When the Answer comes into Question: Survey and open issues

Ana Cristina Mendes and Luísa Coheur

Spoken Language Systems Laboratory - L^2F/INESC-ID
Instituto Superior Técnico, Technical University of Lisbon
R. Alves Redol, 9 - 2º – 1000-029 Lisboa, Portugal
ana.mendes@l2f.inesc-id.pt – luisa.coheur@l2f.inesc-id.pt

Abstract

Although the quality of an answer determines the success of a Question-Answering system, in current literature only a few systems put some special focus on the answering phase. Typically, answer candidates are directly extracted from the information sources and a redundancy-based approach is used: the most frequent candidate is considered to be the final answer. However, this process has many pitfalls. For instance, if the system is not able to find equivalences between answers, the frequency of candidate answers might be erroneously calculated; also, the way an answer is generated should take the user into consideration, thus enabling the usage of appropriate vocabulary and information detail.

This survey covers current efforts on the answering problematic in Question-Answering, under three different lines of research. First, we present several works that focus on relating candidate answers. As we will see, besides equivalence, other relations hold between candidate answers, that should be taken into account in order to accurately compute the system's final answer. Then, we recover the concept introduced by Gaasterland, Godfrey and Minker (1992) of cooperative answer, that is, a correct, useful and non-misleading answer, and we bring up current attempts that target this concept. Finally, we investigate the Question-Answering community endeavours on answer generation, as research in this direction can complement the aforementioned approach based on the direct extraction of answers.

Despite a few first steps that have been taken addressing this problematic, there are still many open issues. Nevertheless, in many related areas, like Information Retrieval or Opinion Mining, solutions for similar problems have already been developed and successfully put into practice. Thus, throughout this survey and whenever appropriate, we make the bridge to other research areas, describing their techniques to solve identical research questions, that can be brought in Question-Answering for the purpose of answering.

1 Introduction

Current Question-Answering (QA) systems search after the correct answer that solves the input question. To this end, a common approach to factual questions relies on the direct extraction of candidate answers and the leverage of the redundancy of the Web, or of document collections (Lin 2007). A score is attributed to each candidate, usually as a function of its frequency, and the final answer is selected from

the stack of candidates sorted in decreasing order of score, in which the topmost is considered to be the final, and correct, answer.

In the following, we address the problematic of answering in QA systems. We start from the common answer extraction and selection approach, as previously described, and survey systems that took a step further on three different dimensions: relating candidate answers, targeting the cooperative answer and generating the final answer.

The redundancy-based approach is recurrent in the development of QA systems that use free text data as information source; however, it only works if a way of **relating the candidate answers** is devised. As noted by Dalmas and Webber (2007), candidates are commonly seen as competitors, and the relationships between them are not considered in the process of selecting the final answer. However, it should be clear that if a system is not able to understand that, for instance, *Oct. 14, 1947* and *14th October, 1947* are equivalent candidates for the question *"When did the test pilot Chuck Yeager break the sound barrier?"*,[1] the topmost candidate answer will probably be miscalculated. Even if most of the systems employ clustering methods that relate equivalent answers, the relations established between candidates are usually confined to equivalence. Nevertheless, considering the question *"What animal is Kermit?"*, candidates *frog* and *amphibian* should also be related, rather than dealt with separately. Thus, to achieve the correct answer to a given question, QA systems need not only to be able to identify groups of answers, but also to understand the relations between them. Some systems already perform (part of) this step, as it is the case of the work done by Moriceau (2005) where the extracted candidate answers are standardized, a procedure which can later help the detection of relations.

When it comes to picking the final answer, trends in QA see the retrieval of the correct answer as the systems' ultimate goal; however, 'providing correct answers to users is simply not enough'. This sentence motivated the work on cooperative answering by Gaasterland, Godfrey and Minker (1992), where the notion of **cooperative answer** was introduced, based previous studies on cooperative behaviour in human conversation (Grice 1975). The authors defend that, in order to an information system to exhibit cooperative behaviour, it must hold the same characteristics as observed between human beings. Being so, they state that better than a correct answer, is a cooperative answer, that is, a **correct**, **non-misleading** and **useful** answer. Finding the cooperative answer introduces a higher level of complexity that systems must deal with: in this case, not only one property of the answer is on stake, but three. Moreover, the way how these properties relate is far from trivial. For instance, a correct answer might not be useful, and an useful answer might not be correct. As an example, consider the question: *"Where did Ayrton Senna have the accident that caused his death?"*. *In Imola*, *in Italy* and *in Europe* are correct answers, but the third possibility is probably not useful. By the same token, for an IT student the useful answer to the question *"How many kilobytes are there in*

[1] The majority of the examples in this survey are taken from (Magnini et al 2003; Magnini et al 2005; Moriceau 2005).

a gigabyte?" would be the correct one: *1,048,576 KB*; however, for someone who wants to have an idea of the magnitude of that amount, the useful answer could be simply *1,000,000 KB*, which is not correct. The YourQA system (Quarteroni and Manandhar 2007) took the first steps in this direction, by adapting the its answers to the user's previously defined profile. Indeed, and despite this example, not many systems in the literature are stated as covering this problematic.

The job of a QA system is not complete until it answers back to the posed question, which is typically done through the direct retrieval of an information chunk extracted from the corpora sources. Even if, at first sight, this can be thought as *being enough*, there is plenty of information that can be part of the answer which is simply disregarded. This is the case, for instance, of the degree of confidence of the system in the given answer. In this context, we should not disregard the amount of work done in Natural Language Generation (NLG), from which QA has no yet taken the proper benefits. Being so, the last step for a QA system is to **generate the answer**. Here, Moriceau (2005) is a rare example, with an approach that lexicalizes and computes the answer to be returned to the user.

In this paper, we survey the state of the art in the process of answering in QA, and adopt the concept of *cooperative answer* as previously presented. This survey is composed by three main sections, each corresponding to the slope of answering in focus: relating answers, targeting the cooperative answer and generating the answer. Moreover, since there are still many loose ends concerning this topic, whenever possible we will make the bridge to other areas that face with similar problems, and from which the employed technique can be borrowed.

This paper is organized as follows: Section 2 presents a general typology of relations and describes several techniques for detecting those relations between candidate answers, Section 3 addresses the need of targeting the cooperative answer; and Section 4 focuses on techniques and systems that generate the answer. The paper finishes in Section 5, in which the conclusions are drawn, and where we discuss open issues and point to future directions in QA.

2 Relating candidate answers

In this section, we describe a general typology of relations between answers, like described by Moriceau (2005) and originally introduced by Webber, Gardent and Bos (2002). Regard, however, that both previous works consider the relations in the context of correct answers, assuming a filtering phase where the incorrect ones are discarded. Here, we discuss the relations between candidate answers, how they can influence the computation of the correct answer, and present techniques for their detection.

2.1 A typology of relations

Relations between two candidate answers can be of:

Equivalence – if answers are consistent and entail mutually, namely:

1. answers with notational variations. For instance, *Oct. 14, 1947* and *14th October, 1947* are equivalent answers for *"When did the test pilot Chuck Yeager break the sound barrier?"*;
2. answers that rephrase others, like synonyms or paraphrases. For example, the question *"How did Jack Unterweger die?"* can be answered with *committed suicide* or *killed himself*.

Inclusion – if answers are consistent and differ in specificity, one entailing the other, through:

1. hypernymy, that is, the answers are in a *is-a* relation. For example, *"What animal is Kermit?"* can be answered with *frog* or *amphibian*;
2. meronymy, that is, the answers are in a *part-of* relation. For example, *"Where did Ayrton Senna have the accident that caused his death?"*, in which *Imola*, *Italy*, *Europe* and *earth* are possible answers;
3. membership, that is, the answers are in a *is-a-member-of* relation. For example, *Edvard Munch* is a member of *a Norwegian Symbolist painter*, both being possible answers to *"Who painted the "Scream"?"*.

Aggregation – if answers are consistent and not entailing. All candidate answers are potentially correct and can be integrated in the form of a conjunction. For example, the question *"What is Kermit?"* can be answered with *frog* or *muppet*, or a conjunction of both: *frog and muppet*;

Alternative – if answers are not consistent and not entailing. In the case of questions with unique answers, only one of them can be correct. For example, the question *"How is the Pope?"* can be answered with *ill* or *healthy*, but not with both. In the case of questions with multiple answers, all the alternatives may be distinct answers. For example, *twenty-eight* and *twenty-nine* are alternative answers to the question *"How many days are in February?"*.

Given the described typology, Figure 1 depicts the relations that hold for a set of possible candidate answers to the question *"What is Kermit?"*.

2.2 Techniques for relating answers

In the following we discuss how to encounter relations between answers and by means of which techniques. We also point to actual research on this thematic.

Relations of equivalence are established between lexicographically different answers that represent the same entity or concept.

One technique to detect equivalence relies on string distance metrics – like the Levenshtein distance, the cosine or Jaccard similarities (Cohen, Ravikumar and Fienberg 2003). Despite its simplicity, this technique presents some limitations: for instance, the relation of equivalence existing between *George Bush* and *George W. Bush* can not be recognized on the same way as *John Kennedy* and *John F. Kennedy*. In the former case, context is needed in order to decide if they refer to the same person or to two different persons.

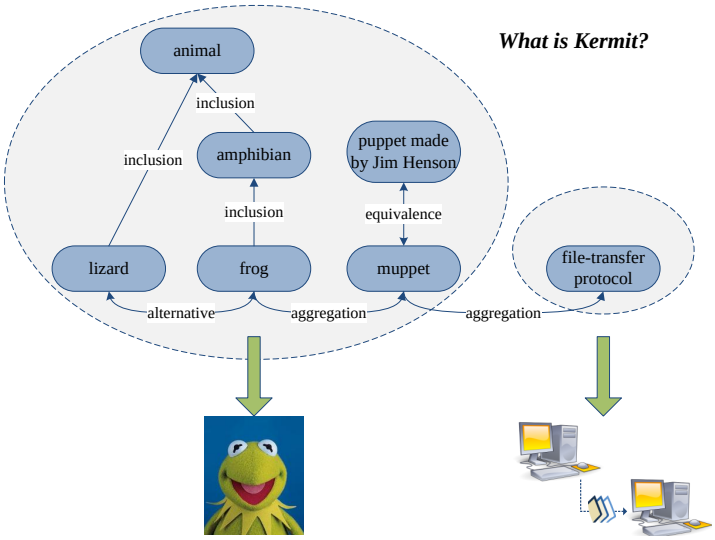

Fig. 1. A typology of relations to organize candidate answers.

Another technique relies on normalization, in which the goal is to reduce the spelling variations of an entity by finding its canonical unambiguous referent. Moriceau uses normalization on two distinct works that address the integration of answers belonging to the categories DATE and NUMBER: in the first (Moriceau 2005), the author assumes that answers are either temporal points (*e.g. September 16th, 1989*) or intervals (*e.g. September 1989, from 10 to 22*), and normalizes them both to the form of intervals (*[16-9-1989,16-9-1989]* and *[10-9-1989,22-9-1989]*, respectively); in the second (Moriceau 2006), the author builds frames to represent each numerical answer, containing all its relevant properties, namely the unit of measure, precision and numerical value. Another example is the system of Chu-Carroll, Czuba, Prager and Ittycheriah (2003) that uses named entity normalization in order to compare and combine the candidates retrieved from two answering agents. As it is employed in the aforementioned works, normalization allows the comparison and integration of different answers, that can be originated from distinct sources. In addition to this, the normalization of named entities has also proved to aid text retrieval for QA (Khalid, Jijkoun and de Rijke 2008). In this case, instead of relating candidate answers, normalization typically precedes the answer extraction phase, allowing the identification of the diverse candidates that refer to the same entity.

The techniques employed for recognizing paraphrases can also be considered in the detection of equivalences between answers; here, and like in named entity normalization, the common approach is to use them prior to the extraction of answers (France, Yvon and Collin 2003; Takahashi, Nawata, Inui and Matsumoto 2003).

When it comes to resources, knowledge repositories are often valuable in the

detection of answer equivalence. For instance, together with several string distance metrics and a set of hand-crafted rules, the framework of Ko, Si and Nyberg (2007) for answer selection in QA relies on lists of synonyms collected from three knowledge bases – WordNet (Fellbaum 1998),[2] Wikipedia[3] and the CIA World Factbook[4] – in order to discover similar candidate answers.

Relations of Inclusion occur between two candidates that represent different entities or concepts, in which one includes or subsumes the other.

To detect inclusion, the lexical relations in the WordNet can be exploited. This resource was considered in the work by Dalmas and Webber (2007), which organizes answer candidates on the geographical domain into directed graphs. Each candidate answer is a node in the graph, and each edge is labelled with the type of relation that exists between the two nodes (candidate answers) it connects. Inclusion relations are captured using the *hypernymy/hyponymy*, *part of* and *member of* pointers of the WordNet, and also by string matching: for instance, *Pacific* is considered as including *western Pacific*. (Equivalence, as previously seen, is also detected by means of string matching, acronym recognition and synonyms from WordNet.)

Within the field of QA, several works benefit from the existence of this relation between words for a variety of applications. In the classification of questions, Huang, Thint and Qin (2008) use the hypernym (found in the WordNet) of the question headword to feed a machine learning-based classifier. In the classification of answers, when the category of a candidate answer can not be found in a dictionary, the QUARK system (Nii, Kawata, Yoshida, Sakai and Masuyama 2004) uses the category of its hypernym, found by means of lexical patterns.

The detection of semantic relations, like the inclusion, between words has been widely explored and addressed for **knowledge acquisition**. Examples are the seminal approach by Hearst (1992), followed by several other related works, like (Pantel and Ravichandran 2004) or (Ritter, Soderland and Etzioni 2009). The goal is here to acquire the relations and their participants present in text; nevertheless, the employed techniques can be transposed to the task of deciding if a relation of inclusion between two candidates exists.

Relations of Aggregation and Alternative exist between candidates that represent different entities/concepts. In aggregation, candidates can together be the answer to the question; in alternative, picking one as the final answer, may imply the others to be discarded. Moreover, and contrary to the previous two relations, these do not exist independently from the posed question: for for the question *"Name one animal."* the relation existing between *frog* and *lizard* is aggregation, but the same candidate answers are alternatives for

[2] Available at `http://wordnet.princeton.edu/`. Last accessed on September 2010.
[3] Available at `http://wikipedia.org/`. Last accessed on September 2010.
[4] Available at `https://www.cia.gov/library/publications/the-world-factbook/`. Last accessed on September 2010.

the question *"What animal is Kermit?"*. Being so, these relations rely on a deeper analysis of the question, than the equivalence and inclusion relations. Typically, QA systems assume that, if two candidates are not equivalent, then they are alternative to each other: once again, the view that sees each candidate as a competitor (Dalmas and Webber 2007). The detection of alternatives requires strategies not often explored in QA, but employed in other natural language processing tasks, that can be used in QA.

Alternative between two candidates can be seen as antonymy, that is, if two answers are antonyms, then they might hold a relation of alternative. Mohammad, Dorr and Hirst (2008) take a pair of words and apply an unsupervised method to determine if they are antonyms and, if so, its degree of antonymy. Other works can be referred whose goal is to detect contradiction among words: based on the WordNet lexical chains (Harabagiu, Hickl and Lacatusu 2006), based on patterns (Lin, Zhao, Qin and Zhou 2003) or based on supervised learning approaches (Turney 2008). When it comes to the effective application of antonyms, the system described by Isozaki (2004) uses them to detect important paraphrases (*e.g.*, *Ann is the wife of Bob* is a paraphrase of *Bob is the husband of Ann*) for the purpose of query expansion.

Different polarity between answer candidates can also indicate the existence of a relation of alternative. The notion of polarity is often applicable in the context of **opinion QA**, where the questions focus on people's opinions towards certain facts or events and, in a broader context, in **opinion mining** (Balahur and Montoyo 2008). Moreover, the detection of polarity among words and word sequences has been object of study in various natural language tasks, like **sentiment analysis** (Wilson, Wiebe and Hoffmann 2009) and **textual entailment** (Danescu-Niculescu-Mizil, Lee and Ducott 2009).

The difficulty in recognizing alternative relations goes beyond purely detecting antonymous or differences of polarity. Consider, for instance, the alternative answers *frog* and *lizard* to the example question: *What is Kermit?*. Probably due to these obstables, allied to the fact that typically the relation of alternative is *a priori* assumed to exist between candidates, to our best knowledge no work in QA has yet focused on the effective detection of alternative and aggregation. Despite that, and according to Moriceau (2007), these relations are the most frequent among those established between correct candidate answers (the author presented this conclusion after analysing the relations between all correct candidate answers to 180 questions).

3 Targeting the *cooperative* answer

In this section, we survey the actual research towards cooperative answers, that is, correct, non-misleading and useful answers.

3.1 Correct answers

Since 2006, the Answer Validation Exercise in the Cross Language Evaluation Forum (CLEF) exhorts systems to automatically verify the correctness of an answer to a question, given a snippet that supports it. Most systems follow an approach based on textual-entailment, that is, first an hypothesis is built with the given question and answer, which is afterwards tested against the text of the snippet.[5] In order to decide about the correctness of the answer, the competing systems use a large variety of approaches that can be roughly split into two categories:

- based on lexico-syntactic information – relying on the analysis of syntactic dependencies and match of syntactic trees. For instance, Iftene and Balahur (2008) compute a fitness value based on the matching between the named entities on the hypothesis and the entities of the supporting text.
- based on machine learning – using a training corpus and a set of features, such as the Levenshtein distance or the size of the longest common subsequence between the hypothesis and the supporting snippet (Téllez-Valero, Juárez-González, Montes-y-Gómez and Villaseñor-Pineda 2008; Kozareva, Vazquez and Montoyo 2006).

It is also commonly agreed that a successful **question classification** (defined as the process that maps a question into a predefined category, representing the type of the information expected to be in the final answer) helps marking presumably correct answers while filtering out the incorrect ones. For instance, the question *"What animal is Kermit?"* asks for the character in *The Muppet Show*, and not the file-transfer protocol. Thus, question classification is an important task when deciding about the correctness of an answer: if the category ANIMAL is assigned to the question above, *frog* and *lizard* are correct candidates, while *muppet*, *puppet made by Jim Hendson* or *file-transfer protocol* are not.[6] Moreover, the relations established between different answers, as described in Section 2, can further reduce the complexity of this problem to the identification of alternatives. For instance, again in the question above, since there is a relation of alternative between *frog* and *lizard*, either one or the other is correct, but not both.

Although an answer that does not comply with the question category is most certainly wrong, the contrary is necessarily true: an answer that agrees with the question category is not surely correct. As presented before, redundancy can provide some hints towards the correction of a candidate answer (Downey, Etzioni and Soderl 2005; Magnini, Negri, Prevete and Tanev 2002). However, systems should be aware that information sources like the Web contain pitfalls that are usually not present when using databases or newspaper collections: those pitfalls are, for instance, users' opinions stated in community portals, fora or blogs and low quality documents (from bad grammar to imprecise or defective content). That is to say,

[5] In 2006, the hypothesis was given instead of the pair question/answer. In the following years, each system was given a triple (question, answer, snippet).

[6] For a detailed analysis of the impact of question classification in open domain question answering see (Moldovan, Paşca, Harabagiu and Surdeanu 2003).

the decision about the correctness of an answer goes beyond merely confirming its accordance with the question: also where it came from has to be taken into consideration. This lead us to the notion of credibility, defined as a measure of the reliability of a candidate with respect to a question and the source text, and arising as a required property of a correct answer. Credibility is deeply intertwined with the *believability* of some information and/or its source (Metzger 2007) and is certainly an open research line in QA.

Some efforts have been done regarding the integration of credibility into current systems. Again, and like in deciding about the answer's correctness, redundancy plays a major role in attributing credibility to an answer, that is, a popular answer is correct and credible. The method proposed by Wu and Marian (2007) on the corroboration of answers from multiple sources, uses several parameters to score candidates, like their global frequency, their importance within a web page, and the importance of the web page itself – by means of the relevance assigned by the search engine that retrieved it and the presence of duplicates. A different idea that takes into account the data intrinsic to the source of information is presented by Sacaleanu and Neumann (2006). The authors describe a wide variety of static and dynamic metadata that can be collected from the information source and exploited when validating the answer, namely: the name of the host school or organization, textual fingerprints of authority, and the text structure and style. Whether the amount of information gathered should differ according to the posed questions and candidate answers is left as open issue.

Recently, the issue of credibility has had a particularly great impact in **community QA**, where the role of QA system is played by a person and which has recently attracted much interest of users and researchers. The QA portal *Yahoo! Answers*[7] relies on a reputation model in which the questioner can choose the best answer and, if not, the answerers can be voted by the community as having given the best answer. In the context of this portal, Dom and Paranjpe (2008) developed a statistical approach to estimate the credibility of an answerer, defining it as the probability of that user being awarded the best answer on determined set of questions.

3.2 Non-misleading answers

A non-misleading answer prevents the user to create a wrong interpretation about the topic under consideration. Here we focus in two major types of answers that can mislead the user when dealing with an open-domain QA system over large collections of text: *1)* ambiguous answers, and *2)* answers that present an erroneous granularity.

[7] Available at `http://answers.yahoo.com/`. Last accessed on September 2010.

3.2.1 Ambiguous answers

Natural language is inherently ambiguous and QA systems are specially mindful to this problematic. Ambiguity, as it is commonly considered in QA, arises either from corpora sources or user questions. Systems that do try to cope with the problematic of ambiguity in the user's question, usually push its resolution to the user side. A common strategy is to interact with the user through *clarification dialogues*, in order to refine the posed question, like approached by Quintano and Rodrigues (2008) and De Boni and Manandhar (2005). The ambiguity in answers, however, is usually not addressed. Returning an ambiguous answer to an unambiguous question is not cooperative, since it leaves room to multiple interpretations, instead of being a clear statement of the semantic meaning of the answer. Regard the next question and a possible answer to it:

Q: *"Who was the President of the US during the Gulf War?"*
A: *President Bush.*

The ambiguity exists since, so far, there were two presidents of the United States called Bush, and the answer does not clearly define which one responds to the question.

Another interesting example is reported by Dalmas and Webber (2007) in which both question and answer are ambiguous:

Q: *"Where is the Danube found?"*
A: *Bulgaria*

Danube can either refer to the Danube River or the Danube breed of horses. Knowing that Bulgaria hosts both (the Danube River flows through Bulgaria, and Danube horses are found in that country), the given answer is ambiguous: it does not indicate to which entity (river or breed of horses) it is referring.

No major problem seems to accrue from the previous example (the user can keep his interpretation of the answer). However, that is surely not appropriate in certain scenarios, like in e-learning. In this context, Hung, Wang, Yang, Chiu and Yee (2005) describe their hybrid method to sense disambiguation of Q/A pairs (stored in a knowledge base) and the posed question, based on the WordNet synsets and on the number of retrieved results by a web search engine. Although the main goal was to overcome the need of students to pose their questions using the same word forms as they exist in the set of questions previously stored, this work is relevant in the sense that each answer is also semantically disambiguated (even if together with its question). Despite this effort, research is still lean on this problematic.

3.2.2 The granularity problem

Whereas ambiguity plays a role of great interest to QA systems' researchers and developers mainly on the question side, the problematic of wrong granularity, when considered, is observed from the answer side. Regard the following two examples:

Q: *"How many people live in London?"*
A: *more than 5 thousand*

Q: *"How many people live in London?"*
A: *more than 5 million*

The issue here is not related with the correctness of the answers (since both are correct), but to decide which one should be presented to the user. A strategy could be to choose the most specific, however for the case of *"What are the components of a car?"*, *molecules* is probably not a desirable answer.

The problem with granularity is that, in many situations, the decision about which answer to choose is fuzzy, bringing problems to the evaluation of the results produced by systems: in certain cases, even among human assessors there is no agreement about what is the answer to a given question (Voorhees and Tice 1999). This also leads to problems when building test collections for QA, as Lin and Katz (2006) described. The authors discuss the characteristics of a correct answer and mention that granularity is a critical issue, since it makes the judgement dependent not only from the human assessor, but also from the question itself, more precisely if it belongs to the categories PERSON, LOCATION or DATE. For instance, is a person surname enough to answer a PERSON question, is a country enough to a LOCATION question, and is a year enough to a DATE question?

Leidner, Sinclair and Webber (2003) briefly discuss the implications of granularity in spacial named entities for the tasks of map generation and QA, while pointing to previous work of Shanon (1979). Here, the author advocates that the granularity of the answers to *where*-questions depends on the reference points of both speaker and listener. For the general case, Breck, Burger, Ferro, Hirschman, House, Light and Mani (2000) proposed the definition of a set of directives to guide the evaluation of certain questions in order to reduce the uncertainty in evaluating answers and minimize the deregulation of the judgement, in a work where an experimental automated evaluation system is presented.

On the one hand, the granularity of the cooperative answer depends on the question and on other factors – like the questioner's position in space and, by analogy, in time – that can be controlled with recourse to rules and guidelines. On the other hand, it should not be considered without regard to the user who posed the question and will get the answer, his/her characteristics and expectations. As Lin and Katz (2006) point out, the granularity has much to do with real users: 'better understanding of real-world user needs will lead to more effective question answering systems in the future'.

3.3 Useful answers

As opposed to what is expected to happen in Dialogue Systems, in which interactions are supposed to go across multiple turns or sentences, with a broad variety of communication phenomenon (speech acts), typical interactions within QA systems are limited to isolated questions from the user side, and answers from the system side – exceptions to this are the **interactive QA** systems, where the project Ritel (Rosset, Galibert, Illouz and Max 2006) and the HITIQA system (Small and Strazalkowski 2009) are examples. Therefore, the only information about the user profile must be obtained from the questions he/she poses. Each question contains

information by itself, but information can also be extrapolated from the current interaction context and, if kept, from previous history of user interactions.

Systems can take into consideration clues about the user at three different levels:

- Question clues: several clues in the question can be checked, namely (and surely not limited to):
  - the vocabulary, that greatly differs depending on the user. For instance, on his/her age, academic background and occupation;
  - the specificity and world knowledge. If a user asks: *Who received the Prince of Asturias Award for Technical and Scientific Research for studies on the discovery of the first synthetic vaccination against malaria?*, probably he/she has deep knowledge on this series of annual prizes.
- Clues in the current context: both the questions and the answers involved in user-system interaction can provide some clues about her/his profile. For instance, consider a chain of questions about the 2^{nd} World War. If the question *"What is the real name of Dr. Death?"* appears, probably the useful answer will be *"Aribert Heim"* or *"Josef Mengele"* (or both) and not any other from the set of possible answers.[8]
- Clues in the history of interactions: this can be seen as a general approach to the previous case, where the interactions with the users are kept, and the system is able to detect that he/she is interested in a certain topic.

Although user adaptation is not a new issue in Information Retrieval (IR), specially in Web search engines, where systems try to adapt the presentation of results to the user (Liu, Yu and Meng 2004; Teevan, Dumais and Horvitz 2005), the first steps in this direction in QA only recently were taken. For instance, considering the clues presented in the question and in the current context, DuARTE Digital (Mendes, Prada and Coheur 2009) uses both in order to adapt to the user. This system answers questions about a piece of jewellery, and dynamically tries to assess its interlocutor characteristics, based on the used vocabulary. With a list of words that naive or expert users might employ, it interprets the user's expertise at every question and it chooses the answer, previously marked with the corresponding difficulty level, from a knowledge base. Moreover, it also tries to acknowledge the user's goals at a contextual level, by measuring the proximity of the user's words in a sequence of questions to different sub-topics, in order to understand the orientation of the interaction. By doing so, it distinguishes from focused to stray interactions, and chooses the answer according to its detail and informative level.

Quarteroni and Manandhar (2009) are also pioneers in this topic, as they included on the system YourQA a component dedicated to user modelling. The goal was to filter the documents where answers are searched and to rerank the candidates based on the degree of match with the user's profile. Users must create their own profile when first interacting with the system (it does not dynamically discover its

[8] Notice the multiple candidate answers for this question: `http://en.wikipedia.org/wiki/Dr._Death`

interlocutor characteristics), and their browsing history is taken into account in future interactions. Any question submitted to the system is answered by taking the user's profile into account.

Besides the above mentioned works and all the work being carried out in dialogue systems and IR, to our best knowledge there is no other QA system that performs user adaptation.

4 Answer generation

In this section we focus on QA systems, with regards to their techniques for answer generation.

4.1 Building answers

When it comes to presenting the answer to the user, a myriad of systems choose to directly return an information chunk extracted from the corpora sources, considered as solving the posed question. The simplicity of this approach makes it attractive specially to factoid QA: it provides the sufficient amount of information, what is *enough*. Nevertheless, there are several reasons that make the generation of a natural language answer preferable: first, it *humanizes* the system; second, it permits the usage of vocabulary and/or sentence constructions adapted to the user; finally, it allows the introduction of additional information that the user did not directly request, but might be interested in.

Indeed, it seems that QA could not yet benefit from the results achieved in NLG, despite the much effort devoted to this field throughout the years. However, generation can only add value to the process of answering if the final result is acceptable: for example, it makes no sense to return a syntactically flawed answer, when the direct answer did not have this problem. Benamara and Saint-Dizier (2004) explore this thematic and identify a set of lexicalisation strategies (*i.e.* the attribution of a word or expression to a concept) employed by humans, based on the analysis of a set of questions and their answers. The authors advocate that the natural language form of a response requires adequate and subtle lexicalisations. For instance, in certain circumstances the meaning of a verb can be decisive for its usage over another verb: since *climb* indicates a faster growth than *go up*, the former is preferred for the generation of *The increase of gas prices climb to 20.3% in October 2005*, and the later for *The increase of gas prices go up to 7.2% in September 2005* (Moriceau 2006).

The simplest approach to answer generation relies on templates, which realize the surface content of the answer through a direct mapping of non-linguistic content. Kosseim, Plamondon and Guillemette (2003) refer also to its importance in the context of answer generation to improve the human-computer interaction. However, in their work, authors use a set of manually built formulation templates, composed by lexico-syntactic patterns, uniquely to aid the extraction of the answer,

Moriceau (2005) employs templates and classical NLG techniques to generate natural language answers to questions of category DATE. The author makes the

realization of the answer depend on several variables: *1)* the characteristics of the event in question, namely its recurrence (*i.e.* if it is unique or iterative) and regularity (*i.e.* if it is recurrent and happens at regular intervals); *2)* the type of answer (*i.e.* date or interval); and *3)* a *certainty degree* calculated for the answer. The *certainty degree* is a measure of the system's belief in certain answer being correct for the question, expressed through the usage of adverbs and their intensities (*e.g.* possibly, most possibly, probably, most probably). With this, the user is given an indication of the confidence the system has in the given answer.

Within a QA system on the cooking domain, Xia, Teng and Ren (2009) divide the generation of natural language answers in two phases. A content planing phase starts by deciding *what* to say: the content of the answer is determined according to the classification of the question into one of four categories. Afterwards, a sentence planing phase decides *how* to say, that is, which lexical and syntactic structures will realize the system's output. This is done by using a a set of templates that take into account several components collected from the question, and its answer.

Another example of the usage of templates to generate answers in QA is the work on intensional answering by Benamara (2004a). An intensional answer represents a higher level of abstraction than the extensional one, by providing a description of the instances that satisfy the question by terms of the distinguishing properties they share, which is particularly important when the cardinality of the set of instances in the extensional answer is too high. Benamara describes an approach to create intensional answers from the elements that compose the extensional answer to a question. First, it finds the elements to be generalized, using either the focus of the question (in *"What are the means of transportation to go to Geneva airport?"* the focus *means of transportation* generalizes the elements *buses*, *trolleys* and *trains*), or a property of the focus of the question (in *"What are the hotels at the border of the sea in Monaco?"*, the property *hotel localization*, variable among all possible elements, is used) or a property in a given ontology of the focus of the question (in *"Which hotels in Cannes have a swimming pool?"*, the property *hotel category* serve as generalizer). Then, the set of intensional answers is augmented with the repeated grouping of the generalized elements with respect to the ontology. Finally, the best abstraction level is chosen through a *variable depth intensional calculus*, that makes the decision depend on the ontological proximity of the elements in the intensional answer. Systems that generate intensional answers are often logic-based and supported by domain knowledge sources, like the work by Cimiano, Rudolph and Hartfiel (2010) similar to that of Benamara (2004a).

A different technique is presented by Kacmarcik (2005), who shows an interesting application of QA in the context of a role-playing games. In this work, the human player can choose from a set of generated questions that can be posed to any non-player character (NPC), which is a character controlled by the game engine. The answer is built based on the match between the parse tree of the question and the trees stored in the knowledge base of the NPC responsible for the answer. In addition, the information contained in each NPC's knowledge base is used to individualize its answer.

More sophisticated techniques to answer generation in QA usually rely on sum-

marization, considered as the process of digesting information from a collection of documents, transforming them into shorter summaries that can be read and interpreted quicker. Wu, Radev and Fan (2002) introduced the notion of answer-focused summary as a specific type of documents summaries that aim to provide answers to user's questions. Authors also indicate three criteria that an answer-focused summary has to satisfy: *1)* accuracy: it must have the answer to the question; *2)* economy: it should be concise; and, *3)* support, it should substantiate the answer.

Summarization is often considered in the context of definitional questions, like *"Who is X?"* or *"What is Y?"*, that expect longer and explanatory answers, and for which the traditional model employed in factoid-questions (based on the retrieval of a short information chunk) does not apply. This is the case, for instance, of the approach of Ye, Chua and Lu (2009) that can build summaries of different lengths using both the Wikipedia text and the information contained in the infoboxes. The authors developed an extended document concept lattice model which is able to select and group sentences representative of local topics. These clusters of sentences are then used as answers. Another example comes from the DefScriber (Blair-goldensohn, Mckeown and Schlaikjer 2003), a system that uses a hybrid approach to answer definitional, biographical and topic-focused questions, borrowing techniques from summarization and generation. The authors introduce the concept of *definitional predicates* as the information types that are often useful in definitions: for instance, the *Genus* predicate captures the category to which the term belongs. The DefScriber starts by applying a goal-driven strategy to capture those predicates from the text, based on a machine-learning classifier and syntax pattern matching; afterwards a data-driven strategy infers and organizes the themes that may arise from the data.

4.2 "No Answer" Questions

Correct answers are greatly rewarded in evaluation campaigns and, although it is often considered that "*no answer is better than a wrong answer*", typically there is no distinction in scoring between returning a wrong answer to a question, or returning nothing. Indeed, albeit a wrong answer totally contradicts the goal of QA (note that to be able to identify a wrong answer is still an open research problem), retrieving "no answer" is not better: indeed, it does not bring any valuable information to the user, and can lead him/her to misinterpretations. If one is to build a cooperative system, what to present when no answer was found is also an issue.

On the one hand, questions with "no answer" can arise on the system side, when it can not find the answer within the available corpora. The Noflail[9] search engine deals with this situation for queries with five or less keywords. The system relaxes the user query by identifying subsets of keywords that produce results. These subsets are listed together with the cardinalities of their result sets, and are available for the user to access, in case he/she is interested in these results.

[9] Available at `http://noflail.com/`. Last accessed on September 2010.

On the other hand, questions with "no answer" can be originated on the user side. Many questions simply have no answer, specifically those with a false presupposition. Consider the question *"Who is the King of France?"*. Knowing that France is a republic, answering *"no one"* or *"NIL"* can drive the user to think the system was unable to find the correct answer. Here an explanation is due showing why no answer exists to the question. The WEBCOOP system, a restricted domain logic-based cooperative QA system, does exactly that. It integrates knowledge representation and advanced reasoning to detect false presuppositions and misunderstandings in questions, in order to deliver non-misleading answers (Benamara 2004b). Misconceptions in the user questions are detected if they fail to satisfy certain constraints and if the entities and/or relations they mention are not contemplated in the knowledge base. To our knowledge, and due to the difficulty of the task without appropriate knowledge, there is no open domain QA system that can deal with this problematic.

4.3 Answer support

Till this point, the focus of this survey was on how to return the answer to a question. Nevertheless, answers can be augmented with content to improve the user's experience with the system, in three different manners:

1. by providing a justification why a certain answer was given;
2. by allowing the user to validate the answer he/she was given;
3. by offering extra information that can be of interest to the user.

Lin, Quan, Sinha, Bakshi, Huynh, Katz and Karger (2003) focused on the role of context to make a good answer, based on the belief that the most natural response presentation style for QA is *focus-plus-context* (Leung and Apperley 1994). Four scenarios were compared in which context can be presented to the user, by varying the length of the text surrounding the correct answer to questions: *exact answer*, *answer-in-sentence*, *answer-in-paragraph*, and *answer-in-document*. The study showed that the preference was for *answer-in-paragraph*, as the users consider the paragraph as a 'good size of information chunk', being the *exact answer* a too short chunk and the entire document too long. Moreover, the authors indicate that the *answer-in-sentence* does not seem to bring any advantage over the *exact answer*, and pose some issues related with information taken out of context, specially due to anaphoric expressions. Kaisser (2008) introduced a supporting paragraph in the demo of his QA system: QuALiM. Besides the answer itself, a paragraph relating the answer to the question and a link to the Wikipedia article are presented, in case the user would like to read further details on that topic.

Again, summarization can be considered in the context of the QA task for the purpose of answer support. In the clinical domain, Demner-fushman and Lin (2006) present a hybrid approach that borrows techniques from IR and summarization to generate answers to users: the answer extraction phase picks short answers from

MEDLINE citations retrieved by the PubMed [10] search engine; a semantic clustering phase groups the retrieved MEDLINE citations; finally, the extractive summarization phase creates summaries from each MEDLINE citation which will work as evidence to each given answer.

5 Conclusions and future directions

To address the user's question, QA systems usually follow a strategy based on the direct extraction of the answer, supported by the redundancy of the information sources. However, as we have seen throughout this paper, it has some flaws. For instance, instead of working jointly for the selection of the final answer, candidate answers are typically considered separately, even if they are semantically related. Moreover, some important procedures to ascertain that the user understands the answer are often neglected: for example, it is not taken into consideration whether the answer is ambiguous, or even if it contains inadequate vocabulary.

Inserted in this problematic, this survey covered current efforts in the process of answering within QA systems. We started from the common answer extraction and selection approach, and investigated current works on three different research lines. First, we addressed the relations existing between candidate answers for the purpose of accurately computing the final answer. In this context, we presented a typology of four general relations (Moriceau 2005) and described several techniques for their detection. We showed the work of Dalmas and Webber (2007) as an example of a system that considers candidate answers as allies, instead of competitors, and relate them by equivalence and inclusion. Afterwards, we recovered the notion of cooperative answer introduced by Gaasterlan, Godfrey and Minker (1992), as being a correct, useful and non-misleading answer. Besides surveying what makes an answer correct, we discussed a different model for QA systems that give the user an individualized (*i.e.* useful) answer, with an unique interpretation (*i.e.* non-misleading). As an example, we presented YourQA (Quarteroni and Manandhar 2009), a system that adapts its answer to the current user's profile. Finally, we considered the generation of answers as a complementary approach to using an information chunk extracted directly from the corpus. In this context, we described from template- to summarization-based strategies. Moreover, we presented systems that build the answer with extra content that can be of the user's interest. We gave the example of the work by Moriceau (2005), that uses generation to provide the user a hint about the system's confidence on the answer.

Despite some existing works addressing this problematic, the answering phase in QA systems has instigated relatively little interest from the scientific community, when compared to its importance. Thus, there is still much to be done when it comes to returning to the user the answer to his/her question. It is our opinion that research in QA should evolve in the following directions:

- user adaptation: QA systems are built for humans. However, QA systems did

[10] Available at `http://www.ncbi.nlm.nih.gov/pubmed`. Last accessed on September 2010.

not achieve the impact to users as IR did, with search engines like Google or Yahoo already making part of people's lives. We believe that one way to make QA an appealing alternative to users has to do with personalisation. In this sense, the path is still long and with a few open research questions: Which characteristics of the users are to be modelled? How to dynamically infer a user model from the posed questions? How to effectively present the results according to the user?

- answer credibility and justification: the web is a resource increasingly used in QA. Systems have to overcome the existence of fake or flawed information, which is typically done with recourse to redundancy. Nevertheless, the incorporation of more robust mechanisms to guarantee the credibility of the answer are required, as well as the justification of how and why that answer was given.
- systems evaluation: cooperativeness theoretically arises as a desirable model for QA. In this context, the definition of the evaluation criteria of QA systems is necessary, allowing both to measure the performance of a system, as well as further comparisons between systems.

References

Balahur, A. and Montoyo, A. (2008) A feature dependent method for opinion mining and classification. In *Int. Conf. on Natural Language Processing and Knowledge Engineering. NLP-KE '08.*IEEE.

Benamara, F. (2004a) Generating Intensional Answers in Intelligent Question Answering Systems. In A. Belz et al (eds.), *Int. Conf. on Natural Language Generation (INLG)*, pp. 11–20. Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag.

Benamara, F. (2004b) Cooperative question answering in restricted domains: the WEB-COOP experiment. In Diego Mollá and José Luis Vicedo (eds.), *ACL 2004: Question Answering in Restricted Domains*, pp. 31–38. ACL.

Benamara, F. and Saint-Dizier, P. (2004) Lexicalisation Strategies in Cooperative Question Answering Systems. In B. Gambäck and K. Jokinen (eds.), *COLING '04: Proc. of the 20^{th} Int. Conf. on Computational Linguistics*, pp. 1179–1185. ACL.

Blair-goldensohn, S. and Mckeown, K. R. and Schlaikjer, A. H. (2003) A Hybrid Approach for QA Track Definitional Questions. In *Proc. of the 12^{th} Annual Text Retrieval Conf.*, pp. 185–192.

Breck, E. J. and Burger, J. D. and Ferro, L. and Hirschman, L. and House, D. and Light, M. and Mani, I. (2000) How to Evaluate Your Question Answering System Every Day and Still Get Real Work Done. In *Proc. of the 2^{nd} Int. Conf. on Language Resources and Evaluation (LREC-2000)*, pp. 1495–1500.

Chu-Carroll, J. and Czuba, K. and Prager, J. and Ittycheriah, A. (2003) In question answering, two heads are better than one. In *NAACL '03: Proc. of the 2003 Conf. of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pp. 24–31. ACL.

Cimiano, P. and Rudolph, S. and Hartfiel, H. (2010) Computing intensional answers to questions – An inductive logic programming approach. *Data and Knowledge Engineering* **69**: 261–278.

Cohen, W. W. and Ravikumar, P. and Fienberg, S. E. (2003) A Comparison of String Distance Metrics for Name-Matching Tasks. In Subbarao Kambhampati and Craig A. Knoblock (eds.), *Proc. of IJCAI-03 Workshop on Information Integration*, pp. 73–78.

Dalmas, T. and Webber, B. (2007) Answer comparison in automated question answering. *Journal of Applied Logic* **5**: 104–120.

Danescu-Niculescu-Mizil, C. and Lee, L. and Ducott, R. (2009) Without a 'doubt'?: unsupervised discovery of downward-entailing operators. In *NAACL '09: Proc. of Human Language Technologies: the 2009 Annual Conf. of the North American Chapter of the Association for Computational Linguistics*, pp. 137–145. ACL.

De Boni, M. and Manandhar, S. (2005) Implementing clarification dialogues in open domain question answering. *Journal of Natural Language Engineering* **11**: 343–362.

Demner-fushman, D. and Lin, J. (2006) Answer Extraction, Semantic Clustering, and Extractive Summarization for Clinical Question Answering In *ACL-44: Proc. of the 21st Int. Conf. on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pp. 841–848. ACL.

Dom, B. and Paranjpe, D. (2008) A Bayesian Technique for Estimating the Credibility of question Answerers. In *Proc. of the SIAM Int. Conf. on Data Mining, SDM 2008*, pp. 399–409. SIAM.

Downey, D. and Etzioni, O. and Soderl, S. (2005) A probabilistic model of redundancy in information extraction In *IJCAI'05: Proc. of the 19th Int. Joint Conf. on Artificial Inteligence*, pp. 1034–1041. Morgan Kaufmann Publishers Inc.

Fellbaum, C. (1998) *WordNet: An Electronic Lexical Database*. MIT Press.

France, F.D. and Yvon, F. and Collin, O. (2003) Learning Paraphrases to Improve a Question-Answering System. In *Proc. of the 10th Conf. of EACL Workshop Natural Language Processing for Question-Answering*, pp. 35–41. ACL.

Gaasterland, T. and Godfrey, P. and Minker, J. (1992) An Overview of Cooperative Answering. *Journal of Intelligent Information Systems* **1**: 123–157.

Grice, H. (1975) *Logic and Conversation*. Syntax and Semantics: Vol. 3: Speech Acts. Academic Press.

Harabagiu, S. and Hickl, A. and Lacatusu, F. (2006) Negation, contrast and contradiction in text processing. In *AAAI'06: Proc. of the 21st national Conf. on Artificial Intelligence*, pp. 755–762. AAAI Press.

Hearst, M. A. (1992) Automatic acquisition of hyponyms from large text corpora. In *Proc. of the 14th Conf. on Computational Linguistics*, pp. 539–545. ACL.

Huang, Z. and Thint, M. and Qin, Z. (2008) Question classification using head words and their hypernyms. In *EMNLP '08: Proc. of the Conf. on Empirical Methods in Natural Language Processing*, pp. 927–936. ACL.

Hung, J. C. and Wang, C. and Yang, C. and Chiu, M. and Yee, G. (2005) Applying Word Sense Disambiguation to Question Answering System for e-Learning. In *AINA '05: Proc. of the 19th Int. Conf. on Advanced Information Networking and Applications*, pp. 157–162. IEEE Computer Society.

Iftene, A. and Balahur, A. (2008) Answer Validation on English and Romanian Languages. In C. Peters et al. (eds.), *Evaluating Systems for Multilingual and Multimodal Information Access*, pp. 448-451. Lecture Notes in Computer Science. Berlin: Springer-Verlag.

Isozaki, H. (2004) NTT's question answering system for NTCIR QAC2 In *Proc. of the NTCIR Workshop 4 Meeting (NTCIR-4)*, pp. 326–332.

Kacmarcik, G. (2005) Question answering in role-playing games. In *Workshop on Question Answering in Restricted Domains. 20th National Conf. on Artificial Intelligence (AAAI-05)*, pp. 51–55.

Kaisser, M. (2008) The QuALiM Question Answering Demo: Supplementing Answers with Paragraphs drawn from Wikipedia. In *Proc. of ACL-08: HLT Demo Session (Companion Volume)*, pp. 32–35. ACL.

Khalid, M.A. and Jijkoun, V.B. and de Rijke, M. (2008) The Impact of Named Entity Normalization on Information Retrieval for Question Answering. In Craig Macdonald et al. (eds.), *30th European Conf. on Information Retrieval (ECIR 2008)*, pp. 705–710. Lecture Notes in Computer Science. Berlin: Springer/Heidelberg.

Ko, J. and Si. L. and Nyberg, E. (2007) A Probabilistic Framework for Answer Selection in Question Answering. In Candance L. Sidner et al. (eds.), *Proc. of Human Language Technology Conf. of the NAACL*, pp. 524–531. ACL.

Kosseim, L. and Plamondon, L. and Guillemette, L. (2003) Answer Formulation for Question-Answering. In Y. Xiang and B. Chaib-draa (eds.), *Proc. 16th Conf. of the Canadian Society for Computational Studies of Intelligence, AI 2003*, pp. 24–34. Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag.

Kozareva, Z. and Vazquez, S. and Montoyo, A. (2006) Adaptation of a Machine-learning Textual Entailment System to a Multilingual Answer Validation Exercise. In *Working notes of CLEF - ECDL 2006, AVE Workshop.*

Leidner, J. L. and Sinclair, G. and Webber, B. (2003) Grounding spatial named entities for information extraction and question answering. In *Proc. of Human Language Technology Conf. of the NAACL 2003, Workshop on Analysis of Geographic References*, pp. 31–38. ACL.

Leung, Y. K. and Apperley, M. D. (1994) A review and taxonomy of distortion-oriented presentation techniques. *ACM Trans. Comput.-Hum. Interact.* **1**: 126–160.

Lin, D. and Zhao, S. and Qin, L. and Zhou, M. (2003) Identifying synonyms among distributionally similar words. In *IJCAI'03: Proc. of the 18th Int. joint Conf. on Artificial Intelligence*, pp. 1492–1493. Morgan Kaufmann Publishers Inc.

Lin, J. and Quan, D. and Sinha, V. and Bakshi, K. and Huynh, D. and Katz, B. and Karger, D.R. (2003) What Makes a Good Answer? The Role of Context in Question Answering. In M. Rauterberg et al. (eds.) *Human-Computer Interaction (INTERACT 2003)* , pp. 25–32.

Lin, J. and Katz, B. (2006) Building a reusable test collection for question answering. *Journal of the American Society for Information Science and Technology* **57**: 851–861.

Lin, J. (2007) An exploration of the principles underlying redundancy-based factoid question answering. *ACM Transactions on Information Systems* **25**: 1–55.

Liu, F. and Yu, C. and Meng, W. (2004) Personalized Web Search For Improving Retrieval Effectiveness. *IEEE Transactions on Knowledge and Data Engineering* **16**: 28–40.

Magnini, B. and Negri, M. and Prevete, R. and Tanev, H. (2002) Is it the right answer?: exploiting web redundancy for Answer Validation. In *ACL '02: Proc. of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 425–432. ACL.

Magnini, B. and Romagnoli, S. and Vallin, A. and Herrera, J. and Peñas, A. and Peinado, V. and Verdejo, F. and De Rijke, M. and Vallin, R. (2003) The Multiple Language Question Answering Track at CLEF 2003. In *CLEF 2003. CLEF 2003 Workshop.*Berlin:Springer-Verlag.

Magnini, B. and Vallin, R. and Ayache, C. and Erbach, G. and Peñas, A. and De Rijke, M. and Rocha, P. and Kiril, S. and Sutcliffe, R. (2005) Overview of the CLEF 2004 Multilingual Question Answering Track. In *Results of the CLEF 2004 Cross-Language System Evaluation Campaign*, pp. 371–391. Berlin:Springer-Verlag.

Mendes, A.C. and Prada, R. and Coheur, L. (2009) Adapting a Virtual Agent to Users' Vocabulary and Needs. In Rutkay et al. (eds.), *IVA '09: Proc. of the 9th Int. Conf. on Intelligent Virtual Agents*, pp. 529–530. Lecture Notes in Artificial Intelligence. Berlin: Springer-Verlag.

Metzger, M. J. (2007) Making sense of credibility on the Web: Models for evaluating online information and recommendations for future research. *Journal of the American Society for Information Science and Technology* **58**: 2078–2091.

Mohammad, Saif and Dorr, Bonnie and Hirst, G. (2008) Computing word-pair antonymy. In *EMNLP '08: Proc. of the Conf. on Empirical Methods in Natural Language Processing*, pp. 982–991. ACL.

Moldovan, Dan and Paşca, Marius and Harabagiu, Sanda and Surdeanu, Mihai (2003) Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.* **21**: 133–154.

Moriceau, V. (2005) Answer Generation with Temporal Data Integration. In Graham Wilcock et al. (eds.), *Proceeding of the 10th European Workshop on Natural Language Generation (ENLG-05)*, pp. 197–202. University of Aberdeen.

Moriceau, V. (2006) Numerical Data Integration for Cooperative Question-Answering. In *Proc. of the Workshop KRAQ'06 on Knowledge and Reasoning for Language Processing*, pp. 43–50. ACL.

Moriceau, V. (2007) Intégration de données dans un système question-réponse sur le Web. PhD Thesis, Université Paul Sabatier.

Nii, Y. and Kawata and K. Yoshida, T. and Sakai, H and Masuyama, S. (2004) Question Answering System QUARK. In *Proc. of the NTCIR-4*.

Pantel, P. and Ravichandran, D. (2004) Automatically labeling semantic classes. In Susan Dumais, Daniel Marcu and Salim Roukos (eds.) *Proc. of the North American Association for Computational Linguistics and the Human Language Technologies Conferences (NAACL/HLT2004)*, pp. 321–328. ACL.

Quarteroni, S. and Manandhar, S. (2007) User Modelling for Personalized Question Answering. In R. Basili and M. Pazienza (eds.), *AI*IA '07: Proc. 10th Congress of the Italian Association for Artificial Intelligence on AI*IA 2007*, pp. 386–397. Lecture Notes in Computer Science. Berlin: Springer-Verlag.

Quarteroni, S. and Manandhar, S. (2009) Designing an interactive open-domain question answering system. *Journal of Natural Language Engineering* **25**: 73–95.

Quintano, L. and Rodrigues, I. P. (2008) Question/Answering Clarification Dialogues. In A. F. Gelbukh and E. F. Morales (eds.), *MICAI 2008: Advances in Artificial Intelligence, 7th Mexican Int. Conf. on Artificial Intelligence*, pp. 155–164. Lecture Notes in Computer Science. Berlin: Springer-Verlag.

Ritter, A. and Soderland, S. and Etzioni, O. (2009) What is this, anyway: Automatic hypernym discovery. In *Proc. of AAAI-09 Spring Symposium on Learning by Reading and Learning to Read*, pp. 88–93. AAAI.

Rosset, S. and Galibert, O. and Illouz, G. and Max, A. (2006) Integrating Spoken Dialog and Question Answering: The Ritel Project. In *Proc. of Interspeech 2006—ICSLP: 9th Int. Conf. on Spoken Language Processing*, pp. 1914–1917.

Sacaleanu, B. and Neumann, G. (2006) Cross-Cutting Aspects of Cross-Language Question Answering Systems In *MLQA '06: Proc. of the Workshop on Multilingual Question Answering*, pp. 15–22. ACL.

Shanon, B. (1979) Where questions. In *Proc. of the 17th annual meeting on Association for Computational Linguistics*, pp. 73–75. ACL.

Small, S. and Strazalkowski, T. (2009) HITIQA: High-quality intelligence through interactive question answering. *Natural Language Engineering* **15**: 31-54.

Stoyanov, V. and Cardie, C. and Wiebe, J. (2005) Multi-perspective question answering using the OpQA corpus. In *HLT '05: Proc. of the Conf. on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 923–930. ACL.

Takahashi, T. and Nawata, K. and Inui, K. and Matsumoto, Y. (2003) Effects of Structural Matching and Paraphrasing in Question Answering (Special Issue on Text Processing for Information Access). *IEICE transactions on information and systems* **86**: 1677–1685.

Teevan, J. and Dumais, S.T. and Horvitz, E. (2005) Personalizing search via automated analysis of interests and activities. In *SIGIR '05: Proc. 28th annual Int. ACM SIGIR Conf. on Research and development in information retrieval*, pp. 449–456. ACM.

Téllez-Valero, A., Juárez-González, A., Montes-y-Gómez, M., & Villaseñor-Pineda, L. (2008) INAOE at QA@CLEF 2008: Evaluating Answer Validation in Spanish Question Answering. In *Working notes of CLEF - ECDL 2008, AVE Workshop*

Turney, P.D. (2008) A uniform approach to analogies, synonyms, antonyms, and associations In *COLING '08: Proc. of the 22nd Int. Conf. on Computational Linguistics*, pp. 905–912. ACL.

Voorhees, E. M. and Tice, D. M. (1999) The TREC-8 Question Answering Track Evaluation. In *In Text Retrieval Conf. TREC-8*, pp. 83–105.

Webber, B. and Gardent, C. and Bos, J. (2002) Position statement: Inference in Question Answering. In *In Proc. of the LREC Workshop on Question Answering: Strategy and Resources*, pp. 19–25.

Wilson, T. and Wiebe, J. and Hoffmann, P. (2009) Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics* **35**: 399–433.

Wu, H., Radev, D. R., and Fan, W. (2002) Towards answer focused summarization. In *Proc. of the 1st Int. Conf. on Information Technology and Applications.*

Wu, M. and Marian, A. (2007) Corroborating Answers from Multiple Web Sources. In *10th Int. Workshop on the Web and Databases (WebDB).*

Xia, L. and Teng, Z. and Ren, F. (2009) Answer generation for Chinese cuisine QA system. In *Int. Conf. on Natural Language Processing and Knowledge Engineering (NLP-KE)*, pp. 1–6. IEEE.

Ye, S. and Chua, T. and Lu, J. (2009) Summarizing definition from Wikipedia. In *ACL-IJCNLP '09: Proc. of the Joint Conf. of the 47th Annual Meeting of the ACL and the 4th Int. Joint Conf. on Natural Language Processing of the AFNLP: Volume 1*, pp. 199–207. ACL.

Yu, H. and Hatzivassiloglou, V. (2003) Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences In *Proc. of the 2003 Conf. on Empirical Methods in Natural Language Processing*, pp. 129–136. ACL.