Event-based Hyperspace Analogue to Language for Query Expansion

Tingxu Yan
Tianjin University
Tianjin, China
sunriser2008@gmail.com

Tamsin Maxwell
University of Edinburgh
Edinburgh, United Kingdom
t.maxwell@ed.ac.uk

Dawei Song
Robert Gordon University
Aberdeen, United Kingdom
d.song@rgu.ac.uk

Yuexian Hou
Tianjin University
Tianjin, China
yxhou@tju.edu.cn

Peng Zhang
Robert Gordon University
Aberdeen, United Kingdom.
p.zhang1@rgu.ac.uk

Abstract

Bag-of-words approaches to information retrieval (IR) are effective but assume independence between words. The Hyperspace Analogue to Language (HAL) is a cognitively motivated and validated semantic space model that captures statistical dependencies between words by considering their co-occurrences in a surrounding window of text. HAL has been successfully applied to query expansion in IR, but has several limitations, including high processing cost and use of distributional statistics that do not exploit syntax. In this paper, we pursue two methods for incorporating syntactic-semantic information from textual 'events' into HAL. We build the HAL space directly from events to investigate whether processing costs can be reduced through more careful definition of word co-occurrence, and improve the quality of the pseudo-relevance feedback by applying event information as a constraint during HAL construction. Both methods significantly improve performance results in comparison with original HAL, and interpolation of HAL and relevance model expansion outperforms either method alone.

1 Introduction

Despite its intuitive appeal, the incorporation of linguistic and semantic word dependencies in IR has not been shown to significantly improve over a bigram language modeling approach (Song and Croft, 1999) that encodes word dependencies assumed from mere syntactic adjacency. Both the dependence language model for IR (Gao et al., 2004), which incorporates linguistic relations between non-adjacent words while limiting the generation of meaningless phrases, and the Markov Random Field (MRF) model, which captures short and long range term dependencies (Metzler and Croft, 2005; Metzler and Croft, 2007), consistently outperform a unigram language modelling approach but are closely approximated by a bigram language model that uses no linguistic knowledge. Improving retrieval performance through application of semantic and syntactic information beyond proximity and co-occurrence features is a difficult task but remains a tantalising prospect.

Our approach is like that of Gao et al. (2004) in that it considers semantic-syntactically determined relationships between words at the sentence level, but allows words to have more than one role, such as predicate and argument for different events, while link grammar (Sleator and Temperley, 1991) dictates that a word can only satisfy one connector in a disjunctive set. Compared to the MRF model, our approach is unsupervised where MRFs require the training of parameters using relevance judgments that are often unavailable in practical conditions.

Other work incorporating syntactic and linguistic information into IR includes early research by (Smeaton, O'Donnell and Kelledy, 1995), who employed tree structured analytics (TSAs) resembling dependency trees, the use of syntax to detect paraphrases for question answering (QA) (Lin and Pantel, 2001), and semantic role labelling in QA (Shen and Lapata, 2007).

Independent from IR, Pado and Lapata (2007) proposed a general framework for the construction of a semantic space endowed with syntactic

Proceedings of the ACL 2010 Conference Short Papers, pages 120–125,
Uppsala, Sweden, 11-16 July 2010. ©2010 Association for Computational Linguistics

information. This was represented by an undirected graph, where nodes stood for words, dependency edges stood for syntactical relations, and sequences of dependency edges formed paths that were weighted for each target word. Our work is in line with Pado and Lapata (2007) in constructing a semantic space with syntactic information, but builds our space from events, states and attributions as defined linguistically by Bach (1986). We call these simply *events*, and extract them automatically from predicate-argument structures and a dependency parse. We will use this space to perform query expansion in IR, a task that aims to find additional words related to original query terms, such that an expanded query including these words better expresses the information need. To our knowledge, the notion of events has not been applied to query expansion before.

This paper will outline the original HAL algorithm which serves as our baseline, and the event extraction process. We then propose two methods to arm HAL with event information: direct construction of HAL from events (eHAL-1), and treating events as constraints on HAL construction from the corpus (eHAL-2). Evaluation will compare results using original HAL, eHAL-1 and eHAL-2 with a widely used unigram language model (LM) for IR and a state of the art query expansion method, namely the Relevance Model (RM) (Lavrenko and Croft, 2001). We also explore whether a complementary effect can be achieved by combining HAL-based dependency modelling with the unigram-based RM.

2 HAL Construction

Semantic space models aim to capture the meanings of words using co-occurrence information in a text corpus. Two examples are the Hyperspace Analogue to Language (HAL) (Lund and Burgess, 1996), in which a word is represented by a vector of other words co-occurring with it in a sliding window, and Latent Semantic Analysis (LSA) (Deerwester, Dumais, Furnas, Landauer and Harshman, 1990; Landauer, Foltz and Laham, 1998), in which a word is expressed as a vector of documents (or any other syntactical units such as sentences) containing the word. In these semantic spaces, vector-based representations facilitate measurement of similarities between words. Semantic space models have been validated through various studies and demonstrate compatibility with human information processing. Recently, they have also been applied in IR, such as LSA for latent semantic indexing, and HAL for query expansion. For the purpose of this paper, we focus on HAL, which encodes word co-occurrence information explicitly and thus can be applied to query expansion in a straightforward way.

HAL is premised on context surrounding a word providing important information about its meaning (Harris, 1968). To be specific, an L-size sliding window moves across a large text corpus word-by-word. Any two words in the same window are treated as co-occurring with each other with a weight that is inversely proportional to their separation distance in the text. By accumulating co-occurrence information over a corpus, a word-by-word matrix is constructed, a simple illustration of which is given in Table 1. A single word is represented by a row vector and a column vector that capture the information before and after the word, respectively. In some applications, direction sensitivity is ignored to obtain a single vector representation of a word by adding corresponding row and column vectors (Bai et al., 2005).

	w_1	w_2	w_3	w_4	w_5	w_6
w_1						
w_2	5					
w_3	4	5				
w_4	3	4	5			
w_5	2	3	4	5		
w_6		2	3	4	5	

Table 1: A HAL space for the text "w_1 w_2 w_3 w_4 w_5 w_6" using a 5-word sliding window ($L = 5$).

HAL has been successfully applied to query expansion and can be incorporated into this task directly (Bai et al., 2005) or indirectly, as with the Information Flow method based on HAL (Bruza and Song, 2002). However, to date it has used only statistical information from co-occurrence patterns. We extend HAL to incorporate syntactic-semantic information.

3 Event Extraction

Prior to event extraction, predicates, arguments, part of speech (POS) information and syntactic dependencies are annotated using the best-performing joint syntactic-semantic parser from the CoNNL 2008 Shared Task (Johansson and

Nugues, 2008), trained on PropBank and NomBank data. The event extraction algorithm then instantiates the template *REL [modREL] Arg0 [modArg0] ...ArgN [modArgN]*, where *REL* is the predicate relation (or root verb if no predicates are identified), and *Arg0...ArgN* are its arguments. Modifiers (*mod*) are identified by tracing from predicate and argument heads along the dependency tree. All predicates are associated with at least one event unless both *Arg0* and *Arg1* are not identified, or the only argument is not a noun.

The algorithm checks for modifiers based on POS tag[1], tracing up and down the dependency tree, skipping over prepositions, coordinating conjunctions and words indicating apportionment, such as *'sample (of)'*. However, to constrain output the search is limited to a depth of one (with the exception of skipping). For example, given the phrase *'apples from the store nearby'* and an argument head *apples*, the first dependent, *store*, will be extracted but not *nearby*, which is the dependent of store. This can be detrimental when encountering compound nouns but does focus on core information. For verbs, modal dependents are not included in output.

Available paths up and down the dependency tree are followed until all branches are exhausted, given the rules outlined above. Tracing can result in multiple extracted events for one predicate and predicates may also appear as arguments in a different event, or be part of argument phrases. For this reason, events are constrained to cover only detail appearing above subsequent predicates in the tree, which simplifies the event structure. For example, the sentence *"Baghdad already has the facilities to continue producing massive quantities of its own biological and chemical weapons"* results in the event output: (1) *has Baghdad already facilities continue producing*; (2) *continue quantities producing massive*; (3) *producing quantities massive weapons biological*; (4) *quantities weapons biological massive*.

4 HAL With Events

4.1 eHAL-1: Construction From Events

Since events are extracted from documents, they form a reduced text corpus from which HAL can be built in a similar manner to the original HAL. We ignore the parameter of window length (L) and treat every event as a single window of length equal to the number of words in the event. Every pair of words in an event is considered to be co-occurrent with each other. The weight assigned to the association between each pair is simply set to one. With this scheme, all the events are traversed and the event-based HAL is constructed.

The advantage of this method is that it substantially reduces the processing time during HAL construction because only events are involved and there is no need to calculate weights per occurrence. Additional processing time is incurred in semantic role labelling (SRL) during event identification. However, the naive approach to extraction might be simulated with a combination of less costly chunking and dependency parsing, given that the word ordering information available with SRL is not utilised.

eHAL-1 combines syntactical and statistical information, but has a potential drawback in that only events are used during construction so some information existing in the co-occurrence patterns of the original text may be lost. This motivates the second method.

4.2 eHAL-2: Event-Based Filtering

This method attempts to include more statistical information in eHAL construction. The key idea is to decide whether a text segment in a corpus should be used for the HAL construction, based on how much event information it covers. Given a corpus of text and the events extracted from it, the eHAL-2 method runs as follows:

1. Select the events of length M or more and discard the others for efficiency;

2. Set an "inclusion criterion", which decides if a text segment, defined as a word sequence within an L-size sliding window, contains an event. For example, if 80% of the words in an event are contained in a text segment, it could be considered to "include" the event;

3. Move across the whole corpus word-by-word with an L-size sliding window. For each window, complete Steps 4-7;

4. For the current L-size text segment, check whether it includes an event according to the "inclusion criterion" (Step 2);

[1]To be specific, the modifiers include negation, as well as adverbs or particles for verbal heads, adjectives and nominal modifiers for nominal heads, and verbal or nominal dependents of modifiers, provided modifiers are not also identified as arguments elsewhere in the event.

5. If an event is included in the current text segment, check the following segments for a consecutive sequence of segments that also include this event. If the current segment includes more than one event, find the longest sequence of related text segments. An illustration is given in Figure 1 in which dark nodes stand for the words in a specific event and an 80% inclusion criterion is used.

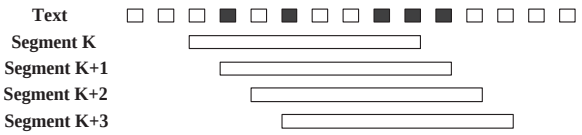

Figure 1: Consecutive segments for an event

6. Extract the full span of consecutive segments just identified and go to the next available text segment. Repate Step 3;

7. When the scanning is done, construct HAL using the original HAL method over all extracted sequences.

With the guidance of event information, the procedure above keeps only those segments of text that include at least one event and discards the rest. It makes use of more statistical co-occurrence information than eHAL-1 by applying weights that are proportional to word separation distance. It also alleviates the identified drawback of eHAL-1 by using the full text surrounding events. A tradeoff is that not all the events are included by the selected text segments, and thus some syntactical information may be lost. In addition, the parametric complexity and computational complexity are also higher than eHAL-1.

5 Evaluation

We empirically test whether our event-based HALs perform better than the original HAL, and standard LM and RM, using three TREC[2] collections: AP89 with Topics 1-50 (*title* field), AP8889 with Topics 101-150 (*title* field) and WSJ9092 with Topics 201-250 (*description* field). All the collections are stemmed, and stop words are removed, prior to retrieval using the Lemur Toolkit Version 4.11[3]. Initial retrieval is identical for all models evaluated: KL-divergence based LM smoothed using Dirichlet prior with μ set to 1000 as appropriate for TREC style title queries (Lavrenko, 2004). The top 50 returned documents form the basis for all pseudo-relevance feedback, with other parameters tuned separately for the RM and HAL methods.

For each dataset, the number of feedback terms for each method is selected optimally among 20, 40, 60, 80[4] and the interpolation and smoothing coefficient is set to be optimal in [0,1] with interval 0.1. For RM, we choose the first relevance model in Lavrenko and Croft (2001) with the document model smoothing parameter optimally set at 0.8. The number of feedback terms is fixed at 60 (for AP89 and WSJ9092) and 80 (for AP8889), and interpolation between the query and relevance models is set at 0.7 (for WSJ9092) and 0.9 (for AP89 and AP8889). The HAL-based query expansion methods add the top 80 expansion terms to the query with interpolation coefficient 0.9 for WSJ9092 and 1 (that is, no interpolation) for AP89 and AP8889. The other HAL-based parameters are set as follows: shortest event length $M = 5$, for eHAL-2 the "inclusion criterion" is 75% of words in an event, and for HAL and eHAL-2, window size $L = 8$. Top expansion terms are selected according to the formula:

$$P_{HAL}(t_j|\oplus t) = \frac{HAL(t_j|\oplus q)}{\sum\limits_{t_i} HAL(t_i|\oplus q)}$$

where $HAL(t_j|\oplus q)$ is the weight of t_j in the combined HAL vector $\oplus q$ (Bruza and Song, 2002) of original query terms. Mean Average Precision (MAP) is the performance indicator, and t-test (at the level of 0.05) is performed to measure the statistical significance of results.

Table 2 lists the experimental results[5]. It can be observed that all the three HAL-based query expansion methods improve performance over the LM and both eHALs achieve better performance than original HAL, indicating that the incorporation of event information is beneficial. In addition, eHAL-2 leads to better performance than eHAL-1, suggesting that use of linguistic information as a constraint on statistical processing, rather than the focus of extraction, is a more effective strategy. The results are still short of those achieved

[2]TREC stands for the Text REtrieval Conference series run by NIST. Please refer to http://trec.nist.gov/ for details.

[3]Available at http://www.lemurproject.org/

[4]For RM, feedback terms were also tested on larger numbers up to 1000 but only comparable result was observed.

[5]In Table 2, brackets show percent improvement of eHALs / RM over HAL / eHAL-2 respectively and * and # indicate the corresponding statistical significance.

Method	AP89	AP8889	WSJ9092
LM	0.2015	0.2290	0.2242
HAL	0.2299	0.2738	0.2346
eHAL-1	0.2364 (+2.83%)	0.2829 (+3.32%*)	0.2409 (+2.69%)
eHAL-2	0.2427 (+5.57%*)	0.2850 (+4.09%*)	0.2460 (+4.86%*)
RM	0.2611 (+7.58%#)	0.3178 (+11.5%#)	0.2676 (+8.78%#)

Table 2: Performance (MAP) comparison of query expansion using different HALs

with RM, but the gap is significantly reduced by incorporating event information here, suggesting this is a promising line of work. In addition, as shown in (Bai et al., 2005), the Information Flow method built upon the original HAL largely outperformed RM. We expect that eHAL would provide an even better basis for Information Flow, but this possibility is yet to be explored.

As is known, RM is a pure unigram model while HAL methods are dependency-based. They capture different information, hence it is natural to consider if their strengths might complement each other in a combined model. For this purpose, we design the following two schemes:

1. Apply RM to the feedback documents (original RM), the events extracted from these documents (eRM-1), and the text segments around each event (eRM-2), where the three sources are the same as used to produce HAL, eHAL-1 and eHAL-2 respectively;

2. Interpolate the expanded query model by RM with the ones generated by each HAL, represented by HAL+RM, eHAL-1+RM and eHAL-2+RM. The interpolation coefficient is again selected to achieve the optimal MAP.

The MAP comparison between the original RM and these new models are demonstrated in Table 3[6]. From the first three lines (Scheme 1), we can observe that in most cases the performance generally deteriorates when RM is directly run over the events and the text segments. The event information is more effective to express the information about the term dependencies while the unigram RM ignores this information and only takes

[6]For rows in Table 3, brackets show percent difference from original RM.

Method	AP89	AP8889	WSJ9092
RM	0.2611	0.3178	0.2676
eRM-1	0.2554 (-2.18%)	0.3150 (-0.88%)	0.2555 (-4.52%)
eRM-2	0.2605 (-0.23%)	0.3167 (-0.35%)	0.2626 (-1.87%)
HAL +RM	0.2640 (+1.11%)	0.3186 (+0.25%)	0.2727 (+1.19%)
eHAL-1 +RM	0.2600 (-0.42%)	0.3210 (+1.01%)	0.2734 (+2.17%)
eHAL-2 +RM	0.2636 (+0.96%)	0.3191 (+0.41%)	0.2735 (+2.20%)

Table 3: Performance (MAP) comparison of query expansion using the combination of RM and term dependencies

the occurrence frequencies of individual words into account, which is not well-captured by the events. In contrast, the performance of Scheme 2 is more promising. The three methods outperform the original RM in most cases, but the improvement is not significant and it is also observed that there is little difference shown between RM with HAL and eHALs. The phenomenon implies more effective methods may be invented to complement the unigram models with the syntactical and statistical dependency information.

6 Conclusions

The application of original HAL to query expansion attempted to incorporate statistical word association information, but did not take into account the syntactical dependencies and had a high processing cost. By utilising syntactic-semantic knowledge from event modelling of pseudo-relevance feedback documents prior to computing the HAL space, we showed that processing costs might be reduced through more careful selection of word co-occurrences and that performance may be enhanced by effectively improving the quality of pseudo-relevance feedback documents. Both methods improved over original HAL query expansion. In addition, interpolation of HAL and RM expansion improved results over those achieved by either method alone.

Acknowledgments

This research is funded in part by the UK's Engineering and Physical Sciences Research Council, grant number: EP/F014708/2.

References

Bach E. The Algebra of Events. 1986. *Linguistics and Philosophy*, 9(1): pp. 5–16.

Bai J. and Song D. and Bruza P. and Nie J.-Y. and Cao G. Query Expansion using Term Relationships in Language Models for Information Retrieval 2005. In: *Proceedings of the 14th International ACM Conference on Information and Knowledge Management*, pp. 688–695.

Bruza P. and Song D. Inferring Query Models by Computing Information Flow. 2002. In: *Proceedings of the 11th International ACM Conference on Information and Knowledge Management*, pp. 206–269.

Deerwester S., Dumais S., Furnas G., Landauer T. and Harshman R. Indexing by latent semantic analysis. 1990. *Journal of the American Sociaty for Information Science*, 41(6): pp. 391–407.

Gao J. and Nie J. and Wu G. and Cao G. Dependence Language Model for Information Retrieval. 2004. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 170–177.

Harris Z. 1968. *Mathematical Structures of Language.*. Wiley, New York.

Johansson R. and Nugues P. Dependency-based Syntactic-semantic Analysis with PropBank and NomBank. 2008. In: *CoNLL '08: Proceedings of the Twelfth Conference on Computational Natural Language Learning*, pp. 183–187.

Landauer T., Foltz P. and Laham D. Introduction to Latent Semantic Analysis. 1998. *Discourse Processes*, 25: pp. 259–284.

Lavrenko V. 2004. *A Generative Theory of Relevance*, PhD thesis, University of Massachusetts, Amherst.

Lavrenko V. and Croft W. B. Relevance Based Language Models. 2001. In: *SIGIR '01: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 120–127, New York, NY, USA, 2001. ACM.

Lin D. and Pantel P. DIRT - Discovery of Inference Rules from Text. 2001. In: *KDD '01: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 323–328, New York, NY, USA.

Lund K. and Burgess C. Producing High-dimensional Semantic Spaces from Lexical Co-occurrence. 1996. *Behavior Research Methods, Instruments & Computers*, 28: pp. 203–208. Prentice-Hall, Englewood Cliffs, NJ.

Metzler D. and Bruce W. B. A Markov Random Field Model for Term Dependencies 2005. In: *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 472–479, New York, NY, USA. ACM.

Metzler D. and Bruce W. B. Latent Concept Expansion using Markov Random Fields 2007. In: *SIGIR '07: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 311–318, ACM, New York, NY, USA.

Pado S. and Lapata M. Dependency-Based Construction of Semantic Space Models. 2007. *Computational Linguistics*, 33: pp. 161–199.

Shen D. and Lapata M. Using Semantic Roles to Improve Question Answering. 2007. In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 12–21.

Sleator D. D. and Temperley D. Parsing English with a Link Grammar 1991. *Technical Report CMU-CS-91-196*, Department of Computer Science, Carnegie Mellon University.

Smeaton A. F., O'Donnell R. and Kelledy F. Indexing Structures Derived from Syntax in TREC-3: System Description. 1995. In: *The Third Text REtrieval Conference (TREC-3)*, pp. 55–67.

Song F. and Croft W. B. A General Language Model for Information Retrieval. 1999. In: *CIKM '99: Proceedings of the Eighth International Conference on Information and Knowledge Management*, pp. 316–321, New York, NY, USA, ACM.