

WASPBENCH: a lexicographer’s workbench supporting state-of-the-art word sense disambiguation.

Adam Kilgarriff, Roger Evans, Rob Koeling
Michael Rundell, David Tugwell

ITRI, University of Brighton

Firstname.Lastname@itri.brighton.ac.uk

1 Background

Human Language Technologies (HLT) need dictionaries, to tell them what words mean and how they behave. People making dictionaries (lexicographers) need HLT, to help them identify how words behave so they can make better dictionaries. Thus a potential for synergy exists across the range of lexical data - in the construction of headword lists, for spelling correction, phonetics, morphology and syntax, but nowhere more than for semantics, and in particular the vexed question of how a word’s meaning should be analysed into distinct senses. HLT needs all the help it can get from dictionaries, because it is a very hard problem to identify which meaning of a word applies. Lexicographers need all the help they can get because the analysis of meaning is the second hardest part of their job (Kilgarriff, 1998), it occupies a large share of their working hours, and it is one where, currently, they have very little to go on beyond intuition and other dictionaries.

Thus HLT system developers and corpus lexicographers can both benefit from a tool for finding and organizing the distinctive patterns of use of words in texts. Such a tool would be an asset for both language research and lexicon development, particularly for lexicons for Machine Translation. We have developed the WASPBENCH, a tool that (1) presents a “word sketch”, a summary of the corpus evidence for a word, to the lexicographer; (2) supports the lexicographer in analysing the word into its distinct meanings and (3) uses the lexicographer’s analysis as the input to a state-of-the-art word sense disambiguation (WSD) al-

gorithm, the output of which is a “word expert” which can then disambiguate new instances of the word.

2 WASPBENCH

2.1 Grammatical relations database

The central resource of WASPBENCH is a collection of all grammatical relations holding between words in the corpus. WASPBENCH is currently based on the British National Corpus¹ (BNC): 100 million words of contemporary British English, of a wide range of genres. Using finite-state techniques operating over part-of-speech tags, we process the whole corpus finding quintuples of the form:

$$\{\text{Rel}, W1, W2, \text{Prep}, \text{Pos}\}$$

where Rel is a relation, W1 is the lemma of the word for which Rel holds, W2 is the lemma of the other open-class word involved, Prep is the preposition or particle involved and Pos is the position of W1 in the corpus. Relations may have null values for W2 and Prep. The database contains 70 million quintuples.

The inventory of relations is shown in Table 1. There are nine *unary* relations (ie. with W2 and Prep null), seven *binary* relations with Prep null, two *binary* relations with W2 null and one *ternary* relation with no null elements. All inverse relations, ie. **subject-of** etc, found by taking W2 as the head word instead of W1 are explicitly repre-

¹<http://info.ox.ac.uk/bnc>

relation	example
bare-noun	the angle of bank ¹
possessive	my bank ¹
plural	the banks ¹
passive	was seen ¹
reflexive	see ¹ herself
ing-comp	love ¹ eating fish
finite-comp	know ¹ he came
inf-comp	decision ¹ to eat fish
wh-comp	know ¹ why he came
subject	the bank ² refused ¹
object	climb ¹ the bank ²
adj-comp	grow ¹ certain ²
noun-modifier	merchant ² bank ¹
modifier	a big ² bank ¹
and-or	banks ¹ and mounds ²
predicate	banks ¹ are barriers ²
particle	grow ¹ up ^p
Prep+gerund	tired ¹ of ^p eating fish
PP-comp/mod	banks ¹ of ^p the river ²

Table 1: Grammatical Relations

sented, giving six extra binary relations² and one extra trinary relation, to give a total of twenty-six distinct relations. These relations provide a flexible resource to be used as the basis of the computations of WASPBENCH.

The relations contain a substantial number of errors, originating from POS-tagging errors in the BNC, attachment ambiguities, or limitations of the pattern-matching grammar. However, as the system finds high-salience patterns, given enough data, the noise does not present great problems.

2.2 Word Sketches

When the lexicographer starts working on a word, s/he enters the word (and word class) at a prompt. Using the grammatical relations database, the system then composes a **word sketch** for the word. This is a page of data such as Table 2, which shows, for the word in question (W1), ordered lists of high-salience grammatical relations, relation-W2 pairs, and relation-W2-Prep triples for the word.

The number of patterns shown is set by the user, but will typically be over 200. These are listed for each relation in order of salience³, with the

²**and-or** is considered symmetrical so does not give rise to a new inverse relation.

³Salience is estimated as the product of Mutual Infor-

count of corpus instances. The instances can be instantly retrieved and shown in a concordance window. Producing a word sketch for a medium-to-high frequency word takes around ten seconds.⁴

2.3 Matching patterns with senses

The next task is to enter a preliminary list of senses for the word, in the form of some arbitrary mnemonics, perhaps MONEY, CLOUD and RIVER for three senses of *bank*. This inventory may be drawn from the user's knowledge, from a perusal of the word sketch, or from a pre-existing dictionary entry.

As Table 2 shows, and in keeping with "one sense per collocation" (Yarowsky, 1993) in most cases, high-salience patterns or **clues** indicate just one of the word's senses. The user then has the task of associating, by selecting from a pop-up menu, the required sense for unambiguous clues. Reference can be made at any time to the actual corpus instances, which demonstrate the contexts in which the triple occurs.

The number of relations marked will depend on the time available to the lexicographer, as well as the complexity of the sense division to be made. The act of assigning senses to patterns may very well lead the lexicographer to discover fresh, unconsidered senses or subsenses of the word. If so, extra sense mnemonics can be added.

When the user deems that sufficient patterns have been marked with senses, the pattern-sense pairs are submitted to the next stage: automatic disambiguation.

2.4 The Disambiguation Algorithm

WASPBENCH uses Yarowsky's decision list approach to WSD (Yarowsky, 1995). This is a bootstrapping algorithm that, given some initial seeding, iteratively divides the corpus examples into the different senses. Given a set of classified collocations, or **clues**, and a set of corpus **instances** for the word, the algorithm is as follows:

mation and log frequency. Our experience of working lexicographers' use of Mutual Information or log-likelihood lists shows that, for lexicographic purposes, these over-emphasise low frequency items, and that multiplying by log frequency is an appropriate adjustment.

⁴A set of pre-compiled word sketches can be seen at <http://www.itri.brighton.ac.uk/adam.kilgarriff/wordsketches.html>

subj-of	num	sal	obj-of	num	sal	modifier	num	sal	n-mod	num	sal
lend	95	21.2	burst	27	16.4	central	755	25.5	merchant	213	29.4
issue	60	11.8	rob	31	15.3	Swiss	87	18.7	clearing	127	27.0
charge	29	9.5	overflow	7	10.2	commercial	231	18.6	river	217	25.4
operate	45	8.9	line	13	8.4	grassy	42	18.5	creditor	52	22.8
modifies			PP			inv-PP			and-or		
holiday	404	32.6	of England	988	37.5	governor of	108	26.2	society	287	24.6
account	503	32.0	of Scotland	242	26.9	balance at	25	20.2	bank	107	17.7
loan	108	27.5	of river	111	22.1	borrow from	42	19.1	institution	82	16.0
lending	68	26.1	of Thames	41	20.1	account with	30	18.4	Lloyds	11	14.1

Table 2: Extract of word sketch for *bank*

1. assign instances containing a classified clue to the appropriate sense
2. for each clue C (already classified or not)
 - for each sense, count the instances where C holds which are assigned to it
 - identify C’s ‘preferred’ sense P
 - calculate the ratio of C-instances assigned to P, to C-instances assigned to some sense other than P
3. order clues according to the value of the ratio to give a ‘decision list’
4. assign each instance to a sense according to the first clue in the decision list which holds for the instance
5. if all instances are classified (or no new instances have been newly classified/re-classified on this iteration, or some other stopping condition is met) STOP; else return to step 2

Yarowsky notes that the most effective initial seeding option he considered was labelling salient corpus collocates with different senses. The user’s first interaction with WASPBENCH is just that.

At the user-input stage, only clues involving grammatical relations are used. At the WSD algorithm stage, some “bag-of-words” and n -gram clues are also considered. Any content word (lemmatised) occurring within a k -word window of the nodeword is a bag-of-words clue. (The user can set the value of k . The default is currently 30.) N -gram clues capture local context which may not be covered by any grammatical relation. The n -gram clues are all bigrams and trigrams including the nodeword.

Yarowsky’s algorithm was selected because it operated with easily human-readable clues, integrated straightforwardly with the WASPBENCH *modus operandi*, and was or was close to being the highest-performing system in the SENSEVAL evaluations (Kilgarriff and Rosenzweig, 2000; Edmonds and Kilgarriff, 2002). The algorithm is a “winner-take-all” algorithm: for an instance to be disambiguated, the first matching context in the decision-list is identified, and this alone classifies the data instance⁵.

3 Evaluation

Evaluation presented a number of challenges:

- We straddle three communities - commercial dictionary-making, HLT/WSD research, commercial/research MT - each with very different ideas about what makes a technology useful.
- There are no precedents. WASPBENCH performs a function – corpus-based disambiguating-lexicon development with human input – which no other technology performs. This leaves us with no points of comparison.
- On the lexicography front: human analysis of meaning is decidedly ‘craft’ rather than ‘science’. WASPBENCH aims to help lexicographers do their job better and faster. But there is no tradition for even qualitative, let alone

⁵Recent work (Yarowsky and Florian, 2002) has suggested that the winner-take-all strategy is not always the best strategy if the best clue is not a very good clue. In future work we would like to extend the WASPBENCH to take account of this insight.

quantitative, analysis of performance at this task, either for speed or quality of output.

- A critical question for commercial MT would be “does it take less time to produce a word expert using WASPBENCH, than using traditional methods, for the same quality of output”. We are constrained in pursuing this route, being without access to MT companies’ lexicography budgets or strategies.

In the light of these issues, we have adopted a ‘divide and rule’ strategy, setting up different evaluation themes for different perspectives. We pursued five approaches:

SENSEVAL – seen purely as a WSD system, WASPBENCH performed on a par with the best in the world (Tugwell and Kilgarriff, 2001).

Expert review – three experienced lexicographers reviewed WASPBENCH very favourably, also providing detailed feedback for future development.

Comparison with MT – students at Leeds University⁶ were able to produce (with minimal training) word experts for medium-complexity words in 30 minutes which outperformed translation of ambiguous words by commercially-available MT systems (Koeling et al., 2003).

Consistency of results – subjects at IIIT, Hyderabad, India⁷ confirmed the Leeds result and established that different subjects produced consistent results from the same data (Koeling and Kilgarriff, 2002).

Word sketches – lexicographers preparing the new Macmillan English Dictionary for Advanced Learners (Rundell, 2002) successfully used word sketches as the primary source of evidence for the behaviour of all medium and high frequency nouns, verbs and adjectives (Kilgarriff and Rundell, 2002).

These evaluations demonstrate that WASPBENCH does support accurate, efficient, semi-automatic, integrated meaning analysis and WSD

lexicon development, and that word sketches are useful for lexicography and other language research.

The WASPBENCH can be trialled at <http://wasps.itri.brighton.ac.uk>.

References

- Philip Edmonds and Adam Kilgarriff. 2002. Introduction to the special issue on evaluating word sense disambiguation systems. *Journal of Natural Language Engineering*, 8(4).
- Adam Kilgarriff and Joseph Rosenzweig. 2000. Framework and results for English SENSEVAL. *Computers and the Humanities*, 34(1–2):15–48. Special Issue on SENSEVAL, edited by Adam Kilgarriff and Martha Palmer.
- Adam Kilgarriff and Michael Rundell. 2002. Lexical profiling software and its lexicographical applications - a case study. In *EURALEX 02*, Copenhagen, August.
- Adam Kilgarriff. 1998. The hard parts of lexicography. *International Journal of Lexicography*, 11(1):51–54.
- Rob Koeling and Adam Kilgarriff. 2002. Evaluating the WASPBENCH, a lexicography tool incorporating word sense disambiguation. In *Proc. ICON, International Conference on Natural Language Processing*, Mumbai, India, December.
- Rob Koeling, Adam Kilgarriff, David Tugwell, and Roger Evans. 2003. An evaluation of a Lexicographer’s Workbench: building lexicons for Machine Translation. In *Proc. EAMT workshop at EACL03*, Budapest, Hungary, April.
- Michael Rundell, editor. 2002. *Macmillan English Dictionary for Advanced Learners*. Macmillan, London.
- David Tugwell and Adam Kilgarriff. 2001. WASPBENCH: a lexicographic tool supporting WSD. In *Proc. SENSEVAL-2: Second International Workshop on Evaluating WSD Systems*, pages 151–154, Toulouse, July. ACL.
- David Yarowsky and Radu Florian. 2002. Evaluating sense disambiguation performance across diverse parameter spaces. *Journal of Natural Language Engineering*, 8(4):In press. Special Issue on Evaluating Word Sense Disambiguation Systems.
- David Yarowsky. 1993. One sense per collocation. In *Proc. ARPA Human Language Technology Workshop*, Princeton.
- David Yarowsky. 1995. Unsupervised word sense disambiguation rivalling supervised methods. In *ACL 95*, pages 189–196, MIT.

⁶We would like to thank Prof. Tony Hartley for his help in setting this up.

⁷We would like to thank Prof. Rajeev Sangal and Mrs. Amba Kulkani for their help in setting this up.