Webcrow: a web-based crosswords solver.

Giovanni Angelini, Marco Ernandes, and Marco Gori

Dipartimento di Ingegneria dell'Informazione,
Via Roma 56, 53100 Siena, ITALY,
{angelini, ernandes, marco }@dii.unisi.it
http://airgroup.dii.unisi.it
http://webcrow.dii.unisi.it

Abstract. Webcrow is a software system whose aim is to solve crosswords. Problems of like solving crosswords have been informally defined as AI-Complete and are extremely challenging for machines. Webcrow represents the first solver for Italian crosswords and the first system that tackles this language game using the Web as knowledge base. Currently, Webcrow is competitive against an average crossword player. Crosswords that are "easy" for expert humans (i.e. crosswords from the cover pages of *La Settimana Enigmistica* [1]) are solved, in a 15 minutes time limit, with 80% of correct words and over 90% of correct letters. Webcrow well performs on crosswords that are designed for experts and, in general, it outperforms an average undergraduate student on this kind of crosswords.

1 Introduction

Crossword puzzles are probably the most popular language game played around the world. It is challenged every day by millions of people and it requires a combination of skills. These include a good comprehension of the clues expressed some times in an ambiguous or tricky way, a large knowledge base and a clever heuristic in order to fill correctly the puzzle.

Problems like solving crosswords from clues are reputed as AI-complete [1]. This complexity is due to its semantics and the large amount of general knowledge required.

AI started developing an interest for crossword solving only recently. The first experience reported in the literature is the Proverb system [2] that reached human-like performances on American crosswords using a great number of knowledge-specific expert modules and a crossword database of great dimensions.

Webcrow's challenge is to become competitive against expert crosswords solvers and to achieve this using the web as its primarily source of knowledge and without knowledge-specific modules, that are severely problematic to maintain. At the moment, the system presented here is basic, nevertheless it is giving very promising results.

2 The System

We believe that recent developments in computer technology, such as the Web, search engines, information retrieval and machine learning techniques, can enable computers

[1] The most popular Italian crossword magazine.

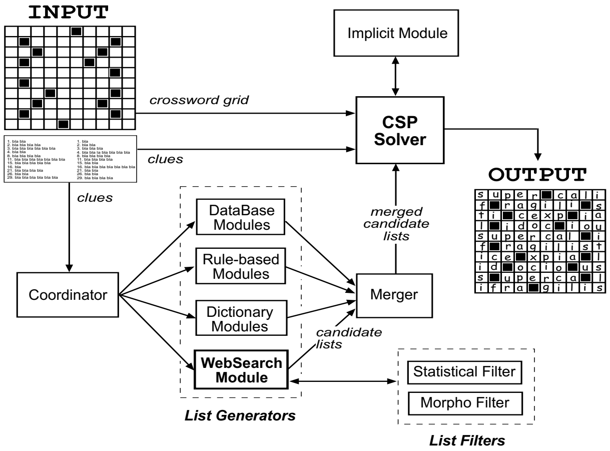

Fig. 1. WebCrow. A general overview of WebCrow's architecture.

to enfold with semantics real-life concepts. With this in mind we designed a software system whose major assumption is to attack crosswords (within competition time limits) making use of the Web, being this the most extremely rich and self-updating repository of human knowledge.

Following, it will be given a brief description of the system. Further details should be found in the works [3] and [4].

2.1 An Overview

Figure 1 shows the general information flows of the system. Webcrow processes each crossword puzzle by solving two main subproblems. Firstly, for each clue a list of candidate solutions is generated. Secondly, the grid is filled making use of the confidence scores of each clue-list and the probabilities associated to each candidate in the lists. The latter task has to be done taking into account the intrinsic constrains of the puzzle.

2.2 Clue Answering

Solving crosswords means to associate each clue to the correct word answer. This task enables us to investigate a novel and very challenging version of the question answering problem, which we call clue-answering. Clue-answering differs from question answering in various forms:

- there is no standard interrogative form.
- there is an intrinsic and volunteer ambiguity.
- the topic of the question can be both factoid and non-factoid.
- there is a unique and precise correct answer: a single or a compound word.
- very high recall is required: reaching close to 100% recall is more important than to place the correct answer in the very first position of the candidate list. Missing the target could lead to disaster in grid-filling.

– the only evident advantage in crossword solving is that we know in advance the exact length of the words that we are seeking. Nevertheless, this feature does not provide a great advantage with average-length words.

This task is primarily achieved through the Web Search Module. In this module, each clue goes through a reformulation step, which aims to give to the Search Engine the most effective queries in oder to download useful web documents. These documents are then parsed and candidate solutions are extracted from them. The candidates are finally scored using a statistical filter and a morphological filter. The first one scores the candidates in a IDF-TF fashion, taking into account the distance of the candidate word from the clue words in each document. The second one gives a score according to the morphological classes associated to the candidate word inside the documents using a PoS tagger and the morphological classes guessed by a kernel-based machine which takes as input the clue itself.

There are other three modules. The Data-Based Module returns possible candidates making exact and partial matching on the clues of solved crosswords. The Rule-Based Module deals with clues whose answers have no semantic relation, but that are cryptically hidden inside the clues them-self. Finally, the Dictionary Module is used to increase the global coverage of the clue-answering.

2.3 Filling the Grid

In order to fill the grid, WebCrow has to face a probabilistic constrain-satisfaction problem. From each clue list a candidate has to be chosen and inserted in the crossword-puzzle, trying to satisfy the intrinsic constrains. The aim of this phase is to find an admissible solution which maximize the number of correct words inserted.

Due to the competition time restrictions and to the complexity of the problem the use of the standard optimization algorithm A*was discarded. We adopted as a solving algorithm a CSP version of WA*. In our implementation the path cost function is the product of the probabilities of the already assigned variables and the heuristic function is the product of the best remaining values of the unassigned variable.

The grid-filling module works together with the Implicit Module in order to overcome the missing of a word within the candidates list. Whenever a variable x_i remains with no available values then a heuristic score is computed by taking the tetra-gram probability of the pattern present in x_i.

3 Performances

The whole crossword test set (sixty examples) has been partitioned in five subsets. The first two belong to *La Settimana Enigmistica*, T^1_{ord} containing examples of ordinary difficulty (mainly taken from the cover pages of the magazine) and T^1_{dif} composed by crosswords especially designed for skilled cruciverbalists. An other couple belongs to *La Repubblica*, T^2_{new} and T^2_{old} respectively containing crosswords that were published in 2004 and in 2001-2003. Finally, T^3 is a miscellaneous of examples from crossword-specialized web sites.

Table 1. Statistics of the test subsets. # of examples of each subset reported in brackets.

	T^1_{ord} (15 ex.)	T^1_{dif} (10 ex.)	T^2_{new} (15 ex.)	T^2_{old} (10 ex.)	T^3 (10 ex.)
# Letters	160.7	229.5	156.6	141.1	168.5
# Blanks	69.4	37.5	29.8	31.3	37.8
Clues	59.7	79.5	59.5	61.4	50.5
Words correct	**80.0**%	**67.6**%	**62.9**%	**61.3**%	**69.1**%
Letters correct	**90.1**%	**81.2**%	**72.0**%	**72.9**%	**82.1**%

Table 1 gives some statistics on each test set and the average performances of Webcrow. Altogether, WebCrow's performance over the test set is of 68.8% correct words and 79.9% correct letters.

Furtherly, we observed WebCrow's performances on a preliminary human vs. machine experiment. Webcrow challenged a group of 25 undergraduate students. The students solved four crosswords extracted randomly from the test set with the same 15 minute time limit. The comparison took into consideration the number of correct words inserted. Our system averaged better than its human competitors: only three students managed to outperform WebCrow and none of them beat him in all the four examples.

4 Conclusions

One of our main objectives is to build a system competitive with human experts in solving crosswords, and hopefully challenge masters in a real competition. Furthermore, we aim for a system capable of solving crosswords in different languages by exploiting language-independent and data-driven techniques, such as machine learning, avoiding (or limiting) pre-compiled rules, as usually done by question answering systems.

Leaving aside a justified cognitive curiosity towards the problem, the development of a technology able to face this challenging task provides significant insights in real-world applications like automatic dictionary creation, support for text-summarization, writing prompting aid (in text editors), support for classical question answering and solutions to problems involving probabilistic constraints.

References

1. Michael L. Littman: Review: computer language games. Journal of Computer and Games. **134** (2000) 396–404
2. Michael L. Littman, Greg A. Keim and Noam M. Shazeer: A probabilistic approach to solving crossword puzzles. Journal of Artificial Intelligence. **134** (2002) 23–55
3. Marco Ernandes, Giovanni Angelini and Marco Gori: WebCrow: a WEB-based system for CROssWord solving. AAAI '05: Proceedings of the twentieth national conference on Artificial Intelligence. (2005)
4. Giovanni Angelini, Marco Ernandes and Marco Gori: Solving italian crosswords using the web. AI*IA '05: Proceedings of the 9th congress for the Italian Association for Artificial Intelligence. (2005)