Regis Fillbin
A Crossword Puzzle Solver

Sam Hage

December 2015

Abstract

This paper describes a New York Times crossword puzzle solver, Regis Fillbin. The program takes its name from Dr. Matt Ginsberg's 2011 paper and corresponding program, "Dr.Fill" [1]. The program works by generating lists of candidate answers for each clue, then searching for the configuration of answers that best satisfies all constraints (constraints being the intersecting answers for other clues). The program was developed from September to December 2015. Work on both parts of this problem will continue through the Spring of 2016, focusing on both accuracy and performance.

1 Introduction

Crossword puzzles present several interesting problems from a computer science perspective, because they comprise a few processes at which humans remain superior to computers. Each clue, for example, corresponds to a specific location on the puzzle's grid, which location is not definitively described by the clue's single 'number', but must instead be located by finding the corresponding number on the grid. Nothing about the clue itself describes whether it runs 'across' or 'down'; this information comes simply from the header an indeterminate distance above the clue. Even the puzzle grid has a pattern of white (empty) and black squares, which determine the length of the clue's answer. These are computer vision problems, and are solved in this program by manually preprocessing the PDF image of the crossword puzzle into a plaintext representation of the clues and of the puzzle's structure [Fig. 1].

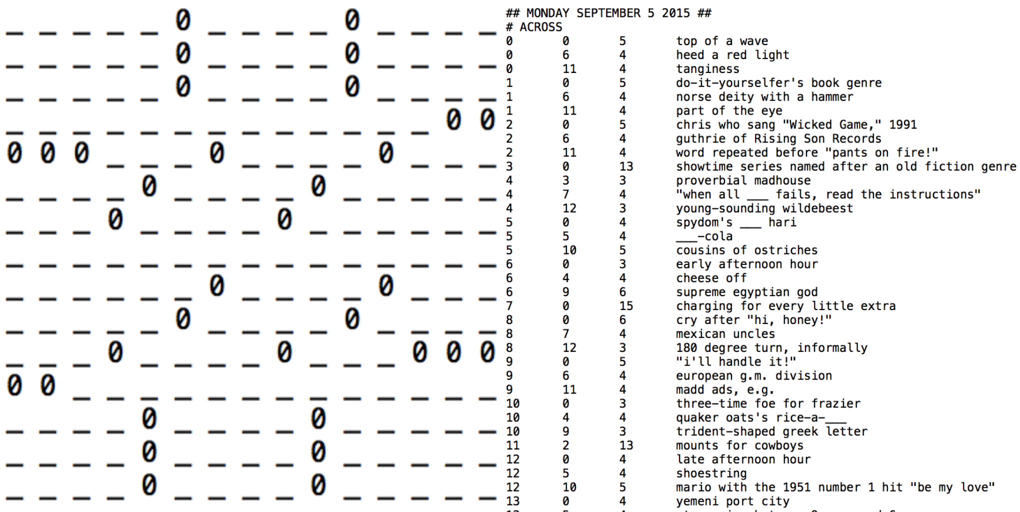

Figure 1: An example of the plaintext input given to the puzzle.

But there are other 'human intelligence' tasks that the program must accomplish, starting with the generation of possible answers from each clue. Regis FILLbin uses a combination of natural language processing and intelligent data analysis for this piece of the problem.

The final piece, on the other hand, represents a problem at which computers generally exceed humans. The reconciling of candidate answer lists is essentially a constraint satisfaction problem, which approach will be outlined later in this paper.

The New York Times crossword was chosen as the focus of the project because it is the best written and edited crossword, by general consensus among avid solvers. Consequently, the best crossword solving computer program should be able to solve the New York Times puzzle. Sample puzzles are also more readily available than from other sources.

The program is written mainly in Python, in addition to one JavaScript file that generates styled HTML to display the finished puzzles.

2 Answer Generation

Without the process of generating potential answers for each clue, the second part of the problem is not possible, so much of the work on this project was

```
('frying fat', 'LARD')
('frying liquid', 'CORNOIL')
('frying liquid', 'OIL')
('frying medium', 'CORNOIL')
('frying medium', 'DEEPFAT')
('frying medium', 'LARD')
('frying mess', 'SPATTER')
('frying need', 'FAT')
('frying pan hazard', 'SPATTER')
('frying pan mishap', 'SPATTER')
('frying pan sound', 'SSS')
('frying pan spray', 'PAM')
('frying pan', 'GRIDDLE')
('frying pan.', 'SPIDER')
('frying pans.', 'SPIDERS')
('frying sound', 'SSS')
('frying, for one', 'PAN')
('frying-butter sound', 'SSS')
('frying-pan coating', 'TEFLON')
('fryolator fill', 'DEEPFAT')
('fsu and nc state are in it', 'ACC')
('fsu or uf', 'SCH')
('ft. --- (former military base near monterey, ca)', 'ORD')
('ft. ---: former california military base, near monterey', 'ORD')
('ft. above land', 'ALT')
```

Figure 2: A screenshot of the data.

in this first process. This section will describe the algorithms and the data used.

2.1 Data

Regis FILLbin's primary data source is a corpus of single clue-answer pairs drawn from nearly twenty years of previous puzzles [Fig. 2]. It performs an initial search on this data, and another on a reduced version of the same data. The program also draws on a word list generated from a crossword dicitonary, and a parsed list of all Wikipedia article titles.

2.2 Modified Binary Search

The first search module is a binary search on the alphabetically sorted list of clues and answers. Rather than using exact string matching, this search uses the Levenshtein distance to judge whether two clues have the same meaning. The theory behind this type of search is that similar clues arise across many crossword puzzles. Certain clues, therefore, even from a brand new puzzle, will show up in or be a very close match for a clue in the data set. Experimentally, this search will generate a match for one-third to one-fourth of a puzzle's clues. The accuracy of these answers is classified with a high degree of confidence.

2.3 Fuzzy Search

Next a keyword-based search using Python's regex library is performed to generate answers for the rest of the clues. This module does not have the benefit of logarithmic performance, as the binary search does, and ends up much slower. The author of this paper had to prepare a demonstration of Regis FILLbin's capabilities in a short time frame and could not afford to run the fuzzy search on the whole module. Consequently, before performing the search, the data is compressed by randomly discarding pairs from the set. This leaves approximately one-third of the original data and results in much faster performance. On Monday and Tuesday puzzles, the difference in the solver's accuracy was not noticeable in the handful of trials run in each case.

This module generates many more candidates than the first. It is therefore able to find an answer where the first module cannot, but it also generates many false positives. A central challenge of the problem is eliminating these candidates to determine the correct one, a challenge that becomes even more difficult in the third module.

2.4 Letter Pattern Matching

The third and final module is performed repeatedly, after the first two. It takes all clues whose answers still have blank squares and runs these patterns of known and unknown letters against the two dictionaries mentioned in the data section. This module generates even more candidates, and therefore more false positives, since it ignores the clue entirely and focuses purely on the letters. For shorter answers in particular, this is problematic, as there are myriad three- and four-letter acronyms that show up in both the crossword dictionary and the list of Wikipedia articles. All of these come through as potential answers, and are therefore treated with a low degree of confidence.

```
3       10      down    ['units named for physicist enrico', '??????']  ['FERMIS']
4       13      down    ['land chronicled by c. s. lewis', '??????']    ['NARNIA']
4       14      down    ["grammar nazis' concerns", '??????']   ['USAGES']
5       6       down    ['greeting in rio', '???']      ['OLA']
5       11      down    ["skirt's edge", '???'] ['HEM']
7       8       down    ['six-sided roller', '???']     ['DIE']
10      5       down    ['dupont acrylic fiber', '?????']       ['ORLON']
10      10      down    ['kind of energy with panels', '?????'] ['SOLAR']
12      1       down    ['poem of praise', '???']       ['ODE']
0       11      down    ['letter after x-ray and yankee in the nato alphabet', '????']  ['ECHO', 'ZULU']
7       3       down    ['floored, as a boxer', '???']  ['MAS', 'ALI']
11      12      down    ["slight hitch in one's plans", '????'] ['REUP', 'SNAG']
1       6       across  ['norse deity with a hammer', '????']   ['THOR', 'ODIN', 'IDUN']
0       9       down    ['nudges', '?????']     ['PRODS', 'PESTS', 'POKES']
11      3       down    ['vases', '????']       ['MING', 'URNS', 'OILS']
0       1       down    ['by any other name it would smell as sweet, per juliet', '????']       ['ROSE', 'NAME',
'ARAB', 'IKEA']
0       6       across  ['heed a red light', '????']    ['STOP', 'EMTS', 'PINK', 'IDLE', 'CORN', 'EXIT']
8       12      across  ['180 degree turn, informally', '???']  ['SSE', 'ESE', 'ENE', 'UEY', 'DOA', 'ELL']
10      0       across  ['three-time foe for frazier', '???']   ['ORC', 'ALI', 'FOE', 'RAF', 'NAT', 'NEO']
13      5       across  ['store sign between 9 a.m. and 6 p.m.', '????']        ['AGUN', 'ARGO', 'ALMA', 'SIMS',
'RHEE', 'MOON', 'SOSA', 'ABMT']
12      0       down    ['pre-airconditioning cooler', '???']   ['ITD', 'ICE', 'ACS', 'ADE', 'CON', 'FAN', 'PEN',
'TEA', 'EIS']
6       4       down    ['fish that can attach itself to a boat', '??????']     ['FINGER', 'PLAQUE', 'REMORA',
'ELOISE', 'OTTAWA', 'AEGEAN', 'BODEGA', 'PEELER', 'CLEAVE', 'BUCKLE']
9       0       across  ['"i\'ll handle it!"', '?????'] ['ONONE', 'HITME', 'LETME', 'WIELD', 'EWERS', 'PAWED', 'NAPES',
'IDTAG', 'IDAHO', 'TEXAS', 'WAGON', 'ALIAS']
```

Figure 3: Lists of answers generated by the three search modules.

```
Module 1:
"___ Beta Kappa (3)"  -->  ["PHI"]


Module 2:
"___ Beta Kappa (3)"  -->  ["PHI", "TAU", "ETA", "RHO", "CHI", "PSI"]


Module 3:
"P?I"  -->  ["PPI", "PLI", "PSI", "PTI", "PCI", "PAI", "PHI", "PRI", "PII", "PEI", "PJI", "PKI", "PNI"]
```

Figure 4: An example of one clue over the three modules.

3 Filling the Puzzle

Compared to the generation of answers, the filling of the puzzle is much less complex. Consider, for comparison, the problem of solving a sudoku puzzle, in which each square of a nine by nine grid may contain any integer from 1 to 9, such that no row, column, or three by three sub grid (of which there are also nine) contains duplicate integers. A simple computer approach (though not the most intelligent or performant) is to assign possible numbers one at a time, eliminate possibilities from other squares, and follow a depth first search through the puzzle, backtracking when necessary. This approach was considered for Regis FILLbin, but abandoned because of the difficulty in spanning the problem space. In order for the recursive backtracking to work on its own, each subdomain (a single square in a sudoku puzzle and a single clue in a crossword) must contain all its possibilities at the start of the filling process, or the search may dead end. This is not acceptable in solving crosswords, since a successful program must be able to fully execute even when complete accuracy is not possible.

Instead, Regis FILLbin's approach is similar to the way a human would solve a sudoku puzzle (and, of course, a crossword); it begins with the answers in which it is the most confident, moving to less confident answers, updating other answer lists by eliminating candidates that would conflict with crossing answers. Although, strictly speaking, the program uses unweighted constraint satisfaction (the individual candidates are not weighted within the lists), in effect it does contain an element of weighting. Answers gathered from the first module are treated with greater confidence because of their closeness to previously seen clues. What's more, the candidates from shorter answer lists are incidentally weighted higher, because there are simply fewer possibilities. This is similar to the approach taken by Littman et al. [2], and used in the website based on their program, oneacross.com.

4 Results

As noted earlier this program is in progress, and only minimal testing has been done so far. Repeated tests on Monday and Tuesday puzzles have shown good results, generally over 75 percent of squares filled correctly. This author found this performance acceptable for a prototype, but actually worse than most humans.

```
C R E S T 0 S S G P 0 Z E S T
H O W T O 0 T H U R 0 U V E A
I S A A K 0 A U E O 0 L I A R
R E N N Y L R T R D F U L 0 0
0 0 0 Z O O 0 T R S E 0 G N U
M A T A 0 D O L E 0 R H E A S
O N E 0 R I L E 0 A M E N R A
R D E M E N A C D E I M I N G
R E H A M E 0 T I A S 0 U I E
O N E S O 0 U S M C 0 P S A S
W B C 0 R O N S 0 P S I 0 0 0
0 0 C H A R L E R H O R S E S
F I V E 0 L A C E 0 L A N Z S
A D E N 0 O C K N 0 A T A R I
N E S S 0 N E S T 0 R E G A L
```

Figure 5: Sample plaintext output of the program.

Because of random data compression, one could expect highly variable performance between trials. While some variance is evident, the solutions tend to be similar, likely because of a high degree of redundancy in the full data set.

5 Limitations

As discussed above, crossword puzzles are specifically designed as a challenge for humans. Many of their particular nuances are extremely difficult for a machine, and have not yet been incorporated into Regis Fillbin.

Many puzzles have themes; related longer answers that appear throughout the grid and are generally tied together with a single "theme clue". These themes almost always involve wordplay, intentional misspelling, or other humor; famously difficult problems for computers. Regis Fillbin's natural language processing algorithms will fail to correctly interperet such wordplay, as these puns were essentially designed to do.

Richtext, non-alphanumeric characters, and so-called "rebus" puzzles also exceed the program's capabilities. These puzzles will have a special character, multiple characters, or even an entire word in a single square, generally requiring some level of human intelligence to interperet them correctly. Other puzzles will have a shape described in the black squares of the grid itself and

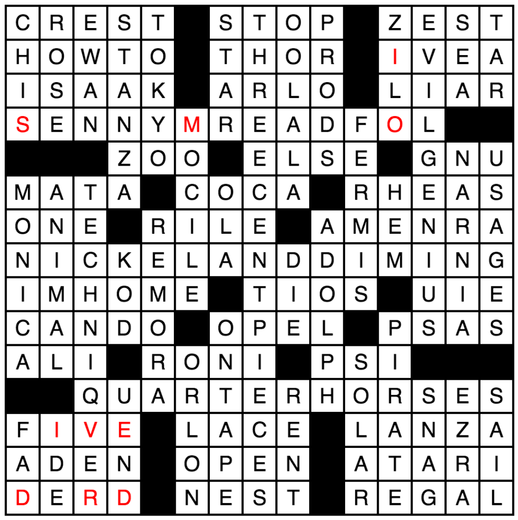

Figure 6: A solved Monday puzzle with incorrect letters in red.

Figure 7: A solved Tuesday puzzle with incorrect letters in red.

Figure 8: A puzzle with a DNA structure depicted in the black squares. DNA is referenced in the five theme clues and the 'revealer'.

referenced from the clues.

In-puzzle reference also poses a problem. Fourteen-across may, for instance, say "With 5-Across, lime green nocturnal insect", while five-across simply reads "See 14-Across". Since each clue is processed individually in the current version of the program, that clue provides no information about the actual corresponding answer.

Still, some of these are more easily surmountable than others, and future work on the program will focus on resolving limitations.

6 Future Work

The biggest focus of the next semester's work on Regis FILLbin will be resolving the current tradeoff between speed and accuracy. Because of the random data compression, vital data may simply be discarded on a given trial. Without compressing the data in some way, however, the run time is too long to be truly practical (15 minutes compared to 4). The easiest way to resolve this will be partitioning the large data into subsets based on answer length, allowing the second search module to run on a smaller data set without actually discarding any relevant data. There is also room to improve the program's performance by incorporating more natural language

Figure 9: A rebus puzzle whose theme is Theseus and the Minotaur.

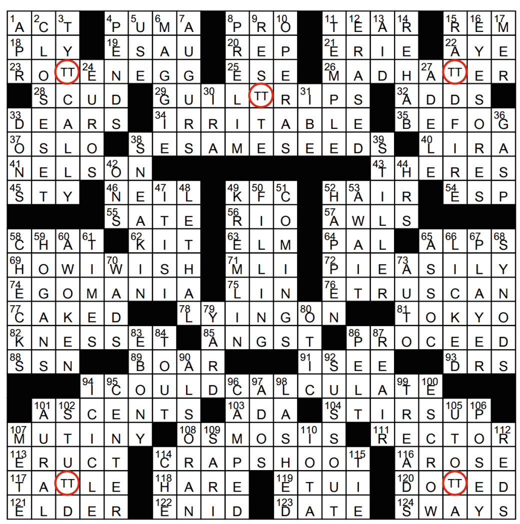

Figure 10: A Pi Day puzzle with five occurences of the Greek letter pi. The character reads as the letters 'p' and 'i' from top to bottom, and as a double 't' from left to right.

processing and less data analysis.

An interesting extension of the project will also be to experiment with computer vision and optical character recognition to allow the program to read a puzzle directly from its PDF, instead of preprocessing the puzzle into various plaintext inputs.

References

[1] Ginsberg, M.L.: Dr.fill: Crosswords and an implemented solver for singly weighted csps. CoRR abs/1401.4597 (2014), `http://arxiv.org/abs/1401.4597`

[2] Littman, M.L., Keim, G.A., Shazeer, N.: A probabalistic approach to solving crossword puzzles. Artificial Intelligence 134(1-2), 23–55 (2002)

[3] Shazeer, N.M., Littman, M.L., Keim, G.A.: Solving crossword puzzles as probabilistic constraint satisfaction. In: National Conference on Artificial Intelligence. pp. 156–162 (1999)

[4] Shazeer, N.M., Littman, M.L., Keim, G.A.: Solving crosswords with proverb. In: National Conference on Artificial Intelligence. pp. 914–915 (1999)