Correcting Object-Related Misconceptions: How Should The System Respond?[1]

Kathleen F. McCoy
Department of Computer & Information Science
University of Pennsylvania
Philadelphia, PA 19104

Abstract

This paper describes a computational method for correcting users' misconceptions concerning the objects modelled by a computer system. The method involves classifying object-related misconceptions according to the knowledge-base feature involved in the incorrect information. For each resulting class sub-types are identified, according to the structure of the knowledge base, which indicate what information may be supporting the misconception and therefore what information to include in the response. Such a characterization, along with a model of what the user knows, enables the system to reason in a domain-independent way about how best to correct the user.

1. Introduction

A major area of AI research has been the development of "expert systems" – systems which are able to answer user's questions concerning a particular domain. Studies identifying desirable interactive capabilities for such systems [Pollack et al. 82] have found that it is not sufficient simply to allow the user to ask a question and have the system answer it. Users often want to question the system's reasoning,to make sure certain constraints have been taken into consideration, and so on. Thus we must strive to provide expert systems with the ability to interact with the user in the kind of cooperative dialogues that we see between two human conversational partners.

Allowing such interactions between the system and a user raises difficulties for a Natural-Language system. Since the user is interacting with a system as s/he would with a human expert, s/he will most likely expect the system to *behave* as a human expert. Among other things, the user will expect the system to be adhering to the cooperative principles of conversation [Grice 75, Joshi 82]. If these principles are not followed by the system, the user is likely to become confused.

In this paper I focus on one aspect of the cooperative behavior found between two conversational partners: responding to recognized differences in the beliefs of the two participants. Often when two people interact, one reveals a belief or assumption that is incompatible with the beliefs held by the other. Failure to correct this disparity may not only implicitly confirm the disparate belief, but may even make it impossible to complete the ongoing task. Imagine the following exchange:

U. Give me the HULL_NO of all Destroyers whose MAST_HEIGHT is above 190.
E. All Destroyers that I know about have a MAST_HEIGHT between 85 and 90. Were you thinking of the Aircraft-Carriers?

In this example, the user (U) has apparently confused a Destroyer with an Aircraft-Carrier. This confusion has caused her to attribute a property value to Destroyers that they do not have. In this case a correct answer by the expert (E) of "none" is likely to confuse U. In order to continue the conversation with a minimal amount of confusion, the user's incorrect belief must first be addressed.

My primary interest is in what an expert system, aspiring to human expert performance, should include in such responses. In particular, I am concerned with system responses to recognized disparate beliefs/assumptions *about objects*. In the past this problem has been left to the tutoring or CAI systems [Stevens et al. 79, Stevens & Collins 80, Brown & Burton 78, Sleeman 82], which attempt to correct student's misconceptions concerning a particular domain. For the most part, their approach has been to list *a priori* all misconceptions in a given domain. The futility of this approach is emphasized in [Sleeman 82]. In contrast,the approach taken here is to classify, in a domain independent way, object-related disparities according to the Knowledge Base (KB) feature involved. A number of response strategies are associated with each resulting class. Deciding which strategy to use for a given misconception will be determined by analyzing a user model and the discourse situation.

2. What Goes Into a Correction?

In this work I am making the following assumptions:

- For the purposes of the initial correction attempt, the system is assumed to have *complete* and *correct* knowledge of the domain. That is, the system will initially perceive a disparity as a misconception on the part of the user. It will thus attempt to bring the user's beliefs into line with its own.

- The system's KB includes the following *features*: an object taxonomy, knowledge of object attributes and their possible values, and information about possible relationships between objects.

- The user's KB contains similar features. However, much of the information (content) in the system's KB may be missing from the user's KB (e.g., the user's KB may be sparser or coarser than the system's KB, or various attributes of concepts may be missing from the user's KB). In addition, some information in the user's KB may be wrong. In this work, to say that the user's KB is *wrong* means that it is *inconsistent with the system's KB* (e.g., things may be classified differently, properties attributed differently, and so on).

[1] This work is partially supported by the NSF grant #MCS81-07290.

- While the system may not know exactly what is contained in the user's KB, information about the user can be derived from two sources. First, the system can have a model of a canonical user. (Of course this model may turn out to differ from any given user's model.) Secondly, it can derive knowledge about what the user knows from the ongoing discourse. This later type of knowledge constitutes what the system discerns to be the mutual beliefs of the system and user as defined in [Joshi 82]. These two sources of information together constitute the system's *model* of the user's KB. This model itself may be incomplete and/or incorrect with respect to the system's KB.

- A user's utterance reflects either the state of his/her KB, or some reasoning s/he has just done to fill in some missing part of that KB, or both.

Given these assumptions, we can consider what should be included in a response to an object-related disparity. If a person exhibits what his/her conversational partner perceives as a misconception, the very least one would expect from that partner is to deny the false information[2] - for example -

U. I thought a whale was a fish.
S. It's not.

Transcripts of "naturally occurring" expert systems show that experts often include more information in their response than a simple denial. The expert may provide an alternative true statement (e.g., "Whales are mammals"). S/he may offer justification and/or support for the correction (e.g., "Whales are mammals because they breathe through lungs and feed their young with milk."). S/he may also refute the faulty reasoning s/he thought the user had done to arrive at the misconception (e.g., "Having fins and living in the water is not enough to make a whale a fish."). This behavior can be characterized as confirming the correct information which may have led the user to the wrong conclusion, but indicating why the false conclusion does not follow by bringing in additional, overriding information.[3]

The problem for a computer system is to decide what kind of information may be supporting a given misconception. What things may be relevant? What faulty reasoning may have been done?

I characterize object-related misconceptions in terms of the *KB feature* involved. Misclassifying an object, "I thought a whale was a fish", involves the superordinate KB feature. Giving an object a property it does not have, "What is the interest rate on this stock?", involves the attribute KB feature. This characterization is helpful in determining, in terms of the structure of a KB, what information may be supporting a particular misconception. Thus, it is helpful in determining what to include in the response.

In the following sections I will discuss the two classes of object misconceptions just mentioned: superordinate misconceptions and attribute misconceptions. Examples of these classes along with correction strategies will be given. In addition, indications of how a system might choose a particular strategy will be investigated.

3. Superordinate Misconceptions

Since the information that human experts include in their response to a superordinate misconception seems to hinge on the expert's perception of why the misconception occurred or what information may have been supporting the misconception, I have sub-categorized superordinate misconceptions according to the kind of support they have. For each type (sub-category) of superordinate misconception, I have identified information that would be relevant to the correction.

In this analysis of superordinate misconceptions, I am assuming that the user's knowledge about the superordinate concept is correct. The user therefore arrives at the misconception because of his/her incomplete understanding of the object. I am also, for the moment, ignoring misconceptions that occur because two objects have similar names.

Given these restrictions, I found three major correction strategies used by human experts. These correspond to three reasons why a user might misclassify an object:

TYPE ONE - Object Shares Many Properties with Posited Superordinate - This may cause the user wrongly to conclude that these shared attributes are inherited from the superordinate. This type of misconception is illustrated by an example involving a student and a teacher:[4]

U. I thought a whale was a fish.
E. No, it's a mammal. Although it has fins and lives in the water, it's a mammal since it is warm blooded and feeds its young with milk.

Notice the expert not only specifies the correct superordinate, but also gives additional information to justify the correction. She does this by acknowledging that there are some properties that whales share with fish which may lead the student to conclude that a whale is a fish. At the same time she indicates that these properties are not sufficient for inclusion in the class of fish. The whale, in fact, has other properties which define it to be a mammal.

Thus, the strategy the expert uses when s/he perceives the misconception to be of TYPE ONE may be characterized as: (1) Deny the posited superordinate and indicate the correct one, (2) State attributes (properties) that the object has in common with the posited superordinate, (3) State defining attributes of the real superordinate, thus giving evidence/justification for the correct classification. The system may follow this strategy when the user model indicates that the user thinks the posited superordinate and the object are similar because they share many common properties (not held by the real superordinate).

TYPE TWO - Object Shares Properties with Another Object which is a Member of Posited Superordinate - In this case the

[2] Throughout this work I am assuming that the misconception is important to the task at hand and should therefore be corrected. The responses I am interested in generating are the "full blown" responses. If a misconception is detected which is not important to the task at hand, it is conceivable that either the misconception be ignored or a "trimmed" version of one of these responses be given.

[3] The strategy exhibited by the human experts is very similar to the "grain of truth" correction found in tutoring situations as identified in [Woolf & McDonald 83]. This strategy first identifies the grain of truth in a student's answer and then goes on to give the correct answer.

[4] Although the analysis given here was derived through studying actual human interactions, the examples given are simply illustrative and have not been extracted from a real interaction.

misclassified object and the "other object" are similar because they have some *other* common superordinate. The properties that they share are not those inherited from the posited superordinate; but those inherited from this other common superordinate. Figure 3-1 shows a representation of this situation. OBJECT and OTHER-OBJECT have many common properties because they share a common superordinate (COMMON-SUPERORDINATE). Hence, if the user knows that OTHER-OBJECT is a member of the POSITED SUPERORDINATE, s/he may wrongly conclude that OBJECT is also a member of POSITED SUPERORDINATE.

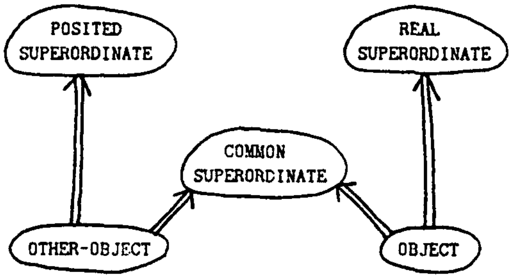

Figure 3-1: TYPE TWO Superordinate Misconception

For example, imagine the following exchange taking place in a junior high school biology class (here U is a student, E a teacher):

U. I thought a tomato was a vegetable.
E. No it's a fruit. You may think it's a vegetable since you grow tomatoes in your vegetable garden along with the lettuce and green beans. However, it's a fruit because it's really the ripened ovary of a seed plant.

Here it is important for the student to understand about plants. Thus, the teacher denies the posited superordinate, vegetable, and gives the correct one, fruit. She backs this up by refuting evidence that the student may be using to support the misconception. In this case, the student may wrongly believe that tomatoes are vegetables because they are like some other objects which are vegetables, lettuce and green beans, in that all three share the common superordinate: plants grown in vegetable garden. The teacher acknowledges this similarity but refutes the conclusion that tomatoes are vegetables by giving the property of tomatoes which define them to be fruits.

The correction strategy used in this case was: (1) Deny the classification posited by the user and indicate the correct classification. (2) Cite the other members of the posited superordinate that the user may be either confusing with the object being discussed or making a bad analogy from. (3) Give the features which distinguish the correct and posited superordinates thus justifying the classification. A system may follow this strategy if a structure like that in figure 3-1 is found in the user model.

TYPE THREE - Wrong Information - The user either has been told wrong information and has not done any reasoning to justify it, or has misclassified the object in response to some complex reasoning process that the system can't duplicate. In this kind of situation, the system, just like a human expert, can only correct the wrong information, give the corresponding true information, and possibly give some defining features distinguishing the posited and actual superordinates. If this correction does not satisfy the user, it is up to him/her to continue the interaction until the underlying misconception is cleared up (see [Jefferson 72]).

The information included in this kind of response is similar to that which McKeown's TEXT system, which answers questions about database structure [McKeown 82], would include if the user had asked about the difference between two entities. In her case, the information included would depend on how similar the two objects were according to the system KB, not on a model of what the user knows or why the user might be asking the question.[5]

U. Is a debenture a secured bond?
S. No it's an unsecured bond - it has nothing backing it should the issuing company default.
AND
U. Is the whiskey a missile?
S. No, it's a submarine which is an underwater vehicle (not a destructive device).

The strategy followed in these cases can be characterized as: (1) Deny posited superordinate and give correct one, (2) Give additional information as needed. This extra information may include defining features of the correct superordinate or information about the highest superordinate that distinguishes the object from the posited superordinate. This strategy may be followed by the system when there is insufficient evidence in the user model for concluding that either a TYPE ONE or a TYPE TWO misconception has occurred.

4. Attribute Misconceptions

A second class of misconception occurs when a person wrongly attributes a property to an object. There are at least three reasons why this kind of misconception may occur.

TYPE ONE - Wrong Object - The user is either *confusing* the object being discussed with another object that has the specified property, or s/he is making a *bad analogy* using a similar object. In either case the second object should be included in the correction so the problem does not continue.

In the following example the expert assumes the user is confusing the object with a similar object.

U. I have my money in a money market certificate so I can get to it right away.
E. But you can't! Your money is tied up in a certificate - do you mean a money market fund?

The strategy followed in this situation can be characterized as: (1) Deny the wrong information, (2) Give the corresponding correct information, (3) Mention the object of confusion or possible analogical reasoning. This strategy can be followed by a system when there is another object which is "close in concept" to the object being discussed and which has the property involved in the misconception. Of course, the perception of how "close in concept" two objects are changes with context. This may be because some attributes are highlighted in some contexts and hidden in others. For this reason it is anticipated that a closeness

[5] McKeown does indicate that this kind of information would improve her responses. The major thrust of her work was on text structure; the use of a user model could be easily integrated into her framework.

measure such as that described in [Tversky 77], which takes into account the salience of various attributes, will be useful.

TYPE TWO - Wrong Attribute - The user has confused the attribute being discussed with another attribute. In this case the correct attribute should be included in the response along with additional information concerning the confused attributes (e.g., their similarities and differences). In the following example the similarity of the two attributes, in this case a common function, is mentioned in the response:

U. Where are the gills on the whale?
S. Whales don't have gills, they breathe through lungs.

The strategy followed was: (1) Deny attribute given, (2) Give correct attribute, (3) Bring in similarities/differences of the attributes which may have led to the confusion. A system may follow this strategy when a similar attribute can be found.

There may be some difficulty in distinguishing between a TYPE ONE and a TYPE TWO attribute misconception. In some situations the user model alone will not be enough to distinguish the two cases. The use of past immediate focus (see [Sidner 83]) looks to be promising in this case. Heuristics are currently being worked out for determining the most likely misconception type based on what kinds of things (e.g., sets of attributes or objects) have been focused on in the recent past.

TYPE THREE - The user was simply given bad information or has done some complicated reasoning which can not be duplicated by the system. Just as in the TYPE THREE superordinate misconception, the system can only respond in a limited way.

U. I am not working now and my husband has opened a spousal IRA for us. I understand that if I start working again, and want to contribute to my own IRA, that we will have to pay a penalty on anything that had been in our spousal account.
E. No - There is no penalty. You can split that spousal one any way you wish. You can have 2000 in each.

Here the strategy is: (1) Deny attribute given, (2) Give correct attribute. This strategy can be followed by the system when there is not enough evidence in the user model to conclude that either a TYPE ONE or a TYPE TWO attribute misconception has occurred.

5. Conclusions

In this paper I have argued that any Natural-Language system that allows the user to engage in extended dialogues must be prepared to handle misconceptions. Through studying various transcripts of how people correct misconceptions, I found that they not only correct the wrong information, but often include additional information to convince the user of the correction and/or refute the reasoning that may have led to the misconception. This paper describes a framework for allowing a computer system to mimic this behavior.

The approach taken here is first to classify object-related misconceptions according to the KB feature involved. For each resulting class, sub-types are identified in terms of the structure of a KB rather than its content. The sub-types characterize the kind of information that may support the misconception. A correction strategy is associated with each sub-type that indicates what kind of information to include in the response. Finally, algorithms are being developed for identifying the type of a particular misconception based on a user model and a model of the discourse situation.

6. Acknowledgements

I would like to thank Julia Hirschberg, Aravind Joshi, Martha Pollack, and Bonnie Webber for their many helpful comments concerning this work.

7. References

[Brown & Burton 78]
Brown, J.S. and Burton, R.R. Diagnostic Models for Procedural Bugs in Basic Mathematical Skills. *Cognitive Science* 2(2):155-192, 1978.

[Grice 75] Grice, H. P. Logic and Conversation. In P. Cole and J. L. Morgan (editor), *Syntax and Semantics III: Speech Acts*, pages 41-58. Academic Press, N.Y., 1975.

[Jefferson 72] Jefferson, G. Side Sequences. In David Sudnow (editor), *Studies in Social Interaction*, . Macmillan, New York, 1972.

[Joshi 82] Joshi, A. K. Mutual Beliefs in Question-Answer Systems. In N. Smith (editor), *Mutual Beliefs*, . Academic Press, N.Y., 1982.

[McKeown 82] McKeown, K. . *Generating Natural Language Text in Response to Questions About Database Structure.* PhD thesis, University of Pennsylvania, May, 1982.

[Pollack et al. 82]
Pollack, M., Hirschberg, J., & Webber, B. User Participation in the Reasoning Processes of Expert Systems. In *Proceedings of the 1982 National Conference on Artificial Intelligence.* AAAI, Pittsburgh, Pa., August, 1982.

[Sidner 83] Sidner, C. L. Focusing in the Comprehension of Definite Anaphora. In Michael Brady and Robert Berwick (editor), *Computational Models of Discourse*, pages 267-330. MIT Press, Cambridge, Ma, 1983.

[Sleeman 82] Sleeman, D. Inferring (Mal) Rules From Pupil's Protocols. In *Proceedings of ECAI-82*, pages 160-164. ECAI-82, Orsay, France, 1982.

[Stevens & Collins 80]
Stevens, A.L. and Collins, A. Multiple Conceptual Models of a Complex System. In Richard E. Snow, Pat-Anthony Federico and William E. Montague (editor), *Aptitude, Learning, and Instruction*, pages 177-197. Erlbaum, Hillsdale, N.J., 1980.

[Stevens et al. 79]
Stevens, A., Collins, A. and Goldin, S.E. Misconceptions in Student's Understanding. *Intl. J. Man-Machine Studies* 11:145-156, 1979.

[Tversky 77] Tversky, A. Features of Similarity. *Psychological Review* 84:327-352, 1977.

[Woolf & McDonald 83]
Woolf, B. and McDonald, D. Human-Computer Discourse in the Design of a PASCAL Tutor. In Ann Janda (editor), *CHI'83 Conference Proceedings - Human Factors in Computing Systems*, pages 230-234. ACM SIGCHI/ HFS, Boston, Ma., December, 1983.