ETS Lexical Associations System for the COGALEX-4 Shared Task

Michael Flor
Educational Testing Service
Rosedale Road
Princeton, NJ, 08541, USA
mflor@ets.org

Beata Beigman Klebanov
Educational Testing Service
Rosedale Road
Princeton, NJ, 08541, USA
bbeigmanklebanov@ets.org

Abstract

We present an automated system that computes multi-cue associations and generates associated-word suggestions, using lexical co-occurrence data from a large corpus of English texts. The system performs expansion of cue words to their inflectional variants, retrieves candidate words from corpus data, finds maximal associations between candidates and cues, computes an aggregate score for each candidate, and outputs an *n*-best list of candidates. We present experiments using several measures of statistical association, two methods of score aggregation, ablation of resources and applying additional filters on retrieved candidates. The system achieves 18.6% precision on the COGALEX-4 shared task data. Results with additional evaluation methods are presented. We also describe an annotation experiment which suggests that the shared task may underestimate the appropriateness of candidate words produced by the corpus-based system.

1 Introduction

The COGALEX-4 shared task is a multi-cue association task: finding a target word that is associated with a set of cue words. The task is motivated, for example, by a tip-of-the-tongue search application, as described by the organizers: "Suppose, we were looking for a word expressing the following ideas: 'superior dark coffee made of beans from Arabia', but could not remember the intended word 'mocha'. Since people always remember something concerning the elusive word, it would be nice to have a system accepting this kind of input, to propose then a number of candidates for the target word. Given the above example, we might enter 'dark', 'coffee', 'beans', and 'Arabia', and the system would be supposed to come up with one or several associated words such as 'mocha', 'espresso', or 'cappuccino'."

The data for the shared task were sampled from the Edinburgh Associative Thesaurus (**EAT** - http://www.eat.rl.ac.uk). For each of about 8,000 stimulus words, the EAT lists the associations (words) provided by human respondents, sorted according to the number of respondents who provided the respective word. Generally, when more people provided the same response, the underlying association is considered to be stronger (Kiss et al., 1973). For the COGALEX-4 shared task, the cues were the five strongest responses to an unknown stimulus word, and the task was to recover (guess) the stimulus word (henceforth, **target** word). The data for the task consisted of a training set of 2000 items (for which target words were provided), and a test set of 2000 items. The origin of the data was not disclosed before or during the system development and evaluation phases of the shared task competition.

The ETS entry consisted of a system that uses corpus-based distributional information about pairs of words in English. No use was made of human association data (EAT or other), nor of any other information such as the order of importance of the cue words, or any special preference for the British spelling often used in the EAT.

Zock/Rapp/Huang (eds.): Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon, pages 35–45,
Dublin, Ireland, August 23, 2014.

2 The ETS system for computing multi-cue association

Our system is defined by the following components.

1. Corpus from which the distributional information about word pairs is learned, along with preprocessing steps (database generation).
2. The kind of distributional information collected from the corpus (collocation & co-occurrence).
3. A measure of association between two words.
4. An algorithm for generating candidate associates using the resources above.
5. An algorithm for scoring candidate associates.

2.1 Corpus

Our corpus is composed of two sources. One part is the English Gigaword 2003 corpus (Graff and Cieri, 2003), with 1.7 billion tokens. The second part is an ETS in-house corpus containing texts from the genres of fiction and popular science (Sheehan et al., 2006), with about 430 million tokens.

2.2 Types of distributional information

From this combined corpus we have built two specific lexical resources. One resource is a bigram repository, which stores counts for sequences of two words. The other resource is a first-order co-occurrence word-space model (Turney and Pantel, 2010), also known as a Distributional Semantic Model (**DSM**) (Baroni and Lenci, 2010). In our implementation of DSM, we counted non-directed co-occurrence of tokens in a paragraph, using no distance coefficients (Bullinaria and Levy, 2007). Counts for 2.1 million word-form types, and the sparse matrix of their co-occurrences, are efficiently compressed using the TrendStream toolkit (Flor, 2013), resulting in a database file of 4.7GB.

The same toolkit supports both n-grams and DSM repositories, and allows fast retrieval of word probabilities and statistical associations for pairs of words.[1] It also supports retrieval of co-occurrence vectors. When generating these two resources, we used no lemmatization and no stoplist. All tokens were converted to lowercase. All punctuation was retained and counted as tokens. The only significant filtering was applied to numbers: all digit-based numbers (e.g. 5, 2.1) were converted to the symbol '#' and counted as such. Tokenization was performed by an internal module of the TrendStream toolkit.

The lexical resources described above were not generated for the COGALEX-4 shared task. Rather, those are general-purpose large-scale lexical resources that we have used in previous research, for a variety of NLP tasks. This is an important aspect, as our intention was to find out how well those general resources would perform on this novel task. Our bigrams repository is actually part of a 5-gram language model that is used for context-aware spelling correction. The algorithms for that application are described by Flor (2012). The DSM has been used for spelling correction (Flor, 2012), for essay scoring (Beigman Klebanov and Flor, 2013a), for readability estimation (Flor and Beigman Klebanov, in press; Flor et al., 2013), as well as for a study on quality of machine translation (Beigman Klebanov and Flor, 2013b).

2.3 Measures of association

For the shared task, we used three measures of word association.

Pointwise Mutual Information (Church & Hanks, 1990):

$$PMI(a,b) = \log_2 \frac{P(a,b)}{P(a)P(b)}$$

Normalized Pointwise Mutual Information (Bouma, 2009):

$$NPMI(a,b) = \left(\log_2 \frac{P(a,b)}{P(a)P(b)}\right) / \left(-\log_2 P(a,b)\right)$$

[1] The TrendStream toolkit provides compression and storage for large-scale n-gram models, and for large-scale co-occurrence matrices. In all cases, actual counts are stored and values for statistical association measures are computed on the fly during data retrieval.

Simplified log-Likelihood (Evert, 2008):

$$SLL(a,b) = 2 \cdot P(a,b) \cdot \log \frac{P(a,b)}{P(a)P(b)} - P(a,b) + P(a)P(b)$$

P(a,b) signifies probability of joint co-occurrence. For bigrams, that is joint co-occurrence in a specific sequential order (e.g. AB vs. BA) ; for DSM data the co-occurrence is order-independent.

2.4 Procedure for generating candidate multi-cue associates

Our general procedure for generating target candidates is as follows. For each of the five cue words, candidate targets are generated separately, from the corpus-based resources:

1. From the DSM (generally associated words)
2. Left words from bigrams (words that, in the corpus, appeared immediately to the left of the cue)
3. Right words from bigrams (words that appeared immediately to the right of the cue)

Retrieved lists of candidates can be quite large, with hundreds and even thousands of different neighbors. One specific filter implemented at this stage was that only word-forms (alphabetic strings) were allowed, and any punctuation or '#' strings were filtered out.

Since our resources are not lemmatized, we extended the candidate retrieval procedure by expanding the cue words to their inflectional variants. This provides richer information about semantic association. We used an in-house morphological analyzer/generator. Inflectional expansions were not constrained for part of speech or word sense. For example, given the cue set {*1:letters 2:meaning 3:sentences 4:book 5:speech*} (from the training set of the shared task, target: *'words'*), after expansion the set of cues is {*1:letters, lettered, letter, lettering 2:meaning, means, mean, meant, meanings 3:sentences, sentence, sentenced, sentencing 4:book, books, booking, booked 5:speech, speeches*}. The vector of right neighbors for the cue '*letters*', brings such words as {*sent, from, between, written, came, addressed, ...*}. The vector of left neighbors for same cue word brings such candidates as {*write, send, love, capital, review, ...*}. From the DSM, the vector of co-occurrence may bring some of the same words (but with different values of association), as well as words that do not generally occur immediately before or after the cue word, e.g. {*time, people, word, now,...*}.

Next, we apply filtering that ensures the minimal requirement for multi-word association – a candidate must be related to all cues. The candidate must appear (at least once) on the list of words generated from each cue family. A candidate word that does not meet this requirement is filtered out.[2]

2.5 Scoring of candidate associates

Scoring of candidate associate-words is a two-stage process. First, for each candidate, we look for the strongest association value it has with each of the five cue families. Then, the five strongest values are combined into an aggregated score.

For a given cue family, several instances of the same candidate associate might be retrieved, with various values of association score (from DSM and n-grams, and also for each specific inflectional form of the cue). We pick the highest score, siding with the source that provides the strongest evidence of connection between the cue and the candidate associate. The maximal association value is stored as the best score for this candidate with the given cue family. We note that since the same measure of association is used, the scores from the different sources are numerically comparable.[3] For example, when PMI is used as the association measure, the following values were obtained for candidate '*capital*' with cue family '*letters, lettered, letter, lettering*' (expanded from '*letters*'). General co-occurrence (DSM): *capital & letters*: 0.477, *capital & letter*: 0.074, etc.; left bigrams: *capital letters*: 5.268, *capital letter*: 2.474, etc. The strongest association here is the bigram '*capital letters*', and the value 5.268 is the best association of the candidate '*capital*' with this cue family.

Next, for each candidate we compute an aggregate score that represents its overall association with all five cues. In current study, we experimented with two forms of aggregation: 1) sum of best scores

[2] This is 'baseline' filtering, applied in all experiments. Experiments with additional filtering are described in section 4.2.

[3] In any single experimental run we consistently use the same measure of association (no mixing of different formulae).

(SBS), and 2) product (multiplication) of ranks (MR). Sum of best scores is simply the sum of best association scores that a candidate has with each of the five cues (families). To produce a final ranked list of candidate targets, candidates are sorted by their aggregate sum value (better candidates have higher values). Multiplication of ranks has been proposed as an aggregation procedure by Rapp (2014, 2008). In this procedure, all candidates are sorted by their association scores with each of the five cues (families) separately, and five rank values are registered for each candidate. The five rank values are then multiplied to produce the final aggregate score. All candidates are then sorted by the aggregate score, and in such ranking better candidates have lower aggregate scores. Multiplication of ranks is computationally more intensive than sum of scores – for a given set of candidate words from five cues, multiplication of ranks requires six calls for sorting, while aggregation via sum-of-best-scores performs sorting only once.

Finally, all candidates are sorted by their aggregate score and top N are outputted for the calculation of *precision@N*, to be described below.

3 Results

Our system ran with several different configuration settings, using various association measures and score aggregation procedures. Under any given configuration, the system produces, for each item (i.e. a set of five cue words), a ranked list of candidates. According to the rules of the shared task, official results are computed by selecting the single best candidate for the item as the suggested target word. If the suggested word strictly matches the gold-standard word (ignoring upper/lower case), it is considered a match. If the two strings differ even slightly, it is considered a mismatch. The reported result is precision (percent matches) over the test set of 2000 items.

With strict-matching, our best result for the test-set was precision of 18.6% (372 correctly suggested targets). This was obtained by using NPMI as the association measure, product of ranks as the score aggregation procedure, and with filtering of candidates using a stoplist and a frequency filter.[4]

The shared task was described as multi-cue association for finding a sought-after 'missing' word, a situation not unlike a tip-of-the-tongue phenomenon. In such situation, a person looking for an associated word, might find it useful if the system returns not just one highest-ranked suggestion (which would often be a miss), but a list of several top-ranked suggestions – the target word might be somewhere on such list[5]. Thus, we also present our results in terms of precision for *n*-best suggestions – i.e. in how many cases the target word was among the top *n* returned by the system, with *n* ranging from 1 up to 25.

A similar consideration applies to inflectional variants. A person looking for a word associated with a set of cue words, might be satisfied when a system returns either a base-form or an inflected variant of the target word. Thus, we report our results both in terms of strict matches to gold-standard targets and under a condition of 'inflections-allowed'.[6] On the test set, our best result for precision@1, with inflections allowed, is 24.35% (487 matching suggestions).

First, we present our baseline results. Figure 1 presents the results of our system for the training set of 2000 items, using the NPMI association measure. Panel 1A presents data obtained using aggregation via sum-of-best-scores (SBS). Panel 1B presents data obtained using aggregation via multiplication of ranks (MR). Figure 2 presents similar breakdown for results of the test set. Both sets of results are quite similar. Thus, we restrict our attention to just the results of the test set. [7]

[4] We initially submitted a result of 14.95% strict-match precision@1 (see Figure 2A). This was improved to 16.1% (Figure 2B), and with additional filters – to 18.6% (see section 4.2).

[5] A list of *n*-best suggestions is standard approach for presenting candidate corrections for misspellings (Flor, 2013; Mitton, 2008). Also, precision "at *n* documents" is a well known evaluation approach in information retrieval (Manning et al., 2008). A recent use of *n*-best suggestions in an interactive NLP system is illustrated by Madnani and Cahill (2014).

[6] Each target word form, both in the training set and the test set, was automatically expanded to all its inflectional variants, using our morphological analyzer/generator. In our evaluations, a candidate target is considered a 'hit' if it matches the gold-standard target or one of its inflectional variants.

[7] We did not use the training set for any training or parameter tuning. We used it to select the optimal association measures for this task – we also experimented with t-score, weighted PMI and conditional probability, but PMI and NPMI performed much better than others.

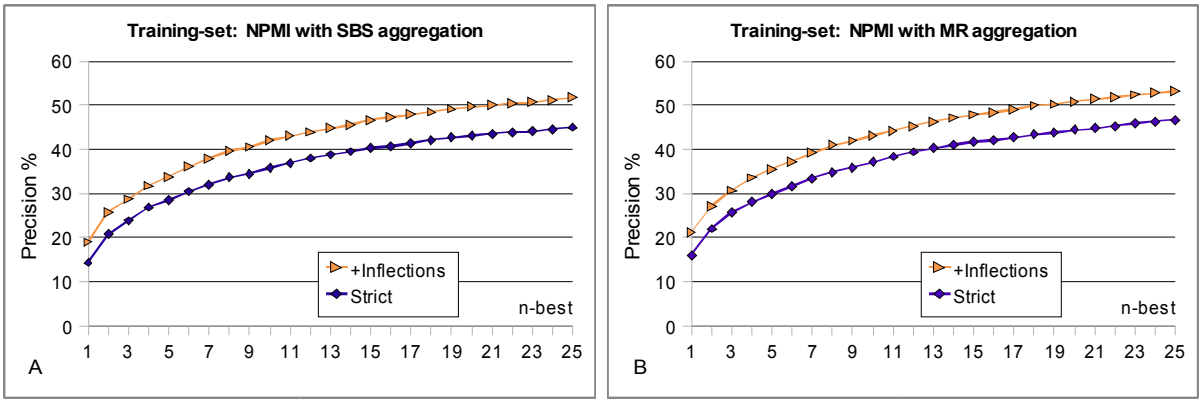


Figure 1. System performance on the training-set (percent correct out of 2000 items), for various values of *n*. Panel A: using sum-of-best-scores aggregation; Panel B: using multiplication-of-ranks aggregation. 'Strict': evaluation uses strict matching to gold-standard target, '+Inflections': inflectional variants are allowed in matching to gold-standard target.

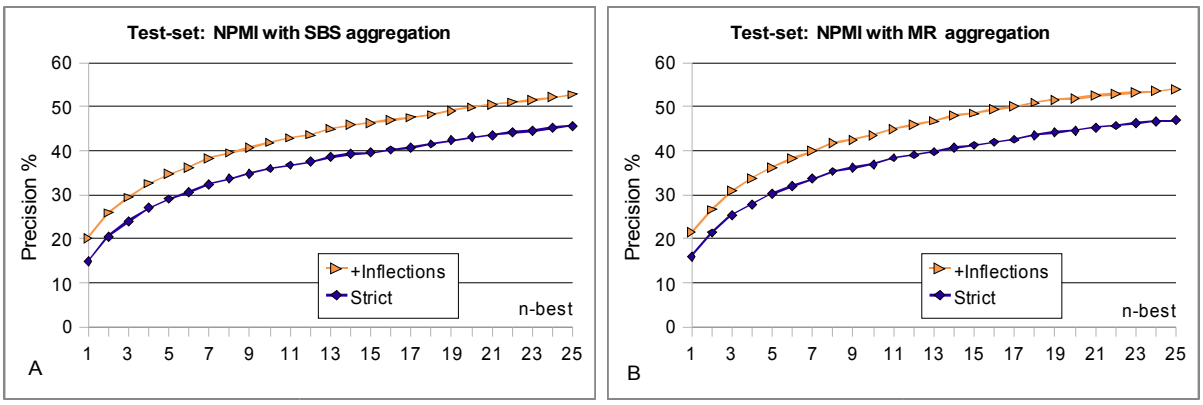


Figure 2. System performance on the test-set (percent correct out of 2000 items).

We found, as expected, that performance improves when the target is sought among the *n*-best candidates produced by the system. With NPMI and MR aggregation, strict-match precision improves from 16.1% for precision@1 to 30.3% for precision@5, 37% for precision@10, and 46.9% for precision@25 (Figure 2B).

Another expected result is that performance is better when matching of targets allows inflectional variants. This is clearly seen on the charts, as the difference between the two lines. With NPMI and MR aggregation, precision@1 improves from 16.1% to 21.45%, precision@5 improves from 30.3% to 36.3%, and precision@25 improves from 46.9% to 54%, Similar improvement is observed when using aggregation via sum-of-best-scores.

Our third finding is that multiplication of ranks achieves slightly better results than sum-of-best-scores (Figure 2, panel B vs. panel A). For precision@1 with strict matches, using NPMI, MR achieves 16.1% and with inflectional variants 21.45%, while SBS achieves 14.95% and 20.25% respectively. For precision@10, MR achieves 37% (43.55%), while SBS achieves 36% (42%). Notably, MR is consistently superior to SBS for all values of *n*-best, from 1 to 25, under both strict or inflections-allowed matching, with both NPMI and PMI (see Figure 3). However, the advantage is consistently rather small – about 1-1.5%. Since MR is computationally more intensive, SBS emerges as a viable alternative.

We have also conducted experiments with three different measures of association. Results are presented in Figure 3. With MR aggregation, NPMI achieves better results than the PMI measure. Both measures clearly outperform the Simplified log-Likelihood. Similar results are obtained with SBS aggregation. For each association measure, allowing inflections provides better results than strict matching to gold-standard targets.

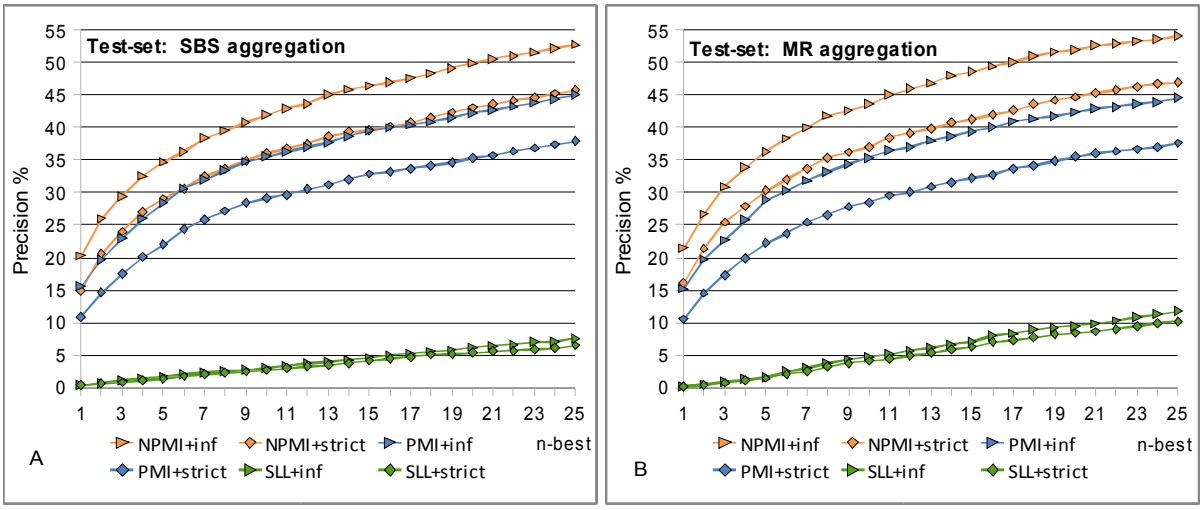


Figure 3. System performance on the test-set (2000 items) with three different association measures. Panel A: using sum-of-best-scores aggregation; Panel B: using multiplication-of-ranks aggregation. Legend: PMI: pointwise mutual information, NPMI: Normalized PMI, SLL: simplified log-likelihood, 'Strict': evaluation uses strict matching to gold-standard target, '+Inf': inflectional variants are allowed in matching to gold-standard target.

4 Additional studies

In several additional experiments we looked at the contribution of different factors to overall performance. We tried several variations of resource combination and also tested filtering of candidates by frequency and by using a list of stopwords.

4.1 Ablation experiments

We investigated how the restriction of resources impacts the performance on this task. Specifically we restricted the resources as follows. In one condition we used only the bigrams data, retrieving candidates only from the vectors of left co-occurring words (immediate preceding words) of each cue word (condition **NL** – n-grams left). A similar restriction is when candidates are retrieved only from right (immediate successor) words (condition **NR** – n-grams right). A third condition still uses only bigrams, but admits candidates from both left and right vectors (condition NL+NR). Under the fourth condition (DSM), n-grams data is not used at all, only the DSM resource is used. In the fifth and sixth conditions we combine candidates from DSM with n-gram candidates (left or right vectors only – respectively). The seventh condition is our standard – candidates from DSM and both left and right neighbors from bigrams are admitted. For those experiments, we used NPMI association measure with MR aggregation, and included inflections in evaluation. The results are presented in Figure 4.

Using only right-hand associates (typical textual successors of cue words) provides very low performance (precision@1 is 2.95%). Using only left-hand associates (typical textual predecessors of cue words) provides slightly better performance (precision@1 is 4.5%). However, it is notable that there are some items in the EAT data where all cues are strong bigrams with the target, e.g. {*orange, fruit, lemon, apple, tomato*} with target '*juice*'. Combining these two resources (condition NL+NR) provides much better performance: precision@1 is 8.5%. Using just the DSM, the system achieves 10.5% precision@1, which may seem rather close to the combined NL+NR 8.5%. However, with DSM, for *n*-best lists precision rises quite sharply (e.g. 24.35% for precision@5), while for the NL+NR setting precision tends to be under 17% for all values of *n* up to 25.

Since our DSM and bigrams resources are built on the same corpus of text, for any given set of cues the DSM produces all the candidates that the bigrams resource does (but with different association values) and a lot of other candidates. However, results for DSM+NR and DSM+NL settings (which are better than DSM alone) indicate that association values from bigrams contribute substantially to overall performance. The best result in this experiment is achieved by a setting that combines candidates (and association values) from all three resources, indicating further that associations from sequential word combinations (bigrams) provide a substantial contribution to performance in this task.

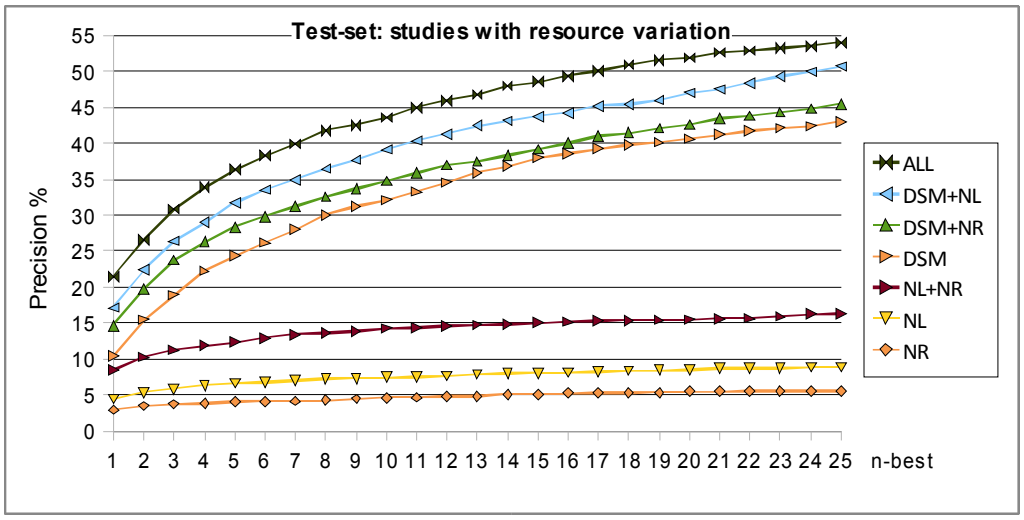


Figure 4. System performance on the test-set (2000 items), with various resource restrictions. All runs used NPMI association measure and MR aggregation. Evaluation allowed inflections. NL/NR – left/right neighbors from bigrams.

4.2 Applying filters on retrieved candidates

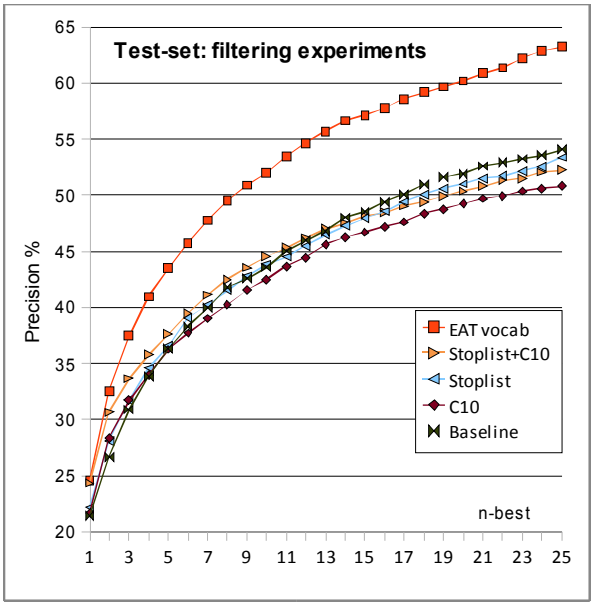
We also experimented with applying some filters on the retrieved candidates for each item. One of the obvious filters to use is to filter out stopwords. For general tip-of-the-tongue search cases, common stopwords are rarely useful as target words; thus presenting stopwords as candidates makes little sense. We used a list of 87 very common English stopwords, including the articles {*the, a, an*}, common prepositions, pronouns, wh-question words, etc. However, since the data of the shared task comes from EAT, common stopwords are actually targets in some cases in that collection. Therefore, we used the following strategy. For a given item, if at least one of the five cue words is a stopword, then we assume that the target might also be a stopword, and so we do not use the stoplist to filter candidates for this item. However, if none of the cues is a stopword, we do apply filtering – any retrieved candidate word is filtered out if it is on the stoplist. An additional filter, applied with the stoplist, was defined as follows: if a candidate word is strictly identical to one of the cue words, the candidate is filtered out (to allow for potentially more suitable candidates).[8]

The other filter considers frequency of words. The PMI measure is known to overestimate the strength of pair association when one of the words is a low-frequency word (Manning & Schütze, 1999). Normalized PMI is also sensitive to this aspect, although less than PMI. Thus, we use a frequency filter to drop some candidate words. For technical reasons, it was easier for us to apply a cutoff on the joint frequency of a candidate and a cue word. We used a cutoff value of 10 – a candidate is dropped if corpus data indicates it co-occurs with the cue words fewer than 10 times in the corpus data.

We applied the stoplist filter, the frequency filter and a combination of those two filters, always using NPMI as our association measure, aggregating scores via multiplication-of-ranks, and allowing inflections in evaluation. No ablation of resources was applied. The results are presented in Figure 5. The baseline condition is when neither of the two filters is applied. The frequency filter with cutoff=10 provides a very small improvement for precision@1, and for higher values of best-*n* it actually hurts performance. Application of a stoplist provides a very slight improvement of performance. The combination of a stoplist and frequency cutoff=10 provides a sizable improvement of performance (precision@1 is 24.35% vs. baseline 21.45%, and precision@10 is 44.55% vs. baseline 43.55%). However, for *n*-best lists of size 15 and above, performance without filters is slightly better than with those filters. For the shared task (using strict matching – no inflections), our best result is 18.6% precision@1 with two filters (16.1% without filters).

[8] Cases when a candidate word is identical to one of the cues do occur when associate candidates are harvested from corpus data. Such candidates have little utility for a missing-word-search task. Notably, however, the training-set for the shared task did have one item where the target word was identical to one of the cues: *Yeah ~ Yeah no Yes Beatles Oh*.

Given that the gold-standard targets in the shared task are original stimulus words form the EAT collection, we can use a special restriction – restrict the candidates to just the EAT stimuli word-list (Rapp, 2014). Notably, this is a very specific restriction, suited to the specific dataset, and not applicable to the general case of multi-cue associations or tip-of-the-tongue word searches. We used the list of 7913 single-word stimuli from EAT as a filter in our system – generated candidates that were not on this list were dropped from consideration. The results (Figure 5) indicate that this restriction (EATvocab) provides a substantial improvement over the baseline condition. For precison@1, using EATvocab (24.55%) is comparable to using a stoplist+cutoff10 (24.35%). However, for larger n-best lists, EATvocab filter provides substantially better performance.



Condition	Precision@1	Precision@10
EAT Vocabulary	24.55%	52.00%
Stoplist & Cutoff10	24.35%	44.55%
Stoplist	22.15%	43.85%
Cutoff10	21.70%	42.50%
Baseline (no filters)	21.45%	43.55%

Figure 5. System performance on the test set with different filtering conditions. All runs use NPMI association and MR aggregation. Inflections allowed in evaluation. C10: frequency cutoff=10.

5 Small-scale evaluation using direct human judgments

Inspecting results from training-set data, we observed a number of cases where the system produced very plausible targets which however were struck down as incorrect (not matching the gold-standard). For example, for the cue set {*music, piano, play, player, instrument*} the gold-standard target was '*accordion*'. But why not '*violin*' or '*trombone*'? To provide a more in-depth evaluation of the results, we sampled 180 items at random from the test set, along with the candidate targets produced by our system,[9] and submitted those to evaluation by two research assistants. For each item, evaluators were given the five cue words and the best candidate target generated by the system. They were told that the word is supposed to be a common associate of the five cues, and asked to indicate, for each item, whether the candidate was (a) *Just Right*; or (b) *OK*; or (c) *Inadequate;* (a,b,c are on ordinal scale).

Out of the 180 items, 80 were judged by both annotators. Table 1 presents the agreement matrix between the two annotators. Agreement on the 3 classes was kappa=0.49. If *Just Right* and *OK* are collapsed, the agreement is kappa=0.60. The discrepancy is largely due to a substantial number of instances that one annotator judged *OK* and the other – *Just Right*.

	Inadequate	OK	Just Right	TOTAL
Inadequate	17	6	1	24
OK	6	25	10	41
Just Right	0	3	12	15
TOTAL	23	34	23	80

Table 1. Inter-annotator agreement matrix for a subset of items from the test-set.

[9] Using all resources, NPMI association measure, MR aggregation, and with the general stoplist filter.

We note that one annotator commented on a difficulty making a decision in a number of cases where the cues are a list of mostly adjectives or possessives, and the target produced by the system is an adverb. For example, the cue set {*busy, house, vacant, engaged, empty*} with the proposed candidate target '*currently*'; the cue set {*food, thirsty, tired, empty, starving*} with the proposed candidate '*perpetually*'; the cue set {*fat, short, build, thick, built*} with the proposed candidate '*slightly*'; the cue set {*mine, yours, his, is, theirs*} with the proposed target '*rightfully*'. This annotator felt that these responses were *OK*, while the other annotator rejected them.

We merged the two annotations to provide a single annotation for the full set of 180 items by taking one annotator's judgment on single-annotated cases and taking the lower of the two judgments for the double annotated disagreed cases (thus, *OK* and *Inadequate* are merged to *Inadequate*; *Just Right* and *OK* are merged to *OK*). We next compare these annotations to the EAT gold standard. Table 2 shows the confusion matrix between the "gold label" from EAT and our annotation. We observe that the totals for *Just Right* and EAT-match are almost identical (43 vs 42); however, only 17 items were both *Just Right* and EAT-matches. There were 24 EAT matches that were judged as *OK* by the annotators (presumably, these did not quite create the "just right" impression for at least one annotator). Examples include: the cue set {*beer, tea, storm, ale, bear*} with the proposed correct target '*brewing*' (one annotator commented that the relationship with "*bear*" was unclear); the cue set {*exam, match, tube, try, cricket*} with the proposed correct target '*test*' (one annotator commented that the relationship with '*cricket*' was unclear); the cue set {*school, secondary, first, education, alcohol*} with the proposed correct target '*primary*' (one annotator commented that the relationship with '*alcohol*' was unclear). These results might reflect cultural differences between original EAT respondents (British undergraduates circa year 1970) and present-day American young adults who, e.g. might not know much about cricket. Another possibility is that in the EAT collection, the 5th cue sometimes corresponds to a very weak associate provided by just a single respondent out of 100, as in *brewing-bear* and *primary-alcohol* cases. Interestingly, the weak cues did not confuse the system, but replicability of the human judgments for such cases is doubtful.

	Just Right	**OK**	**Inadequate**	**Total**
EAT match	17	24	1	**42**
EAT mismatch	26	58	54	**138**
Total	**43**	**82**	**55**	**180**

Table 2. Annotated data vs. gold-standard matches for a set of 180 items.

There were also 26 instances that were judged as *Just Right* yet were not EAT-matches. Three of these were derivationally related, like '*build*' (EAT target) vs '*buildings*' (proposed) for the cue set {*house, up, construct, destroy, bricks*}, the others were '*dwell*' vs '*dwellings*', '*collector*' vs '*collecting*'. In the rest of the cases, the generated candidates seemed as good as, or better, than the EAT words. For example, the cue set {*ships, boat, sea, ship, ocean*} had '*liners*' as the EAT target, whereas the system proposed '*cruise*'. For the cue set {*natural, animal, nature, birds, fear*}, the gold-standard EAT target is '*instinct*', whereas the system proposed '*predatory*'. For the cue set {*sound, speak, sing, noise, speech*} the gold-standard EAT target is '*voice*', while the system produced '*louder*'. For the cue set {*music, band, noise, club, folk*} the target was '*jazz*', whereas the system proposed '*dance*'. For the cue set {*violin, music, orchestra, bow, instrument*} the target was '*cello*', while the system produced '*stringed*'. Furthermore, in as many as 58 cases (32%) the response produced by the system did not match the target from EAT, but was OK-ed by the annotators. Some examples include: the cue set {*fool, loaf, idiot, lout, lazy*} with proposed candidate '*ignorant*'; the cue set {*hard, problems, work, hardship, trouble*} with proposed candidate '*economic*'; {*interesting, intriguing, amazing, book, exciting*} with proposed candidate '*discoveries*'; {*lazy, chair, about, lying, sitting*} with proposed candidate '*motionless*'. In all, if the system were evaluated by counting *Just Right* and *OK* annotations as correct, the precison@1 would have been (43+82)/180 = 69%. The estimation of performance based on gold-standard EAT data for this set is 42/180 = 23%, exactly one-third of what annotators found to be reasonable responses. This suggests that evaluation of multi-cued retrieval on targets from EAT rejects many good semantic associates, and thus might be considered too harsh.

6 Conclusions

This paper presented an automated system that computes multi-cue associations and generates associated-word suggestions, using lexical co-occurrence data from a large corpus of English texts. The system uses pre-existing resources – a large *n*-ngram database and a large word-co-occurrence database, which have been previously used for a range of different NLP tasks. The system performs expansion of cue words to their inflectional variants, retrieves candidate words from corpus data, finds maximal associations between candidates and cues, and then computes an aggregate score for each candidate. The collection of candidates is then sorted and an *n*-best list is presented as output. In the paper we presented experiments using various measures of statistical association and two methods of score aggregation. We also experimented with limiting the lexical resources, and with applying additional filters on retrieved candidates.

For test-set evaluation, the shared task requires strict-matches to gold-standard targets. Our system, in optimal configuration, was correct in 372 of 2000 cases, that is precision of 18.6%. We have also suggested a more lenient evaluation, where a candidate target is also considered correct if it is an inflectional variant of the gold-standard word. When inflections are allowed, our system achieves precision of 24.35%. Performance improves dramatically when evaluation considers in how many cases the gold-standard target (or its inflectional variants) are found among the *n*-best suggestions provided by the system. For example, with a list of 10-best suggestions, precision rises to 45%, and to 54% with a list of 25-best. Using an *n*-best list of suggestions makes sense for applications like tip-of-the-tongue situation.

We note that the specific data set used in COGALEX-4 shared task, i.e. the Edinburgh Associative Thesaurus, might be sub-optimal for evaluation of multi-cue associative search. With the EAT dataset, the gold-standard words were the original stimuli from EAT, and the cue words were the associated words that were most frequently produced by respondents in the original EAT experiment (Kiss et al., 1973). Rapp (2014) has argued that corpus-based computation of reverse-associations is a reasonable test case for multi-cued word search. However, Rapp also notes that in many cases, suggestions provided by a corpus-based system are quite reasonable, but are not correct for the EAT dataset. We have conducted pilot human annotation on a small subset of the test-set – judging how reasonable the top suggestion of our system is in general, and not whether it matched EAT targets. In this experiment, 69% of the system's first responses were judged acceptable by humans, while only 23% matched targets. This provides a quantitative confirmation that EAT-based evaluation underestimates the quality of results produced by a corpus-based multi-cue association system.

The use of data from EAT hints at the following direction for future research. In the original EAT data, the first cue is actually the strongest associate of the target word (original stimulus), while other cues are much weaker associates. In our current implementation, we treated all cues as equally important. Future research may include consideration for relative importance or relevance of the different cues. In potential applications, like the tip-of-the-tongue word search, a user may be able to specify which cues are more relevant than others.

Acknowledgments

Special thanks to Melissa Lopez and Matthew Mulholland for their help with the evaluation study. We also thank Mo Zhang, Paul Deane and Keelan Evanini at ETS, and three anonymous COGALEX reviewers, for comments on earlier drafts of this paper.

References

Marko Baroni and Allesandro Lenci. 2010. Distributional Memory: A General Framework for Corpus-Based Semantics. *Computational Linguistics*, 36(4), 673-721

Beata Beigman Klebanov and Michael Flor. 2013a. Word Association Profiles and their Use for Automated Scoring of Essays. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 1148–1158, Sofia, Bulgaria.

Beata Beigman Klebanov and Michael Flor. 2013b. Associative Texture Is Lost In Translation. In Proceedings of the Workshop on Discourse in Machine Translation (DiscoMT), pages 27–32. ACL 2013 Conference, Sofia, Bulgaria.

Gerlof Bouma. 2009. Normalized (Pointwise) Mutual Information in Collocation Extraction. In: Chiarcos, Eckart de Castilho & Stede (eds), From Form to Meaning: Processing Texts Automatically, Proceedings of the Biennial GSCL Conference 2009, Tübingen, Gunter Narr Verlag, p. 31–40.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39:510–526.

Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information and lexicography. *Computational Linguistics*, 16(1), 22–29.

David Graff and Christopher Cieri. 2003. English Gigaword. LDC2003T05. Philadelphia, PA, USA: Linguistic Data Consortium.

Stefan Evert. 2008. Corpora and collocations. In A. Lüdeling and M. Kytö (eds.), Corpus Linguistics: An International Handbook, Mouton de Gruyter: Berlin.

Michael Flor. 2013. A fast and flexible architecture for very large word n-gram datasets. *Natural Language Engineering*, 19(1), 61-93.

Michael Flor. 2012. Four types of context for automatic spelling correction. *Traitement Automatique des Langues* (TAL), 53:3 (Special Issue: Managing noise in the signal: error handling in natural language processing), 61-99.

Michael Flor and Beata Beigman Klebanov. (in press) Associative Lexical Cohesion as a factor in Text Complexity. Accepted for publication in the International Journal of Applied Linguistics.

Michael Flor, Beata Beigman Klebanov and Kathleen M. Sheehan. 2013. Lexical Tightness and Text Complexity. In Proceedings of the 2th Workshop of Natural Language Processing for Improving Textual Accessibility (NLP4ITA), p.29–38. NAACL 2013 Conference, Atlanta, Georgia.

G.R. Kiss, C. Armstrong, R. Milroy and J. Piper. 1973. An associative thesaurus of English and its computer analysis. In Aitken, A.J., Bailey, R.W. and Hamilton-Smith, N. (Eds.), The Computer and Literary Studies. Edinburgh: University Press.

Nitin Madnani and Aoife Cahill. 2014. An Explicit Feedback System for Preposition Errors based on Wikipedia Revisions. To appear in Proceedings of the 9th Workshop on Innovative Use of NLP for Building Educational applications (BEA-9). ACL 2014 Conference, Baltimore, MD.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Christopher D. Manning, and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*, 1999, Cambridge, Massachusetts, USA: MIT Press.

Roger Mitton. 2008. Ordering the suggestions of a spellchecker without using context. *Natural Language Engineering*, 15(2), 173–192.

Reinhard Rapp. 2014. Corpus-Based Computation of Reverse-Associations. Proceedings of LREC.

Reinhard Rapp. 2008. The computation of associative responses to multiword stimuli. In Proceedings of the Workshop on Cognitive Aspects of the Lexicon (COGALEX) at COLING-2008, p.102–109. Manchester, UK

Kathleen M. Sheehan, Irene Kostin, Yoko Futagi, Ramin Hemat and Daniel Zuckerman. 2006. Inside SourceFinder: Predicting the Acceptability Status of Candidate Reading-Comprehension Source Documents. ETS research report RR-06-24. Educational Testing Service: Princeton, NJ.

Peter Turney and Patrick Pantel. 2010. From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research*, 37, 141-188.