CROSSWORDS AND THE COMPUTER

A. ROSS ECKLER
Morristown, New Jersey

There is little question that the crossword is the most popular type of word puzzle in both England and America. Thousands are solved every year in newspapers and magazines by people who are unaware that the field of recreational linguistics stretches far beyond these limits. The cognitive processes that people use to solve word puzzles are of interest to both psychologists and computer programmers: what is human problem-solving behavior? can it be imitated by a computer? It is obvious that a computer is much faster and more accurate than a human being in repetitive arithmetic calculations or in list-sorting. However, it is not clear how a computer should be programmed to emulate (and, perhaps, improve upon) more creative activities such as playing chess, proving mathematical theorems, writing poetry, or translating one language into another.

Although much computer research has been directed at the problems listed above, much less has been done with word puzzles (excluding the highly secret cryptanalytic work by the NSA and similar organizations). This article briefly describes one computer program designed to solve crossword puzzles, another designed to construct them, and a third designed to solve double-crostics. The reader should be warned that all such programs are in their infancy; much more work is needed before they can be regarded as comparable to humans, either as constructors or solvers. The puzzles are extremely simple, the computer is supplied beforehand with a relevant vocabulary, and the solution is often incomplete.

Crossword Puzzle Solution

"A Program to Solve Crossword Puzzles", an unpublished Northwestern University M.S. thesis written by Helen Ann Bauer in 1973, employs a syntactical approach to the solution of crosswords. Her computer program consists of a list of about 2500 words arranged in small groups in the form of a synonym dictionary, together with rules for matching crossword definitions with words on this list. If a match is found, any synonym having the correct number of letters is considered to be a candidate for the solution. Each candidate synonym is tested for letter-matches with previously-filled-in intersecting words, and the one having the best agreement is selected for entry into the puzzle.

Her program has two interesting features. The first one, alluded to above, is its ability to match crossword definitions and dictionary lists. A few simple rules determine which word of a multi-word definition is to be regarded as the key word. The key word is then matched against the dictionary; if no match can be found, the key word is stripped of suffixes such as -s, -ed, -er, -est, -ing or -ly and tried again (both with and without an augmenting e to take care of words such as 'fired' or 'sighted'). Irregular past tenses, such as 'ran', are handled by special dictionary entries such as ran/ed.

The second feature is an interactive one: the computer, after filling in all the words it can, queries the puzzle-solver about those words it has partially filled in, inviting him to make corrections and (if desired) add the completed word to the synonym dictionary. This in principle allows the computer to improve the quality of its basic synonym dictionary and, therefore, the quality of its crossword solution.

The synonym dictionary was constructed from about 20 crossword puzzles taken from the "Easy" section of Dell Crossword Publications. The program was then tested on nine puzzles, four used for the dictionary and five new ones. Not surprisingly, the program did considerably better on the first four puzzles than on the last five.

The typical performance of the program on a 13-by-13 new puzzle is illustrated below. The definitions are listed line by line, with the correct answers in capitals:

Across

1. holy man SAINT
2. window drape CURTAIN
3. injure HARM   nasty MEAN
4. faith, --- and charity HOPE   myself ME   respond to a curtsy BOW
5. commotion FUSS   spare the --- and spoil the child ROD   faucet TAP
6. raw metal ORE   '--- Joey' PAL   odor SMELL
7. either OR   truthfulness HONESTY   musical note LA
8. restrict LIMIT   bashful SHY   wager BET
9. body of water SEA   dined ATE   loathe HATE
10. faint DIM   exists IS   barnyard 'cluckers' HENS
11. burden LOAD   gloomy DARK
12. get ready PREPARE
13. hobo TRAMP

Down

1. hoodwinks FOOLS
2. hastened HURRIED
3. water a lawn HOSE   post a letter MAIL
4. bottle lids CAPS   hello! HI   wash the floor MOP
5. certain SURE   cooking vessel POT   painting or sculpture ART
6. equip with weapons ARM   raced RAN   helper AIDER
7. that thing IT   bothers MOLESTS   Pittsburgh's state PA
8. appointed NAMED   that woman SHE   beaver's project DAM
9. make a knot in TIE   pigpen STY   stringed instrument HARP
10. seize NAB   '-- heart belongs to Daddy' MY   present HERE
11. short letter NOTE   money house BANK

12. billfolds WALLETS
13. dish PLATE

As illustrated below, the program completely filled in 33 of the 58 definitions, but four of these ('being' for 'saint', 'wet' for 'mop', 'gie' for 'tie', and 'erm' for 'arm') were wrong, leading to a score of 50 per cent correct.

```
        B E I N G
      C U R T A I N
    H A R M   M E A N
  H - P -   M E   B O W
F U S S   - - -     T A P
O R E   - - -   S M E L L
O R   H O N E S T Y   L A
L I M I T   - H Y   - E T
- E -     A - E   H A T E
  D - W   I S   H E - S
    - E - D   D A R K
      T - E - - R -
        T R A M P
```

After the program queried the puzzle-solver and the synonym vocabulary was updated, the score increased to 71 per cent correct (it did not achieve 100 per cent because the puzzle-solver deemed certain definitions, such as 'faith, hope and charity' for HOPE or 'barnyard cluckers' for HENS too specialized for the synonym list.

Crossword Puzzle Construction

Can a computer be used to construct crossword puzzles as well as solve them? Construction should, in principle, be easier, for there is no need to interpret crossword definitions; any word from the dictionary can be inserted into blank spaces in the puzzle, as long as its letters agree with those of previously-inserted intersecting words. In "Machine Selection of Elements in Crossword Puzzles: An Application of Computational Linguistics", on pages 51-72 of the March 1976 issue of the SIAM Journal on Computing, Lawrence J. Mazlack of the University of Guelph in Ontario, Canada, describes a program for constructing a crossword puzzle in a specified pattern of blanks, using a 2000-word vocabulary of words of four letters or less taken from Webster's Elementary Dictionary (for school-children). His program is designed to fill in letters one at a time rather than whole words, a task that would have taxed the capabilities of his computer.

How does the program work? In each iteration of the program, two decisions are made:

1) which blank space should be filled in next?
2) what letter should be put in it?

In choosing blank spaces, the program prefers those which are contained in four-letter words (harder to find than three-letter or two-letter ones), those in 'dense' locations (surrounded by blank spaces or previously-filled-in spaces on three or four sides), and those aligned with already-filled-in letters. In deciding which letter to put in, the program selects that letter which preserves as many dictionary options as possible. For example, if the program seeks to fill in the asterisked space in the diagram at the right, it prefers a letter such as E to a letter such as Y after noting (1) there are more words of the form --E- and --E in the dictionary than --Y- and --Y, (2) there are more words containing the trigram TRE in the dictionary than TRY, (3) there are more words containing the interrupted letter-pattern B-E in the dictionary than B-Y. (When the choice among letters is less clear-cut than this example, the program uses a maximin criterion to decide -- that is, it selects that letter for which the most restrictive option is as rich in possibilities as possible.)

```
    T
    R
B - *
    -
```

There is, of course, no guarantee that a letter-by-letter selection procedure will result in valid words; therefore, Mazlack also includes a back-up procedure to try again if the 'word' formed does not match a dictionary entry. In practice, he found that the program generated dictionary words about 85 per cent of the time. A more subtle defect of his program is its tendency to generate a given word more than once in the same puzzle, as illustrated below.

To start his program, Mazlack specifies a pattern of black squares and blank spaces, together with a filled-in word to prime the pump. The program successfully completed more than one-third of a set of puzzle patterns ranging from 4-by-4 to 13-by-13 in size. The largest one, given below, was primed with ARTS:

```
A R T S   P   T O   I T T
L A W     I   I     S H E
L I O N   T A N G     O R
Y D   O R   S E A   S U M
    W R A P     P I E
A R E   S E A R   T E N T
L O     H E R E         I
T E A R   P E N T   P A N
    R E D     T O R E
S H E   A R T   P A N G
L O     M A R T   N   A S
A L T     S E E       L O
M E O   T H E N   S E E D
```

It is a bit puzzling why words like YD, MEO, ALT and ITT appear in this puzzle; were these 'added' to the dictionary to complete the diagram, or did the program fail to check for 'words' in both the horizontal and vertical directions?

Limitations on computer speed and storage capacity make it doubtful that Mazlack's program can be generalized to construct crossword puzzles with words of any length drawn from a vocabulary of more realistic size (100,000 to 300,000 words, including plurals and derived forms).

Double-Crostic Puzzle Solution

One of the best-known varieties of the crossword is the double-crostic. A double-crostic consists of a set of definition-words, which must be inferred from short definition phrases as in a crossword, and a set of text-words which forms a complete sentence. The sentence is simply a transposal, or rearrangement, of the letters in the definition-words, so that one can work back and forth between the two halves -- a guessed definition-word implies a number of letters scattered throughout the text-words, and a guessed text-word implies a number of letters scattered through the definition-words.

The 1960 Proceedings of the Eastern Joint Computer Conference contains a paper by Edwin S. Spiegelthal entitled " Redundancy Exploitation in the Computer Solution of Double-Crostics". In contrast to Bauer's syntactical approach, he employs a purely statistical one -- that is, he judges the quality of a solution on the properties of its bigrams and trigrams as revealed in typical English-language text. Words of two or three letters are directly assessed, but longer words are judged only on the basis of their components, leading to a good chance that sequences such as ITHERD are allowed.

The idea underlying the operation of Spiegelthal's computer program is simple, although its implementation is relatively complex. Briefly, it proceeds through a series of stages, each stage consisting of the selection of a word from an 8551-word vocabulary (stored in the computer) which matches the letters of a partially-filled-in text-word (or definition-word), and a determination of the plausibility of the new bigrams and trigrams formed in the corresponding definition-words (or text-words). If these bigrams and trigrams pass certain tests, the new letters are filled in for both halves of the puzzle; if they do not, the text-word (or definition-word) is returned to its partial status and furthermore all letters of the 'bad' bigrams and trigrams are erased from both halves of the puzzle. This latter back-up feature, present in Mazlack but not Bauer, enables the computer to correct earlier words which looked plausible at the time they were filled in, but which eventually led to impossible letter combinations.

To get the computer program started, Spiegelthal supplies the computer with two to four definition-words for each definition, including the correct definition-word in many (but not all) cases. Had he not drastically restricted the possibilities in this way, the computer would have taken forever searching through its dictionary and trying all words of proper length for each definition-word.

With this big initial boost, his computer program was fairly successful in solving (or nearly solving) a number of relatively simple double-crostics. One example, incompletely documented by Spiegelthal, may give a flavor of the work. The definition-words were

quanta	halves	third	helmet	ease
cowbell	fodder	teethe	truant	sleet

and the corresponding text was

We hold these truths to be self evident that all men are created equal

At a late stage the following partial solution had been achieved:

---n---	halves	third	helmet	-a-e
cowbell	fodder	teethe	truant	s----

with the corresponding text

We hold -hes- tr--h- to be self evident that -l- men are cr--ted --ual

As Spiegelthal commented, any red-blooded American could finish it with his left hand tied behind its back, but the computer (having no left hand, or perhaps un-American in its outlook) was stymied by the fact that 'truths' did not appear in its dictionary but 'trophy' (matching the first, second and fifth letters) did. The program was unable to rectify this particular error (the only new trigram formed in the definitions was AYE), so after some thrashing about in which some earlier correct choices were despairingly erased, the program ground to a halt with the following piquant text:

We hold these trophy to be self evident that old men are cr--ted usual

Although Bauer's and Spiegelthal's computer programs for solving crosswords may be useful for understanding problem-solving behavior, they are clearly inadequate for anyone who wants a machine to help him with the daily newspaper puzzle. The small size of the vocabulary stored in the computer and the simple nature of the processing rules are drawbacks which might be overcome, but it is harder to see how one solves the syntactical problem of finding words corresponding to the wide variety of different definition phrases encountered. As we have seen, Spiegelthal dodged this problem entirely, and Bauer solved it in only a very limited way. If the syntactical problem could be solved, it is likely that machine translation of one language into another would be vastly improved, avoiding such amusing constructions as the (perhaps apocryphal) transmutation of 'the spirit is willing but the flesh is weak' into Russian and back into English as 'the whiskey is fine but the meat has gone bad'.