Artificial Intelligence Project--RLE and MIT Computation Center
Memo 45--

A Question-Answerer for Algebra Word Problems

by Daniel G. Bobrow

A Question-Answerer for Algebra Word Problems

Introduction

This is a proposal to write a program which, starting from input statements of problems in a restricted English, will be able to formulate problems symbolically and then solve problems from elementary algebra. The program will

1. Accept a restricted natural English as an input language.

2. Extract (on a semantic basis) relevant information from the input statement of the problem.

3. Find which of a stored set of relationships can be used to formulate the problem in algebraic terms to obtain a solution.

4. Add new relationships to this stored set in accord with an English statement of the relationship.

Background

Several question-answering programs have already been written. Among these are Synthex[1], Baseball[2] and SAD SAM[3]. Synthex takes a purely syntactic approach to question-answering. No attempt is made in Synthex, to determine the meaning of the input question. Information is stored as English text, and the machine compiles an index to occurrences of content words in the text. Tose content words (as opposed to function words such as "the" and "of") that appear in the question are extracted. Then those sentences in the input corpus which contain many of these words (in appropriate syntactic relationship) are proposed as answers to the question.

Baseball takes a first step towards understanding questions, in the sense that to a greater extent, the meanings of the words are used to retrieve information. For example, the two questions:

a. Who Beat the Yankees on Independence Day?

b. On July 4, the Yanks were defeated by what team?

have the same meaning, but few words in common. They are both transformed into the same "specification list":

TEAM winning = ?

TEAM losing = New York Yankees

Date = July 4

This common specification list is then used to retrieve the answer from a pre-stored data structure.

SAD SAM takes another step towards understanding English. It maps the input text onto a model which preserves the information needed to answer the questions. The subject of the questions is family relationships. Typical input statements are "John, Mary's brother, came to a supper." and "Mary's daughter, Ruth, had the red car." The irrelevant infromation is discarded, and "John," "Mary," and "Ruth" inserted at the proper nodes in a family tree. Then, although the relationship was never mentioned explicitly, the fact that "John is Ruth's uncle" can be computed from this model of family relations. It is this semantic (model building) approach to question answering which we hope to pursue.

The Program

The proposed program will answer questions requiring algebraic and other symbolic manipulation of input information given to it in English. This will be done by providing a model through which the input English statements can be interpreted as well as a mapping from sentences into this model

The model will consist of a set of relations each of which is represented by a string of symbols and possible interpretation for these symbols. For example, one might be $T = nC$, where T is the total cost of a group of items, C the cost for one item and $\underline{n}$ the number of items in the group. The mapping will determine under what conditions this is a relationship relevant to the solution of the problem, and which quantities given in the English input statement of the problem should be assigned to which variable. Once values have been assigned in the model, a symbolic processor, using elementary mathematical techniques, will be able to compute the answer to the question.

For preliminary processing, a syntactic analyser, similar to the one used in the Baseball Program, or in SAD SaM (say), would be used to parse the sentence. Useful cues, such as quantified noun phrases, would be extracted. Working backwards from the question, other relevant quantities would be found. For example, if the question asked were "What was the total cost of Johnny's books?" possible relationships involving total cost would be considered. From these we could see that the average cost per book, or the costs for the individual books are most likely to be relevant, if present, and not (usually) where Johnny is, or how long he took to get there.

The facts and relationships thus extracted are expressed in algebraic form. If the algebraic relationships thus found allow immediate solution, this is done. Otherwise a further search is made to find relationships involving unevaluated parameters. If the search is unsuccessful (or if a problem should arise in parsing a sentence), the computer will "complain" and interrogate the questioner. The questioner may then insert new information, such as a previously unknown relationship, or a new definition of a word, into the system. The program then processes this new sentence, using the same system of syntax analysis to extend the model itself.

<u>Examples</u>

The following are examples of the types of information that might be stored in the model.

a. "Amount" is a pronoun word which can replace any quantified noun phrase

b. Total Amount = Sum of individual amounts

C. Total Amount = (Number of individuals) multiplied by (Amount for <u>one</u> individual)

d. Total Amount = (Number of individuals) multiplied by (<u>Average</u> amount per individual)

e. One Dollar = 100 cents

The following is a typical (easy) question that might be asked of the program:

Q: John bought <u>five bananas</u> at the store. <u>One banana</u> costs seven cents. What was the total cost of the bananas? The preprocessor would excerpt the underlined phrases; then the requested item from the question would be generated, namely "total cost". This is a particular example of an item, "Total Amount", and "Amount" is replaced by "cost" (using relationship a). Relationships b, c, and d are then proposed as relevant, and examination of the first two sentences, would show that all the given information (noting the phrases underlined) can be mapped onto relationship c, and the question can then be answered.

A much harder example, which we hope to be able to do, illustrating the features of the program, is the following taken from Thomas' Calculus:

Q: "When air expands adiabatically, the pressure p and volume v satisfy the relationship $pv^{1.4}$ = constant. At a certain instant the pressure is 50 psi, and the volume is 32 in^3 and is decreasing at the rate of 4 in^3/sec. How rapidly is the pressure changing at this instant?"

The program must first abstract the infromation about the new relationship given and store it as part of the model. Volume and pressure satisfy the relationship given, when "expansion is adiabatic"--an expression which can be interpreted in the model as another relationship ($\Delta Q = 0$).

Then, form the context (in this case, the physical proximity of the expression), the program must decide to use this newly-added relationship. It must understand "rapidly" implies a question about a rate, in this case, an instantaneous rate. Then, to obtain this instantaneous rate, the relationship used must be differentiated and solved. This requires an elementary knowledge of the calculus--again expressible as a set of symbolic forms--and an ability to combine this knowledge with algebraic manipulations.

The facility to solve such multi-step problems, starting from an English language input, should be an important step toward achieving a reasonable measure of artificial intelligence.

CS-TR Scanning Project
Document Control Form

Date : 11 / 30 / 95

Report # AIM - 45

Each of the following should be identified by a checkmark:
Originating Department:

- [x] Artificial Intellegence Laboratory (AI)
- [ ] Laboratory for Computer Science (LCS)

Document Type:

- [ ] Technical Report (TR)
- [x] Technical Memo (TM)
- [ ] Other: ____________

Document Information

Number of pages: 5 (9-IMAGES)
Not to include DOD forms, printer intstructions, etc... original pages only.

Originals are:

- [x] Single-sided or
- [ ] Double-sided

Intended to be printed as :

- [x] Single-sided or
- [ ] Double-sided

Print type:

- [ ] Typewriter
- [ ] Offset Press
- [ ] Laser Print
- [ ] InkJet Printer
- [ ] Unknown
- [x] Other: MIMEOGRAPH

Check each if included with document:

- [ ] DOD Form
- [ ] Funding Agent Form
- [ ] Cover Page
- [ ] Spine
- [ ] Printers Notes
- [ ] Photo negatives
- [ ] Other: ____________

Page Data:

Blank Pages(by page number): ____________

Photographs/Tonal Material (by page number): ____________

Other (note description/page number):

Description :	Page Number:
IMAGE MAP:	(1-5) TITLE PAGE, 1-4
	(6-9) SCANCONTROL, TRGT'S (3)

Scanning Agent Signoff:

Date Received: 11 / 30 / 95 Date Scanned: 12 / 11 / 95 Date Returned: 12 / 14 / 95

Scanning Agent Signature: Michael W. Cook

Rev 9/94 DS/LCS Document Control Form cstrform.vsd