons etc. The highest nodes, FACT and DESCR refer respectively to these two most general categories. An exemplification of the QFTaxonomy can be observed in Fig. 2.



```
⊟ Top All
    ⊟ FACT;  -la risposta è un singolo dato, come un nome, una data, una lunghezza in metri ecc..
        ⊟ QUANT;  -in genere collegato a domande introdotte da "quanto" ,  attende come risposta una espressione di quantità (spesso espressa in termini numerici).
            AGE;  -Quanti anni ha Andreotti? Qual è l'età di Andreotti? Quanto è vecchio Andreotti?
            SPEED;  -Qual è la velocità di un corpo in caduta libera? Quanto è veloce un ghepardo?
            WEIGHT;  -Quanto pesa un gorilla? Qual è il peso di un gorilla? Quanto è pesante un gorilla?
            TEMP;  -quale è la temperatura del sole? quale è il punto di fusione del bronzo? Quanto è freddo un refrigeratore?
            LENGTH;  -Quanto dista la Terra dal Sole? Quando è distante la Terra dal Sole? Quante miglia misura il fiume Columbia? Quanto è alto il Monte Bianco?
            COST;  -quanto costa una Fiat 600?
            PERCENT;  -Quale percentuale d'acqua c'è nel corpo umano?
            PERIOD;  -quanto può vivere un grillo? qual è la vita media di un grillo? quanto è lungo il film? quanto dura il film? qual è la durata del film?
        ⊟ TIME; -
            DATE; -
        ⊟ ENTITY; -
            ⊟ HUMAN; -
                ROLE; -
                GROUP; -
            CLOTHES; -
            INSTRUMENT; -
            LANG; -
            MONEY;  -Quale è la moneta argentina?
            VEGETAL; -
            BODY_PART; -
            ANIMAL; -
            COLOUR; -
            FOOD; -
            SUBST; -
            ⊞ LOCATION; -
            ABR; -
        ⊟ EVENT; -
            FUNCT; -
            ACTIV; -
    ⊟ DESCR;  -la risposta è in genere ampia e complessa, la descrizione di una serie di eventi, la spiegazione di una situazione o di una procedura
        ⊟ DEF; -
            DISEASE; -
        DIFFERENCE; -
        REASON; -
```

Fig 2: A snapshot of the Question Focus Taxonomy

Many nodes in the QFTaxonomy have been projected on the branches of the ItalWordNet taxonomies[7] but often the QF has to be addressed on scattered and different portions of the semantic net. For example, the node Location of the Question Focus taxonomy can be mapped on the synset {luogo 1 – parte dello spazio occupata o occupabile materialmente o idealmente[8]}, that has 52 first level hyponyms and that we can further organize with other (at least) 10 sub-nodes, such as:

- country (mappable on {paese 2, nazione 2, stato 4- territorio con un governo sovrano e una propria organizzazione politica e amministrativa}),
- river, {fiume 1 – corso d'acqua},

[6] For a complete list of the available semantic relations cf. (Roventini et al., 2003).

[7] The ItalWordNet tool developed at ILC-CNR was used to encode both the QFTaxonomy and the links to IWN.

[8] *place 1- part of the space that can be ideally or physically took up.*

- region, {zona 1, terra 7, regione 1, territorio 1- una particolare regione geografica con caratteristiche proprie fisiche, naturali e culturali},
- etc.

The major part of these taxonomies is leaded by the same synset {luogo 1}, which circumscribes a large taxonomical portion that can be exploit in the QF identification. To this area we had also to add other four sub-hierarchies:

- {corso d'acqua 1, corso 4- *l'insieme delle acque in movimento*},
- {mondo 3, globo 2, corpo_celeste 1, astro 1},
- {acqua 2 – *raccolta di acqua*},
- {edificazione 2, fabbricato 1, edificio 1 – *costruzione architettonica*}.

Fig. 3 gives an idea of this situation: the circumscribed taxonomical portion includes the nodes directly mapped on the QFs, all their hyponyms (of all levels) and all the synsets linked to the hierarchy by means of the BELONGS_TO_CLASS/HAS_INSTANCE relation[9].

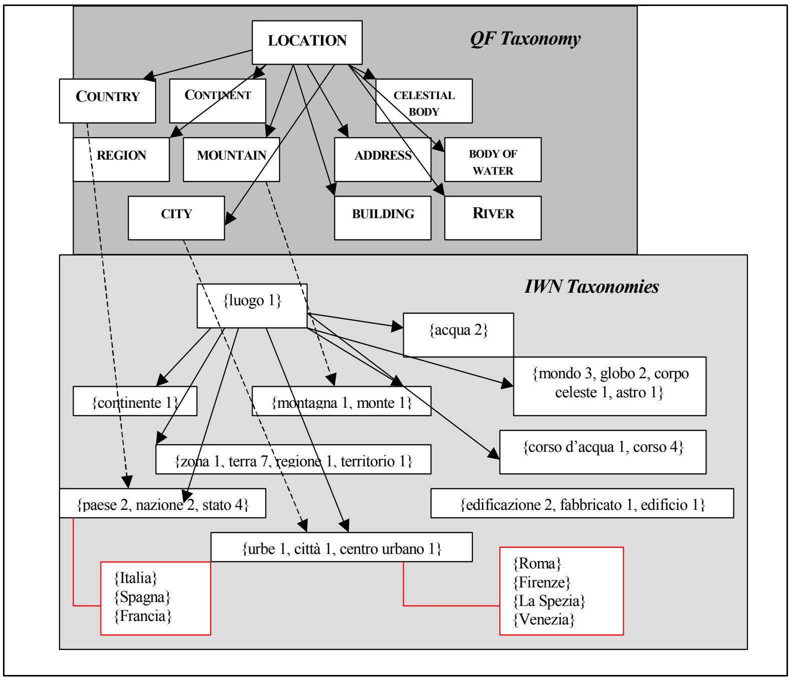


Fig 3: mapping the node Location of the QfTaxonomy on the lexical nodes of IWN

This allows a specific module of the system to retrieve the Question Focus of many question of the type *Quale* and *Che*. For example, the system identifies the Question Focus (CITY) of question#3 (*In quale citta' si trova il carcere di San Vittore?*[10]).

At the moment, no module performing Word Sense Disambiguation is available in this phase. A consequence is that the sub-module retrieves not only the relevant sense but also all the others: for example, for question#155 (*Di quale squadra di calcio francese era presidente Bernard Tapie?*[11]) beyond the correct HUMAN GROUP the system identifies an incorrect QF INSTRUMENT, determined by the fact that the ATT *squadra* has, among the other senses, also the sense of *square*. This is not a strong limit for this specific task: the Information Retrieval phase works as a kind of *implicit* Word Sense Disambiguator since in general the co-occurrence of more than

[9] While in WordNet the synsets of type instance are linked to their superordinates by means of the normal HAS_HYPERONYM relation (not distinguishing, in this way, classes from instances), in ItalWordNet the HAS_INSTANCE/BELONGS_TO_CLASS relation is used in these cases.

[10] *In what city is the San Vittore prison?*

[11] *Of which French football team was president Bernard Tapie?*

one keyword submitted to the Search Engine determines the extraction of pertinent paragraphs which exclude other readings (in this case, for example, no instruments can be found in the paragraph extracted: *Nuovi momenti difficili per l'industriale francese Bernard Tapie, ex ministro delle aree urbane, deputato e presidente della squadra di calcio di Marsiglia, l'Olympique…*[12]). On the contrary, the lack of a WSD module determined the impossibility to exploit the ItalWordNet synonyms to perform query expansion in this first version of the system.

3.3. Keyword Relevance

The selection of the keywords for the query is a very important but difficult task. For example, in the first question of the collection (*In quale anno venne conferito il premio Nobel a Thomas Mann?*[13]), we would like to submit to the search engine a vector containing at least the words: *premio*, *Nobel*, *Thomas*, *Mann*. It will be unlikely to find the word *anno* (year) in the expected paragraph (in its place we will more probably find the year we are looking for) while the word *conferito* can be easily substituted by a synonym (like *assegnato*, assigned) or by *vincere* (win) if in the answer *Thomas Mann* is indicated as the person who win the Nobel prize.

In order to deal with the majority of the cases, we adopted a general rule on the basis of the different Parts Of Speech and of the syntactic and semantic function of the word in the question. To each morphological word is assigned an attribute "relevance" which is set to the minimal value (0) if the word belongs to a list of stopwords, to the maximum value (10) if the word is a number, has a capital letter or is in inverted commas. The Part of Speech of the remaining words is analysed and an intermediate value (7) is assigned to the relevance of nouns while a smaller value (5) is assigned to verbs, adjectives and adverbs (the minimun value is assigned to auxiliary or modal verbs).

All the nouns that are "answer type terms" in questions introduced by the interrogative adjectives *Quale* and *Che* (*What*, *Which*) (for example the word *anno* in the question *In quale anno venne conferito il premio Nobel a Thomas Mann?*) received a low score (2) as well as their modifiers. This choice is not always the best strategy to follow: in case of question#17 (*A quale partito apparteneva Hitler?*[14]), submitting the keyword *partito* to the Search Engine would have significantly cut the number of the retrieved paragraphs, allowing the easy individuation of the correct answer since in the pertinent paragraphs we always find the text "*..il partito nazista..*". At the same way, the choice to assign a higher score to the ATT in case of questions introduced by *Quale* in pronominal function is very useful for questions like *Quale è la capitale della Russia?* but has some negative consequences in the case of question#31 (*Qual è la professione di James Bond?*) since it is highly unlikely to find the word *professione* in the retrieved paragraphs. Some initial observations seem to suggest that in case of questions introduced by the pronoun *Quale*, the Answer Type Terms referring to concrete entities are more likely to appear in the paragraphs containing the answer but the usefulness of a module exploiting the difference between abstract and concrete entities has still to be evaluated.

Other rules handle more specific yet frequent cases, for example assigning the minimum value to the relevance of the verb *chiamare* in question#121 (*Come si chiama la moglie di Kurt Cobain?*[15]) or of the verb *trovarsi* in question#134 (*Dove si trova l'arcipelago delle Svalbard?*[16]).

Other more subtle distinctions may be introduced: for example, the first name is *more optional* than the surname in the retrieval of the paragraphs and this is the reason for the failure of retrieval for question#28 (*Qual è il titolo del film di Stephen Frears con Glenn Close, John Malkovich e Michelle Pfeiffer?*[17]) where all the names with capital letters are submitted together (connected by AND) to the Search Engine while in the answer only the surname of John Malkovich is present. For the moment we prefer not introducing this distinction since we do not have yet a systematic and general strategy to handle proper names.

3.4. Stemming

The Porter stemmer for Italian[18] was used on all the keywords with relevance smaller than the maximum value (so in general only Proper Nouns and keywords in inverted commas were not stemmed). The use of a stemmer was preferred because it seemed more simple and straightforward than the automatic generation of

[12] *..Bernard Tapie, former minister for urban areas etc…*

[13] *What year was Thomas Mann awarded the Nobel Prize?*

[14] *What party did Hitler belong to?*

[15] *What is the name of Kurt Cobain's wife?*

[16] *Where is the Svalbard archipelago?*

[17] *What's the title of the Stephen Frears' movie with Glenn Close, John…?*

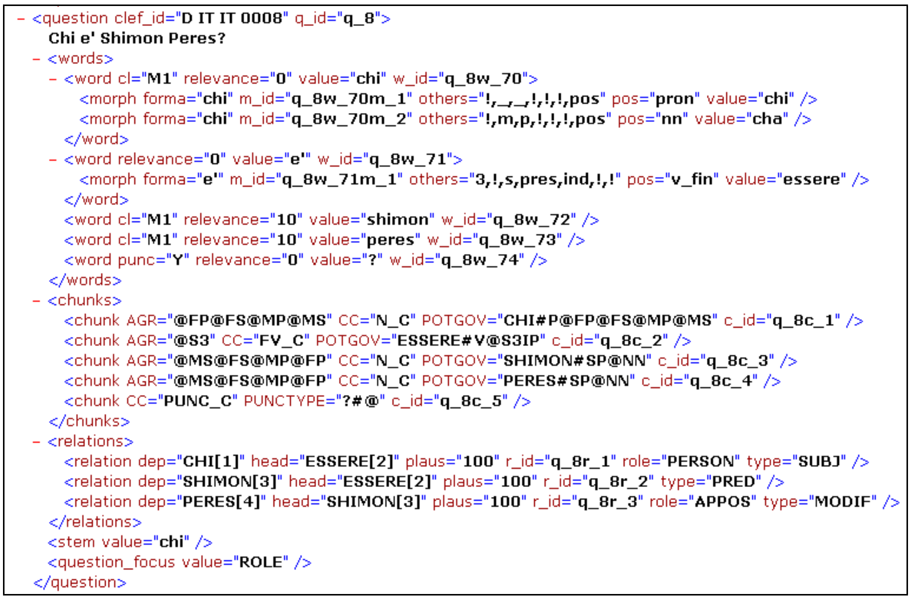
[18] Available free at http://snowball.tartarus.org/italian/stemmer.html

morphological forms but it has some important drawbacks. For example, question#127 (*Quale animale tuba?*[19]) was badly treated because the only keyword sent to the Search Engine was *tub** (the Answer Type Term *animale* was correctly omitted in the query vector). For this reason, the Search Engine retrieved a lot of non pertinent paragraphs, such as paragraphs talking about ***tub**eri* (*tuber*) or ***tub**ercolosi* (*tubercolosis*).

This would be avoided by using the morphological expansion in place of the stemmer, even if this would obviously not avoid retrieving all the document talking about the musical instrument *tuba*.

3.5. Question XML Data Structure

In order to collect all the information derived from the various steps of question analysis, we recurred to an XML representation. Fig. 2 shows an example of question represented in our XML data Structure. It would be very useful in the future fully exploiting the *ids* of the various layers of linguistic representation in order to better represent the links between morphological forms, chunks and the heads/dependents of the functional analysis. This would facilitate the identification of the text portion containing the answer in the answer extraction module.



```
- <question clef_id="D IT IT 0008" q_id="q_8">
    Chi e' Shimon Peres?
  - <words>
    - <word cl="M1" relevance="0" value="chi" w_id="q_8w_70">
        <morph forma="chi" m_id="q_8w_70m_1" others="!,_,_,!,!,!,pos" pos="pron" value="chi" />
        <morph forma="chi" m_id="q_8w_70m_2" others="!,m,p,!,!,!,pos" pos="nn" value="cha" />
      </word>
    - <word relevance="0" value="e'" w_id="q_8w_71">
        <morph forma="e'" m_id="q_8w_71m_1" others="3,!,s,pres,ind,!,!" pos="v_fin" value="essere" />
      </word>
      <word cl="M1" relevance="10" value="shimon" w_id="q_8w_72" />
      <word cl="M1" relevance="10" value="peres" w_id="q_8w_73" />
      <word punc="Y" relevance="0" value="?" w_id="q_8w_74" />
    </words>
  - <chunks>
      <chunk AGR="@FP@FS@MP@MS" CC="N_C" POTGOV="CHI#P@FP@FS@MP@MS" c_id="q_8c_1" />
      <chunk AGR="@S3" CC="FV_C" POTGOV="ESSERE#V@S3IP" c_id="q_8c_2" />
      <chunk AGR="@MS@FS@MP@FP" CC="N_C" POTGOV="SHIMON#SP@NN" c_id="q_8c_3" />
      <chunk AGR="@MS@FS@MP@FP" CC="N_C" POTGOV="PERES#SP@NN" c_id="q_8c_4" />
      <chunk CC="PUNC_C" PUNCTYPE="?#@" c_id="q_8c_5" />
    </chunks>
  - <relations>
      <relation dep="CHI[1]" head="ESSERE[2]" plaus="100" r_id="q_8r_1" role="PERSON" type="SUBJ" />
      <relation dep="SHIMON[3]" head="ESSERE[2]" plaus="100" r_id="q_8r_2" type="PRED" />
      <relation dep="PERES[4]" head="SHIMON[3]" plaus="100" r_id="q_8r_3" role="APPOS" type="MODIF" />
    </relations>
    <stem value="chi" />
    <question_focus value="ROLE" />
  </question>
```

Fig 4: The Question XML Data Structure

4. IR module and Query Definition

The inner part of the ILC-UniPi-QA system consists in a passage retrieval application built on a search engine developed at the Computer Science Department at the University of Pisa. The search engine, the same used in the PiQASso (Attardi et al., 2001) document indexing and retrieval subsystem, is based on IXE (Attardi and Cisternino, 2001), a high-performance C++ class library for building full-text search engines.

The search engine stores the full documents in compressed form and retrieves single paragraphs. However full documents are indexed and sentence boundary information is added to the index, to make possible a wider search to nearby paragraphs. In fact in many cases all the relevant terms do not appear within a paragraph, but some may be present in nearby sentences. If the option to search in a wider context is chosen, those terms may still contribute to the retrieval and ranking of the paragraph.

Whether this feature is effective with respect to a more standard strategy of paragraph indexing is still an open issue and deserves further investigation. The strategy followed to retrieve the candidate answers consists in the iteration of the boolean query on the basis of the relevance score of each keyword and of the number of retrieved documents. In the first loop we send to the Search Engine all the keywords with relevance higher than 2 connected with the AND operator. If no paragraph is retrieved than the system performs the second loop, creating a query connecting with AND all the keywords with relevance higher than 7 and with OR the keywords

[19] *What animal coos?*

with relevance 5. If no paragraphs are retrieved or if at least all the keywords in AND and one in OR are not present in the returned paragraphs than the system performs the third loop. This consists in a query with all the keywords with relevance 10 in AND and the keywords with relevance 5 in OR. Again, if no paragraphs is returned or if at least all the keywords in AND and one in OR are not present in the returned paragraphs than the fourth and last iteration is performed with only the keywords with relevance 10.

The system envisages also a mechanism to restrict the proximity in case of queries that contains a sequence of first name and surname (so the keywords *Thomas* and *Mann* of question#1 are searched in the paragraphs without any other elements in between). This scheme has to be revised and inserted in the future in a more general strategy for handling poly-lexical units of the type name+surname, name+preposition+name (the *Mostro di Firenze* of question#48) etc.

A new version of the IXE Search Engine is under development at the Uni-Pi Computer Science Department: it will allow queries constrained with information about the expected answer type, so for example in case of question#11 (*Qual è la città sacra per gli Ebrei?*[20]) it will be possible to submit a query of the type "*città sacra ebrei location:**" and retrieve only paragraphs containing the name of a city.

5. Answer Processing

The Search Engine returns a file for each query. The file returned follows a specific DTD having the paragraph as sub-element and the information about the match and the source document as attributes. The attribute "best_ranking" is also created at root element level, equivalent to the number of keywords actually submitted to IXE for the current query. For each paragraph, the system also calculates the value of the "ranking" attribute, consisting in the number of keywords of the query actually found in each single paragraph.

After this step, a set of simple regular expressions are used to discover in the paragraphs the named entities that can be found recurring to simple pattern matching; in this way, the element "Named_entity" is created for the pertinent paragraphs, having as attribute the value, the type[21] and the plausibility score of the NE identification..

The meta-information representing the *coordinates* of the journalistic article (i.e. who wrote the article, where and when and for which news agency) are eliminated from the text in order to provide a *clean* input to the text analysis tools and are saved in a specific sub-element of type "MetaInfo".

The paragraphs are then submitted to the morphological and syntactic analysers and the results are saved in specific elements.

5.1 Answer Extraction

This module is the one that most needs a serious rethinking and integration of information sources. Only few rules have been implemented in the current system, partially exploiting:

1. Dependency relations

   Some types of question (determined by the QS and by the QF) can be handled looking in the paragraphs for syntactic structures typically indicating the presence of a possible answer. This is the case, for example, of questions: i) introduced by *Chi* (Who), that can be resolved looking for relations of coordination and of modification of type adposition[22], ii) introduced by *Dove* (*Where*), that can be resolved searching among the complements of the keyword[23] introduced by the preposition *di* (*of*) or *in* (*in*)[24], iii) asking about a quantity, that can be answered searching among the modifications of "card" type. An answer identified by recurring to expected patterns of syntactic relations is probably a right answer but syntactic regularities are quite rare and the rules depend too much on the quality of the parser output.

2. Named Entities

[20] *What is the Jewish holy city ?*

[21] Year, Date, Day, Season, Time, Money, Length, Weight, Speed, HumanName and Company. Names referring to Human and Company are identified only if they are respectively preceded by abbreviations like Dott., Sig. or followed by Inc. etc..

[22] See for example question#2 - *Chi è l'amministratore delegato della Fiat?* – and the candidate answer: *Nel corso dell'assemblea dell'Ugaf, a cui ha partecipato anche l'amministratore delegato della Fiat, Cesare Romiti,…*

[23] Question: "*Dove è Bassora?*", Candidate answer: " *..sono a Bassora nel sud dell'Irak*"

[24] In case of *Dove* questions, a last check consists in verifying in IWN that the proposed answer is of type Location or that at least its PoS is of type Proper Name.

When it is not possible to rely solely on syntactic clues to individuate the answer, it would be very useful to exploit the Named Entities corresponding to the Question Focus of the question. Since for the moment the system doesn't make use of any module of NERecognition, only NEs of the type Time, Year, Day were exploited in answer extraction rules.

3. Pattern matching on the text of the paragraph

In case of *definition* questions asking about organizations, the system follows a very simple strategy consisting in extracting the text between the brackets that follows the keyword. The system accuracy over definition questions is 50%.

4. Paragraph ranking

When no other ways to individuate the answer can be found, the system provides as answer the paragraph with the highest ranking score. The 14.5% of answers judged inexact are due to this strategy.

6. Results and Future Work

The overall accuracy of the system is quite low, only 25.5% of exact answers (22.78% over Factoid questions and 50% over Definition questions). This is the first release of the prototype and many things have still to be fixed or even developed.

Between the question processing phase and the Search Engine, the system does not perform query expansion since we do not have at our disposal a WSD module to individuate the *right sense* to expand. This is the reason for the failure on question#44 (*Chi è l'inventore del televisore?*[25]), where the paragraph containing the answer is not retrieved since it doesn't contain *televisore* but its synonym *televisione*. In the future, we will concentrate our efforts on the possibility to expand the queries using the synonyms in ItalWordNet.

Moreover, it would be useful, during the question processing, being able to individuate multiword expressions, such as *unità di misura* (*unit of measurement* - question#4), *casa discografica* (*record company* - question#43), *parte dell'organismo* (*body part* - question#96), *compagnia di bandiera* (*national airline* - question#113) etc. that would allow an easier individuation of the expected answer type.

As we already said, we think that performing morphological expansion instead of stemming may be a good strategy for QA on Italian language but we are not able at the moment to exactly evaluate the cost and benefits of such a strategy change.

The Answer Extraction module is the one that most needs to be restructured and fixed. First of all, since for about 68% of the questions the expected answer is a Named Entity, the possibility to exploit the results of a NE Recognizer for making emerging important items such as names of people, organization, location etc. would be of great help. With respect to this, the opportunity to use the new version of the Search Engine under development at the Uni-Pi Computer Science Department could determine an important improvement in the system performance.

Moreover, we expected to be able to improve the overall results of the system starting to use at least the hyp(er)onyms and the synonyms of the ItalWordNet synsets in order to individuate the answer. For many questions, also without query expansion, the system was able to retrieve the "right" set of paragraphs and in some case the use of IWN relations could have helped to pinpoint the answer. For example, exploiting the IWN IS-A relation between the word *membro* (*member*) and *uomo* (*men*) could have helped to individuate the answer to question#7 (*Quanti membri della scorta sono morti nell'attentato al giudice Falcone?*[26]) in the retrieved paragraph: "*..nella strage di Capaci… dove furono uccisi il giudice Giovanni Falcone ..e tre uomini della scorta..*"[27]. At the same way, the synonymy between *causare* (*to cause*) and *provocare* (*to provoke*) on one hand and *tumore* (*tumor*) and *cancro* (*cancer*) on the other could have helped to match question and answer in case of question#64 (*Cosa può causare il tumore ai polmoni?*[28]) and the candidate answer text: "*…alimentando l'ipotesi…che gli scarichi diesel provochino il cancro*"[29]. This is something different from performing query expansion since this strategy does not enlarge the set of paragraphs that are obtained using the keywords of the question but rather helps to restrict the number of possible candidates[30].

[25] *Who is the inventor of the television?*

[26] *How many members of the escort died in the attack to Judge Falcone?*

[27] *..in the Capaci massacre…where Judge Falcone..and three men of his escort died..*

[28] *What causes lungs tumor?*

[29] *..it fosters the hypothesis that…diesel exhaust provokes cancer*

[30] In this case, the lack of a module for explicit WSD would not effect the identification of useful connections.

As final remark, we think that CLEF represented a very important occasion to highlight the problems and to look for new solutions and strategies for Italian QA. In the next future, we will work on a new release of the system in order to overcome its current limits and to improve its performance.

References

Attardi G., Cisternino A., *Reflection support by means of template metaprogramming*, Proceedings of Third International Conference on Generative and Component-Based Software Engineering, LNCS, Springer-Verlag, Berlin, 2001.

Attardi G., Costernino A., Formica F., Simi M., Tommasi A., Zavattari C., *PIQAsso: Pisa Question answering System*, in Proceeding of the 10th TREC Conference, 2001.

Bartolini R., Lenci A., Montemagni S., Pirrelli V., *Grammar and Lexicon in the Robust Parsing of Italian: Towards a Non-Naïve Interplay*, in Proceedings of COLING 2002 Workshop on Grammar Engineering and Evaluation, Taipei, Taiwan, 2002.

Battista M, Pirrelli V., *Una Piattaforma di Morfologia Computazionale per l'Analisi e la Generazione delle Parole Italiane*, ILC-CNR Technical Report, 1999.

Harabagiu S., Moldovan D., Pasca M, Mihalcea R., Surdeanu M., Bunescu R., Girju R., Rus R. and Morarescu P., *FALCON: Boosting Knowledge for Answer Engines*, in Proceedings of the Text Retrieval Conference (TREC-9), 2000.

Lenci A., Montemagni S., Pirrelli V., *CHUNK-IT. An Italian Shallow Parser for Robust Syntactic Annotation*, in Linguistica Computazionale, Istituti Editoriali e Poligrafici Internazionali, Pisa-Roma, ISSN 0392-6907, 2001.

Lenci A., Montemagni S., Pirrelli V., Soria C., *FAME: a Functional Annotated Meta-Schema for multi-modal and multilingual Parsine Evaluation*, Proceeding of the LREC-2000, 2000.

Paşca M., *Open-Domain Question Answering from Large Text Collections*, CSLI Studies in Computational Linguistics, USA, 2003.

Roventini A., Alonge A., Bertagna F., Calzolari N., Girardi C., Magnini B., Marinelli R., Speranza M., Zampolli A., *ItalWordNet: Building a Large Semantic Database for the Automatic Treatment of Italian*. In Zampolli A., Calzolari N., Cignoni L. (eds.), Computational Linguistics in Pisa, Special Issue of Linguistica Computazionale, Vol. XVIII-XIX, Istituto Editoriale e Poligrafico Internazionale, Pisa-roma, 2003.

Vossen, P. (ed.), *EuroWordNet General Document*, 1999. http://www.hum.uva.nl/~ewn.