Wordnet Enhanced Automatic Crossword Generation

Aoife Aherne and **Carl Vogel**
Computational Linguistics Group
University of Dublin, Ireland

Abstract

We report on a system for automatically generating and displaying crosswords from a system manager supplied database of potential clues and corresponding words that index those clues. The system relies on the lexical relations encoded in WordNet to enhance the aesthetics of the resulting crossword by making it easier to automatically identify a grid that may be populated with words and clues that have a thematic focus. The system architecture is provided in overview, as is empirical evaluation.

1 Introduction

This paper builds on past work in automatic crossword generation, describing a system enhancement enabled by the availability of WordNet (Fellbaum, 1998), and other freely available software resources. The paper begins by discussing the problem of automatic crossword generation, and some past developments in that area. The task is to use a database of answers and clues, based on a symmetric grid, such as the example in Fig. 3, automatically generating the grid, populating the grid with available clues, and formatting the presentation in useful ways. The grids are styled on the nature of *The New York Times* crossword puzzle, which typically have a theme. Further enhancing the aesthetics of the system, modest amounts of morphological analysis are incorporated to avoid the appearance of the answer or some morphologically related words in the clues. Complete details on the system are available; this includes user guidelines, installation and full implementation details, and inter-package interfacing.[1]

Several facts can be noted about the crossword in Fig 3. One is that it is automatically typeset in LaTeX, and another is that the answers are supplied. This is due to having interfaced the system constructed with LaTeXusing `crosswrd.sty`.[2] This package includes a binary flag which allows the user to print the solved puzzle or the open puzzle without the answers to the cues. Notice that a series of asterisks is used to block out instances of the answer itself or derived forms of it from the provided clue. Another fact is that it has the unsatisfying property of numerous two letter answers. A more satisfying puzzle is given in Fig. 4

Our work on automatic crossword generation is inspired by that of (Berghel, 1987) who developed a method for compiling fixed grids into Prolog predicates used to determine solutions (in the sense of an answer key, not about intelligent solving of preset crosswords but intelligent setting of crosswords to be solved: this is inherently a constraint solving problem in its own right) based on lists of lists corresponding to words, with shared variables constraining their interlocking. Subsequent work aimed to improve upon this (Berghel and Yi, 1989) by beginning the optimization of grid/answer key constraint satisfaction. Since then, work has been done to both further optimize the automatically constructed Prolog code (Gibbons, 2002), and to render the system accessible by a web based interface, and further began working

[1] See (Aherne, 2005), available at: https://www.cs.tcd.ie/courses/csll/aoaherne0405.pdf.

[2] The author of crosswrd.sty, as distributed with the current version of MiKTeX, is Frank Mittelbach. Mittelbach extended Brian Hamilton Kelly's version of the *.sty file to ensure its compatibility with LaTeX2e, the latest version of LaTeX. Our system automaticaly constructs a text file with suitable text mark-up to automatically generate *.dvi, *.ps and *.pdf files compatible with this style file and corresponding to the automatically generated grids, solutions and clue sets. See (Aherne, 2005) for full details.

on the aesthetic issue of themed crossword construction with levels of solver based complexity entered as a parameter by the users. The data source for this was the 1917 Webster's Dictionary, which is available online, as it is out of copyright. (Graham, 2003) overhauled that system completely, providing a Web-based interface akin in functionality to many online newspaper crossword puzzles: potential for greater solver complexity provided by improving the efficiency of overall setting of the puzzle, verification for the solver of correctness of answers so far, a facility for giving up altogether. The present paper summarizes work detailed at greater length elsewhere (Aherne, 2005), which extended the project further: enabling the server to accommodate password controlled discrimination of user types (e.g., a crossword solver in many cases should not have access to adding to or deleting from the database), thus assisting in pedagogical applications, such as vocabulary learning within subject disciplines or language learning by providing the teacher with access to do exactly that; enabling multi-platform presentation of the crosswords by automatically typesetting the puzzles using LaTeXand transformed into Postscript and PDF printable output, such that the output file can be easily printed and distributed in printed form as overnight homework assignments when users might not have access to the crossword server; and most importantly for the context of this paper, extremely enhanced control over the thematic nature of the puzzle, through relying on WordNet as a lexical resource in constructing occasionally enigmatic, but nonetheless thematic crossword puzzles.

In this paper we outline the algorithms used in the integration and implementation of WordNet within the automatic themed crossword generation system. Furthermore we present an informal evaluation of the system where the participants are asked to "guess" the theme of the puzzle. A more detailed discussion of these and other characteristics of the system (differential access, LaTeXprinting interface)are presented in (Aherne, 2005). We conclude our discussion with directions for future work.

2 Evolution of Automatic Crossword Generation

Consider the Grid in Fig. 1.[3] Obviously, it has no blanks, and thus describes a puzzle in which each of the four 'down' answers intersects with each of the four 'across' answers. The approach taken by Berghel to solve grids like this was to construct a Prolog Horn clause as in Fig. 2, and evaluate it with respect to a Prolog database of facts `word`, and related to a Prolog built-in predicate which relates a list of ASCII codes to an atomic word (the first argument). The predicate `word/n` is true of the list of ASCII representations of the letters in a word recorded in the Prolog database as an assertion of the fact (e.g. `word(112,97,99,101)` corresponds to the word 'pace').

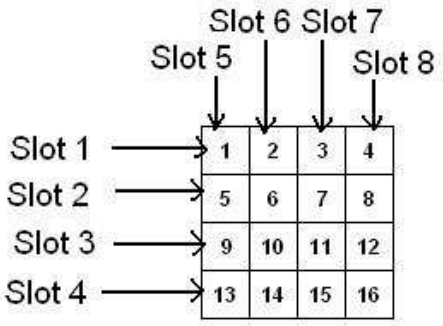


Figure 1: A Four by Four Grid.

An obvious computational time complexity improvement can be gained by using the 'fail-first' strategy of constraint logic programming. That is, verify the satisfiability of each of the 'word' predicates before attempting the 'name' predicates to generate the 'returned' values as the list of slot fillers. An advance of prior work (Gibbons, 2002; Graham, 2003) was in controlling the order of the 'word' predicates themselves, following the 'fail-first' ethic. Other constraints were also introduced to maximize the chances of generating a viable grid and potential solution within a reasonable amount of time: no duplicate words are allowed in the solution;[4]

[3]This figure is borrowed from, and associated discussion encapsulates aspects of related work (Graham and Vogel, 2005).

[4]This trait is shared with the *New York Times* cross-

```
solution([SLOT_1, SLOT_2, SLOT_3, SLOT_4, SLOT_5, SLOT_6]):-

        word(C1, C2, C3, C4),
        name(SLOT_1, [C1, C2, C3, C4]),

        word(C5, C6, C7, C8),
        name(SLOT_2, [C5, C6, C7, C8]),

        word(C9, C10, C11, C12),
        name(SLOT_3, [C9, C10, C11, C12]),

        word(C13, C14, C15, C16),
        name(SLOT_4, [C13, C14, C15, C16]),

        word(C1, C5, C9, C13),
        name(SLOT_5, [C1, C5, C9, C13]),

        word(C2, C6, C10, C14),
        name(SLOT_6, [C2, C6, C10, C14]),

        word(C3, C7, C9, C15),
        name(SLOT_7, [C3, C7, C9, C15]),

        word(C4, C8, C12, C16),
        name(SLOT_8, [C4, C8, C12, C16]).
```

Figure 2: Berghel's Representation of the 4x4 Crossword Grid

words that cause impossible combinations in interlinking words are disallowed; heuristics are used to evaluate particular words as candidates for appearing in the solution; backtrack to replace a prior word only if the word has an impact on the slot that is currently unfillable and hence forcing backtracking.[5] See (Graham and Vogel, 2005) for details about some of these strategies. Here we comment just on the heuristic, as this relates to an important issue in the WordNet-enabled enhancement described later.

The 'best' choice of a word suggested for the grid is one that is not already suggested as part of the solution. The proposed word is examined for each of its letters that interlink with other slots in the grid. For each such intersecting slot, a record is kept of the number of words that are compatible as an intersecting word. The word for the original slot that has the least number of compatible intersections is most likely to cause trouble later on. Thus, that heuristic value is maximized when choosing the filler for the slot to be intersected with. This is only one such strategy that we have explored to date. Other improvements on the original (Berghel, 1987; Berghel and Yi, 1989) which we take as the starting point for our work are discussed further elsewhere (Gibbons, 2002; Graham, 2003; Graham and Vogel, 2005). This section is intended just to give the reader a sense of the underlying constraint satisfaction problem, implemented in the context of Prolog's underlying depth-first theorem prover, in fitting clues to a grid, via words that fit the grid. Grid generation by random seed is described by (Gibbons, 2002).

word.

[5]Note that symmetry can involve disconnected sections of the grid (recall Fig. 3); hence, backtracking across a segment that is independent of one causing problems is a waste of time.

3 Integrating WordNet to Improve Crossword Generation

The system whose evolution has been described so far represents an advance, but also has room for improvement. The full system offered by (Graham, 2003) is clearly coded and very well presented. The work reported in (Graham, 2003) includes exceptional advances in improving the execution time of the program through modification of the Prolog code that is generated. Despite the provision of a viable algorithm, the crosswords produced are very loosely related and at times totally unrelated to the theme specified. However, the fault does not lie within the algorithm employed, it lies with the dictionary database it exploits. Due to the nature of the dictionary, including its limited size, it was necessary, in order that enough words be returned to create a crossword, to select all words and definitions within which the string representation of the target word could be found. Hence, given the theme "sport", words like "transport" and indeed clues containing "transport" would be selected. Similarly the target theme "cat" returns words and clues containing the substring "cat".

On a more practical issue, users cannot ever retrieve or save a copy of the crossword to complete at a later date — they are obliged to finish the crossword in one sitting. An ideal solution to this problem would be the automatic generation of printable versions of the crossword at the users request. In a classroom environment this would be particularly useful, allowing an entire class to work on the same crossword.

Secondly, in (Graham, 2003) the solution to the clue very often appears in the clue text. Graham removes exact occurrences of the solution, but has not dealt with the solution in its plural or participle forms.

A security issue, again, with respect to potential use as a language learning tool in a classroom environment is raised with access to

the database. (Graham, 2003) did not restrict database access. Access must be restricted such that only those with sufficient privileges (e.g. "root" /"teacher") may add/modify databases on the system. Those with insufficient privileges (e.g "student") should only be able to generate and solve crossword puzzles.

We address the system architecture modifications elsewhere (Aherne and Vogel, 2005), followed by a discussion of the enhancements enabled by WordNet in §3.1. Examples in the appendix show that we are still only part way to solving some of these problems (e.g. morphological analysis of clues for derivatives of the answer itself), and indeed, clue 7 down in Fig 4 demonstrates this also.

3.1 Why Integrate WordNet?

WordNet was selected as its inherent structure is closer to that of a thesaurus than to that of a traditional dictionary. Hence *meaning searches* as opposed to *word searches* are carried out on WordNet. This allows the retrieval of words related to a particular theme and bootstraps the lexical resource supplied by the original dictionary into a far larger resource.

Secondly, the source code for this linguistic tool is freely available. Numerous other developers have made available a variety of interfaces to WordNet including a MySQL database version, generated from the Prolog source code and can be sourced from (WordNet2, 2004).

3.2 Some Implementation Details

For the purposes of automatic themed crossword generation WordNets hyponym/hypernym hierarchy is exploited. This permits the collection of words that are more and less specifically related to a given theme. Hence an algorithm exploiting this data structure would ensure that only words directly relevant to the theme be returned to the program.

By selecting all of the synsets in which a given theme word appears a considerable collection of thematically related words can be made available to the program. Furthermore, by taking each one of those synsets and searching for its hyponym and hypernym synsets (should such relations exist) the search space from which words can be selected increases greatly. This step should be iterated many times, each time taking the synsets returned by the previous iteration as a starting point, thus maximising coverage within the WordNet database.

3.3 Structural Changes to the WordNet Database

Word length is of primary importance to the crossword generator. The WordNet database does not record the length of each individual word. With the MySQL version of WordNet it is possible to easily manipulate the primary table "synset" such that the number of characters in each word was included as a tuple in the table relation. The Java methods constructed for the system queried not only for words that were thematically related to each other but also words that pertained to the theme AND were of a length required to satisfy the grid.

3.4 Polysemy Problems

The occurence of multiple interpretations of a given theme submitted by the user is not considered problematic, rather a positive phenomenon. Firstly, the Prolog parser only allows one occurrence of each word in any grid. Secondly, as the user is only asked to submit a singular noun or verb (e.g. "sport") to denote their preferred theme, the system will not "second guess" which instance of sport the user has in mind. Polysemy thus enriches the cryptic nature of the crossword and enhances user experience. The table below illustrates the multiple occurrences of the word sport in WordNet and the definitions associated with each synset:

Word	Definition	POS
sport	an active diversion...	n
sport	occupation of athletes who compete for pay	n
sport	verbal wit	n
sport	temporary summer resident of inland Maine	n
sport	organism with characteristics resulting from chromosomal alteration	n
sport	someone who engages in sports	n
sport	play boisterously	v
sport	wear/display in an ostentatious manner	v

Table 1: WordNet Results for "sport"

4 Basic Algorithm

1. User submits a valid theme
2. MySQL base query for primary synsets
3. MySQL query for hyponym synsets of each synset in 2
4. Iterate 3 five times

5. MySQL query for hypernym synsets of each synset in 2
6. Iterate 5 five times
7. Selecting a word of appropriate length from each synset returned by steps 3 & 5

4.1 Algorithm Explained

- A *Valid* Theme

As described in the Section 3.2, the semantic relations pertinent to the retrieval of words belonging to a particular theme are the "Hyponym" and "Hypernym" relations. These relations can only be applied to singular common nouns and verbs. Hence the user must submit either a singular common noun or a verb to the program for the algorithm to function correctly.

- *Primary* Synsets

The term "primary" synsets is attributed to the synsets in which the subject word occurs. This is the starting point of future queries. All words contained within the primary synsets are made available to the Prolog parser which selects the final word set that satisfies the grid.

- *Hyponym & Hypernym* Synsets

A hyponym synset of a base synset contains words that are more "specifically" related to a particular topic that those words in the base synset. Similarly, a hypernym synset of a given synset will contain words that are more "generally" related. Using the *primary synsets* as an intial starting point, steps 3 and 5 search for hyponym and hypernym synsets of the primary synsets. The first iteration of step 3 and 5 queries for hyponym and hypernym synsets of the primary synsets returned in step 2. However successive iterations of steps 3 and 5 use the last set of hyponym/hypernym synsets returned as their starting point. This creates chains of related hyponym and hypernym synsets.

- Words of *Appropriate* Length

This algorithm must satisfy two separate constraints. Firstly, the words returned must be thematically related to each other. Secondly, the words returned must be able to satisfy the constraints of the random grid generated. That is to say, only words whose length corresponds to the requirements of the grid are accepted. Only one word is selected from each synset - thus ensuring that each clue presented to the user is unique.

4.2 An Iterative Problem

The number of synsets returned by a particular query cannot be determined in advance. This computational problem was solved in the following manner. Each query is executed twice. The first time the total number of synset IDs returned by the query is noted and a data structure of that size is created. The second time the query is executed the synset IDs are stored in the data structure specially created to accommodate them.

4.3 The Magic Number 5

As can be observed in the algorithm, the searches for hyponym and hypernym synsets are iterated five times each. This is to ensure that all words associated with the submitted theme are returned. The approach adopted follows statistics related to the WordNet 1.7 hierarchy presented by (Devitt and Vogel, 2004; Devitt, 2004).[6] Particularly relevant is the following:[7]

- 78.5% of synsets are "leaf-synsets"
- Maximum synset-leaf distance is 5
- Minimum synset-leaf distance is 2

Leaf synsets are situated at the extremes of the WordNet hierarchy in that no other synset inherits from a leaf: there are no hyponym synsets of leaf synsets. Of even greater importance is the fact that the maximum distance from any synset to a leaf synset is five synsets. Hence by iterating the hyponym and hypernym searches five times, full coverage is ensured.

4.4 Removing Multiple Forms of Solution From A Clue

The work presented in (Graham, 2003) does not successfully remove all forms of the solu-

[6] This work grew out of a need to quantify relative specificity of terms when attempting to calculate the content-based similarity of pairs of texts (Devitt and Vogel, 2003; Devitt, 2004).

[7] Other recent work also takes advantage of or otherwise analyzes the topology of WordNet, e.g. (Agirre et al., 2004; Farreres et al., 2004; Teich and Fankhauser, 2004)

tion from the clue. To partially address this problem, it was decided to extend modify inherited programs. The aim was to remove the solution whether it occurred at the end or in the middle of a sentence, in its plural, past particible, present particible, adverbial and superlative forms. (Graham, 2003) successfully removes occurrences of the solution in all parts of the clue, but does not deal with other morphological forms of the solution. An outline of the extension to the method is given below and involves searching for representative strings:

- Solution (ans) in middle of sentence
  ... + ans + ...
- Solution at the end of a sentence
  ... + ans + {"." | "!" | "?" | ":" | ";"}

Plurals, present and past participles are found in a similar manner and are sought both in the middle or at the end of the sentence:

- Plural Forms in the middle of a sentence
  ... + ans + "s" + ...
- Past & Present Participles
  ... + ans + {"ed" | "d" | "ing" + ...}
- Adverbial & Superlative Forms
  ... + ans + {"ly" | "ily" | "er" | "est" + ...}

This clearly does not take into account those words whose plural or other forms do not follow general morphological rules. In many instances exceptional cases result in a complete change to the word itself — (e.g. eat : ate) thus ensuring that the actual solution would not appear in the clue text, therefore coverage of exceptional cases is in many instances not required. Obviously, there remains room to expand on this (see (Aherne and Vogel, 2005) for examples of clues which slip through this minimal morphological analysis; e.g., 'cup' is blanked from the clue while 'cupped' remains in the sample of usage accompanying the definition). Possibly just parsing the definitions, currently used as the text of the clues, to eliminate sample usage would ameliorate the situation without recourse to deep morphological analysis, but certainly deeper analysis is necessary, both for regular patterns and exception cases. Recording exceptions needn't be computationally expensive if a minimized trie data structure is used, or an effective caching mechanism.

5 Independent Analysis of Themed Crosswords

To determine whether or not the system actually succeeds in creating themed crosswords an experiment was carried out. In this experiment a total of twenty individuals, both male and female, spread between the ages of 18 and 60 were presented with a completed crossword puzzle produced by the program. Their task was to "Guess the Theme". There was no time limit set for this task, but participants were asked to follow their "gut" instinct. Participants could respond in sentence form, single word or a couple of words that they felt best described the theme of the crossword. A total of eight crosswords were presented to the participants. The following themes were covered: Sport, Water, Music, Pick, Match, Food, Animal, Colour.

Themes were selected to provide a varying level of abstraction and hence difficulty. Of the themes selected sport, water, pick, match and colour can be interpreted as a noun or a verb. Food, animal and music are all nouns. It would be expected that the first group pose more problems than the second given the extra degree of complexity. Furthermore, given that pick is primarily employed as a verb the crossword generated contains numerous verbal synonyms and is hence more difficult to identify its theme. The crosswords presented to the participants can be viewed in (Aherne and Vogel, 2005).

Participants were asked to state whether or not they ever studied linguistics and if so in what capacity. After the experiment, the subjects noted which crossword they found most difficult and which they found easiest.

5.1 Interpretation Of Responses

In order to accurately analyse results a sliding scale was employed. It was decided that responses that included the exact word entered be considered as positive results. Words that could be considered as being part of the same domain were also accepted but accorded less importance. An example of a word/phrase belonging to a particular domain is: "athletic skill", a member of the "sport" domain. Domain membership was verified using WordNet's hyponym and hypernym semantic relations. Responses that were outside of the subject domain were considered as being negative results.

5.2 Results and Discussion

The crosswords generated to test participants views of thematic coherence of the puzzles are provided in (Aherne and Vogel, 2005); unfortunately, space does not permit their inclusion here. Statistical results are detailed in Table 2.

n=20	Correct	Domain	Incorrect
Crossword 1	70%	20%	10%
Crossword 2	30%	70%	0%
Crossword 3	90%	10%	0%
Crossword 4	0%	10%	90%
Crossword 5	0%	10%	90%
Crossword 6	90%	10%	0%
Crossword 7	60%	40%	0%
Crossword 8	70%	20%	10%

Table 2: Crossword Evaluation Results

Predictablly, themes were largely successfully chosen by the participants for crosswords 1, 2, 3, 6, 7 and 8 — only 10% of responses to crosswords 1 and 8 were entirely unrelated to the theme. Crosswords 4 & 5 scored poorly given the high level of abstraction associated with the theme words selected: pick and match. Although the participants were not informed of the syntactic category that the theme words belonged to, the responses returned for crosswords 4 & 5, although outlandish, were largely verbs.

6 Conclusions

Obviously, WordNet has been incorporated in a wide range of applications. This paper supplies yet another application that makes crucial advantage of the lexical resource.

Through the integration of WordNet, the crossword generator now successfully generates "themed" crosswords. The findings of the crossword evaluation survey confirm this fact in that subjects generally converged on descriptions closely related to the seed themes, except where the seed was perhaps too abstract in nature. This should be analyzed further in terms of the location of the seeding theme with respect to its location in WordNet's topology, paying attention to the fact that the topology is not evenly distributed in its coverage. Since WordNet is hand collated, some concepts have been developed in greater detail than others. This is highlighted in (Devitt, 2004) with this example:

- Drogheda: site of 16th century battle
- Limerick: poem or a port city

Given that both Drogheda and Limerick are Irish towns of similar size, it is unusual that Drogheda is not considered a city.

The algorithm employed for the extraction of thematically related words ensures that all relevant words are returned to the program for consideration. Adaptation of the WordNet2 MySQL database to include the word length property contributes to this process. Another distinct advantage of the current system is that it opens door to easily integrating EuroWordNet and its Inter-Lingual-Index, which will allow the production of multi-lingual themed crosswords, where the clues could be presented in one language (e.g. French), with the solutions expected in the target language (e.g. English).

Careful theme selection produces interesting results. Conventional themes, like for example "language", "country", "food", "sport" produce consistently good crosswords, indicating that in a language learning environment, crosswords can be reliably produced for "popular" themes . More "original" themes, like for example "pick" or "match", provide highly-interesting, abstract crosswords which require a greater level of lateral thinking to determine their theme. These could be described as "verbally synonymous" crosswords, where light is shed on the relationship between verbs and expressions as opposed to noun-noun relationships. Other aspects of the existing system are hinted at here, but not fully detailed. For example, the Appendices show the fruit of the LaTeXinterface and modest morphological manipulation of clues. Mechanisms for differential permission to database access and update rights depending on user types and passwords are provided, but the implementation of those features is not detailed here. Finally a user's installation guide is available, as is a general user's help guide: see (Aherne, 2005).

In further work, an obvious extension of the current program would be the inclusion of other WordNets. Given that all WordNets are built on the same underlying structure, this aspect should be relatively easy to implement. Of greater difficulty, however, would be the integration of WordNet's inter-lingual index, that links synsets in one language with corresponding synsets in another language. This would render the generation of multi-lingual themed

crosswords possible. A more efficient algorithm for the generation of random symmetric grids free of one and two letter words would enhance the program's execution time. At present, grids are fully complete before they are tested for one and two letter words. Ideally, grids should be tested at regular intervals during the grid building process to remove doomed grids as early as possible. Decreasing the number of black squares present in a grid will inevitably result in fewer grids containing one and two letter words. An interesting study could involve the comparison of themed crosswords generated by the crossword generator and the New York Times themed crosswords. Not only would this highlight the quality or otherwise of the generated crosswords, it may provide an insight into which concepts require further development within WordNet. Expansion of the morphological analysis to avoid repeated word forms or elaborations of the theme in clues would also be valuable. Finally, overcoming the technical difficulties currently experienced with Linux operating systems the Java Virtual Machine and SICStus Prolog would make the program accessible to a wider audience.

References

Eneko Agirre, Enrique Alfonseca, and Oier Lopez de Lacalle. 2004. Approximating hierarchy-based similarity for wordnet nominal synsets using topic signatures. In *Second International Wordnet Conference*, pages 15–22. Brno, Czech Republic.

Aoife Aherne and Carl Vogel. 2005. Crossing wordnet with crosswords, netting enhanced automatic crossword generation. TCD-CS-2005-52.

Aoife Aherne. 2005. Generation of themed portable crosswords: A wordnet implementation. Centre for Computing and Language Studies, Trinity College, University of Dublin. Bachelor of Arts (Moderatorship) in Computer Science, Linguistics and a Language. Final Project Dissertation.

H. Berghel and C. Yi. 1989. Crossword compiler compilation. *The Computer Journal*, 32(3):276–280.

H. Berghel. 1987. Crossword compilation with horn clauses. *The Computer Journal*, 30(2):183–88.

A Devitt and Carl Vogel. 2003. Using wordnet hierarchies to pinpoint differences in related texts. In *Proceedings of EUROLAN 2003, Ontologies, and Information Extraction International Workshop*, pages 37–44. Bucharest, Romania.

A Devitt and Carl Vogel. 2004. The topology of wordnet: Some metrics. In *Second International Wordnet Conference*, pages 106–111. Brno, Czech Republic.

A Devitt. 2004. *Methods for Meaningful Text Representation and Comparison*. Ph.D. thesis, Trinity College Dublin.

Javier Farreres, Karina Gibert, and Horacio Rodrigues. 2004. Towards binding spanish senses to wordnet senses through taxonomy alignment. In *Second International Wordnet Conference*, pages 259–264. Brno, Czech Republic.

Christiane Fellbaum, editor, 1998. *WordNet: An Electronic Lexical Database*. MIT Press, first edition.

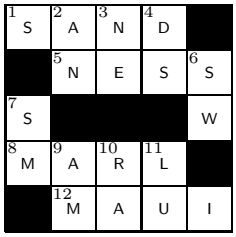
Anne-Marie Gibbons. 2002. Random crossword generator with a web interface. Department of Computer Science, Trinity College, University of Dublin. Bachelor in Engineering. Final Project Dissertation.

Yvette Graham and Carl Vogel. 2005. Computer construction of crossword puzzles using horn clauses and constraint programming.

Y Graham. 2003. Random crossword generator with a web user-interface. Centre for Computing and Language Studies, Trinity College, University of Dublin. Bachelor of Arts (Moderatorship) in Computer Science, Linguistics and a Language. Final Project Dissertation.

Elke Teich and Peter Fankhauser. 2004. Wordnet for lexical cohesion analysis. In *Second International Wordnet Conference*, pages 326–331. Brno, Czech Republic.

WordNet2, 2004. *WordNet2 Reference Manual*. Princeton University, http://wordnet.princeton.edu/doc.



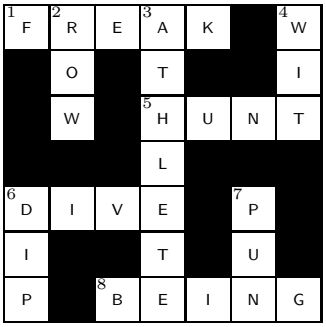
ACROSS

1 a loose material consisting of grains of rock or coral (4)
5 a strip of land projecting into a body of water (4)
8 a loose and crumbling earthy deposit consisting mainly of calcite or dolomite; used as a fertilizer for soils deficient in lime (4)
12 the second largest of the Hawaiian Islands (4)

DOWN

7 a master's degree in science (2)
2 ***** associate degree in nursing (2)
9 modulation of the amplitude of the (radio) carrier wave (2)
3 a midwestern state on the Great Plains (2)
10 ancient hawk-headed Egyptian sun god; a universal creator (2)
4 an honorary degree in science (2)
11 a trivalent metallic element of the rare earth group; usually occurs in association with yttrium (2)
6 the compass point midway between south and west; at 225 degrees (2)

Figure 3: Themed Crossword: Earth—Grid Containing 1 & 2 Letter 'Words'



ACROSS

1 a person or animal that is markedly unusual or deformed (5)
5 the pursuit and killing or capture of wild animals regarded as a sport (4)
6 a headlong plunge into water (4)
8 a living thing that has (or can develop) the ability to act or function independently (5)

DOWN

6 a brief swim in water (3)
2 the act of ****** as a sport (3)
3 a person trained to compete in sports (7)
7 a humorous play on words; 'I do it for the ****** of it'; 'his constant punning irritated her' (3)
4 a message whose ingenuity or verbal skill or incongruity has the power to evoke laughter (3)

Figure 4: Themed Crossword: Sport—Grid Free of 1 & 2 Letter Words

[8] https://www.cs.tcd.ie/publications/tech-reports/reports.05/TCD-CS-2005-52.pdf