# Evaluation of Syntactic Phrase Indexing — CLARIT NLP Track Report

Chengxiang Zhai[1], Xiang Tong[1], Nataša Milić-Frayling[2], David A. Evans[2]

[1] Laboratory for Computational Linguistics
Carnegie Mellon University
Pittsburgh, PA 15213
{cz25@andrew.cmu.edu, xt22@andrew.cmu.edu}

[2] CLARITECH Corporation
5301 Fifth Ave.
Pittsburgh, PA 15232-2124
{natasa@clarit.com, dae@clarit.com}

## 1 Introduction

The CLARIT$^{TM}$[1] NLP track effort is focused on evaluating the usefulness of syntactic phrases for document indexing. The CLARIT system has several NLP techniques integrated with the vector space retrieval model [Evans et al. 91, Evans et al. 95]. The NLP techniques used in CLARIT include morphological analysis, robust noun-phrase parsing, and automatic construction of first order thesauri, among others. One main feature of CLARIT indexing is that it emphasizes phrase-based indexing with different options for decomposing noun phrases into smaller constituents, including single words. In past TRECs, the default mode for indexing involved full noun phrases, single words, and occasionally selected sub-phrases. [2] While some early experiments have shown the effectiveness of noun phrases for indexing [Evans et al. 91], there is no direct evaluation of their effectiveness in the context of TREC. In particular, the contribution of small sub-phrases to retrieval performance has not been evaluated outside the context of overall system performance.

The version of the CLARIT system that we used in the experiments has its NLP component tightly integrated with the rest of the system. This does not allow easy evaluation of the individual NLP components. We, therefore, developed separate NLP modules and used CLARIT as a retrieval engine only to evaluate the effectiveness of phrase-based indexing.

## 2 Phrases for indexing

Most current IR systems use word-based indexing often supplemented with phrases that are obtained using simple statistical approaches, such as a frequency counting of adjacent word pairs. Intuitively, it seems that syntactic phrases (i.e., phrases with certain syntactic relations) can represent document content more precisely than single words and phrases discovered using simple statistical methods, and are thus potentially useful for indexing [Evans et al. 91, Evans et al. 95, Strzalkowski et al. 95, Evans & Zhai 96, Hull et al. 96, Strzalkowski et al. 97].

Single words, as indexing units, may have two different kinds of problems:

[1] CLARIT is a registered trademark of CLARITECH Corporation.

[2] In the TREC-4 ad hoc task, attested subcompounds were used in addition to the full noun phrase and single words.

1. They may be misleading.

   In the context of lexical atoms,[3] such as "hot dog", the contained single words do not carry their regular meanings, and are thus very misleading if used as separate indexing terms;

2. They may be too general.

   For example, the individual words "junior" and "college" are not specific enough to distinguish "college junior" from "junior college".

In the first case, it is desirable to use the phrase as a whole nonseparable unit for indexing and not to include the constituent single words in the index. In the second case, it is desirable to supplement single words with more specific and discriminative phrases. The use of simple phrases as additional indexing terms addresses the second problem. While it is very interesting and important to compare syntactic phrases with such simple phrases (cf., for example, [Fagan 87, Hull et al. 96, Strzalkowski et al. 97], we are more interested in evaluating different kinds of syntactic phrases. Based on these observations, we are inclined to propose the following two hypotheses:

1. The use of lexical atoms, such as "hot dog", to replace single words for indexing would increase both precision and recall;

2. The use of syntactic phrases, such as "junior college" to supplement single words would increase precision without hurting recall and using more such phrases results in greater improvement in precision.

The goal of our efforts with the NLP track is to test these two hypotheses.

In the literature, syntactic phrases have been reported to show no significant improvement in retrieval performance (cf., for example, [Lewis 91, Belkin & Croft 87, Fagan 87], among others). However, the size of the collection used in early experiments is relatively small. It is important to determine whether a larger size collection will make a difference.

# 3 Experiment Design

## 3.1 Lexical atom experiment

The goal of lexical atom experiments is to test the effect of taking adjacent word pairs such as "hot dog" as non-separable units, or lexical atoms, for indexing. Single words would not be included in the index unless they occur in the document as single word noun phrases, or as part of decomposable phrases (i.e., phrases that are not lexical atoms). This is expected to have an impact on the retrieval performance by eliminating the possible false matching with single words that form a lexical atom.

Lexical atoms are identified automatically by the two heuristics tests based on the following observations: (1) words in lexical atoms usually have strong fixed association, and thus tend to co-occur as a phrase and (2) when the words in a lexical atom co-occur in a noun phrase, they are never or rarely separated. Intuitively, we are looking for those high frequency word pairs that tend not to be separated by other words within the context of noun phrases. For the experiments reported in this paper, we only considered the pairs formed by two nouns or one adjective followed by a noun. A detailed description of the method can be found in [Evans & Zhai 96].

To test the effect of lexical atoms for any particular configuration of the system, we add the lexical atoms to the CLARIT lexicon, and specify that they should be treated as a non-separable indexing unit. The database and the queries are then processed using the extended lexicon to prevent lexical atoms from being decomposed. The results can be compared with a baseline obtained by using the standard CLARIT lexicon[4] to index the database and process the queries. Since the lexicon is the only difference, any performance difference will reflect the effect of the lexical atoms added to the lexicon.

[3] It is hard to give a strict definition for lexical atoms, but it generally refers to a fixed or "sticky" phrase where at least one constituent word's regular meaning is quite different from the meaning of the phrase.

[4] The lexicon has currently about 80,000 lexical items including a few common proper names.

## 3.2 Phrase combination experiments

The goal of phrase combination experiments is to test the effectiveness of using syntactic phrases to supplement single words for indexing. Specifically, we want to see if adding phrases to the index will improve the performance significantly and to see how the performance changes according to different combinations of phrases.

The general procedure of the phrase combination experiments is described in Figure 1. We first use a separately developed phrase extraction module to generate different sets of indexing terms including both single words and phrases, and then make the CLARIT system treat every term we generated as a non-decomposable noun phrase. Basically we "force" the CLARIT system to accept the generated indexing set as is. In this way, we have total control over the indexing terms actually used by the CLARIT system, and the CLARIT system essentially serves only as a "blind" retrieval engine, although the thesauri-based feedback mechanism in CLARIT is still used.

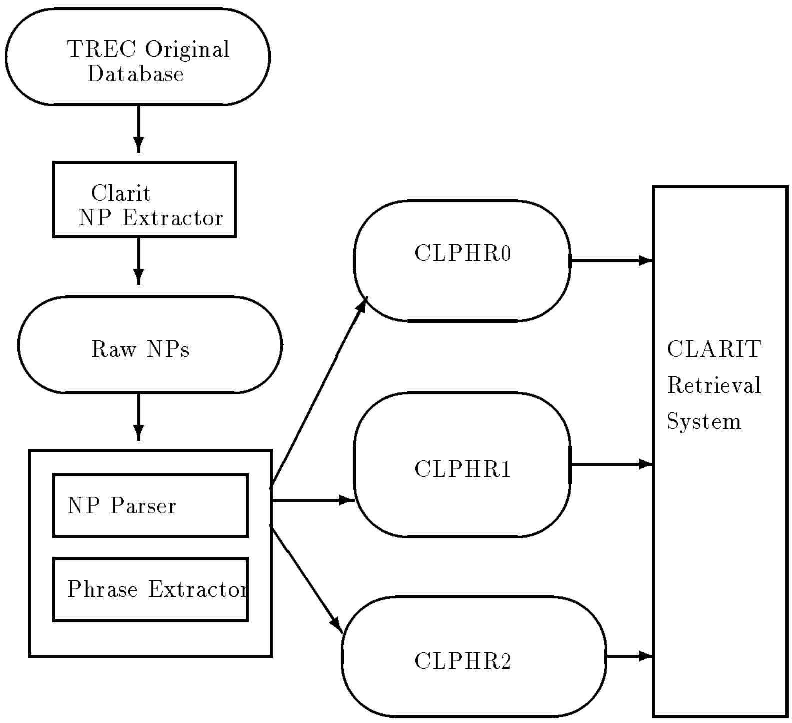


Figure 1: Procedure for Phrase Combination Experiments

The noun phrase parser uses an Expectation Maximization (EM) algorithm to obtain statistical evidence of word modifications from the noun phrases in the corpus. Such evidence is then used to produce an internal modification structure for each noun phrase. The basic idea is as follows. A noun phrase can be assumed to be generated from a word modification structure. Since noun phrases with more than two words are structurally ambiguous, if we only observe the noun phrase, then the actual structure that generates the noun phrase is "hidden". We therefore treat the noun phrases with their possible structures as the complete data and the noun phrases occurring in the corpus (without the structures) as the observed incomplete data. In the training phase, an EM algorithm can be used to estimate the parameters of word modification probabilities by maximizing the conditional expectation of the likelihood of the complete data (i.e., noun phrases with explicit structures) given the observed incomplete data (i.e., noun phrases in the corpus). In the parsing phase, a noun phrase is assigned the structure that has the maximum conditional probability given the noun phrase (See [Zhai 97] for more details). Once the structure of a noun phrase is available, four different kinds/levels of terms may be generated from the noun phrase by the phrase extractor: (1) single words; (2) head modifier pairs (i.e., any word pair that has a linguistic modification relation); (3) adjacent subphrases; and (4) the full noun phrase. For example, from the phrase structure "[[[heavy=construction]=industry]]=group]" (a real example from WSJ90), it is possible to generate the following candidate terms:

```
heavy, construction, industry, group (single words)
construction industry, industry group, heavy construction (head modifiers)
heavy construction industry (adjacent subphrase)
heavy construction industry group (full noun phrase)
```

Different combinations of the four kinds of terms can be selected for indexing. In particular, the indexing set formed solely of single words is used as a baseline to test the effect of using phrases. For example, the three automatic official runs (i.e., CLPHR0, CLPHR1, and CLPHR2) correspond to the following three levels of term combinations:

```
-- CLPHR0: single word only (no phrases, baseline)
-- CLPHR1: single word + head modifier pair + full NP
-- CLPHR2: single word + head modifier pair + adjacent subphrase + full NP
```

# 4 Result analysis

## 4.1 Lexical atom experiments

In lexical atom experiments we used the same topic profiles that were used in the main manual ad hoc task. The profiles were created manually with the aid of the CLARIT thesaurus extraction module. In addition to the term vector representation of the topic, they also contain Boolean type constraints which may be used to select documents for CLARIT automatic feedback. (See [Milić-Frayling et al. 97] for details of the construction procedure.) The two runs that we submitted for evaluation, CLATMN and CLATMC, differ in the use of constraints. CLATMN topic profiles involve only the query term vector, while CLATMC uses both the query term vector and the constraints. Therefore, automatic query augmentation in CLATMN is based on the top 50 documents in the list of documents retrieved by using the initial query term vector. In CLATMC, however, the list of the retrieved documents is reranked by "pulling up" the documents satisfying the constraints, and the feedback involves the top 50 documents in the reranked list of documents.[5]

In order to include lexical atoms into database index, we first identified approximately 13,000 lexical atoms from the WSJ90 database using the method described in [Evans & Zhai 96], and generated the extended CLARIT lexicon. Both CLATMC and CLATMN used the extended lexicon to parse the document text and query. For the purpose of comparison, we performed the experiments that use the standard CLARIT lexicons: CLATMC-Base and CLATMN-Base, and treat them as the baseline runs. Since the only difference between the official and baseline runs is in their respective lexicons, the difference in the retrieval performance is attributed to the addition of lexical atoms. The results are shown in Figure 2 and Table 1.

From Table 1, we see that indexing with lexical atoms yields higher recall and average precision but a lower initial precision than the corresponding baseline runs. This is different from the results that we obtained in our preliminary experiments using TREC-4 topics. For TREC-4 topics, the lexical atoms improved average precision and initial precision but did not consistently help recall. (See Table 2.)

From both the TREC-5 and the preliminary experiments we see that the use of lexical items leads to a slight but consistent improvement in average precision. This indicates that using lexical atoms to replace single words for indexing generally results in a greater retrieval accuracy than

[5] If the total number of documents that satisfy the constraints is less than 50, it is possible that fewer than 50 documents are actually used for feedback.

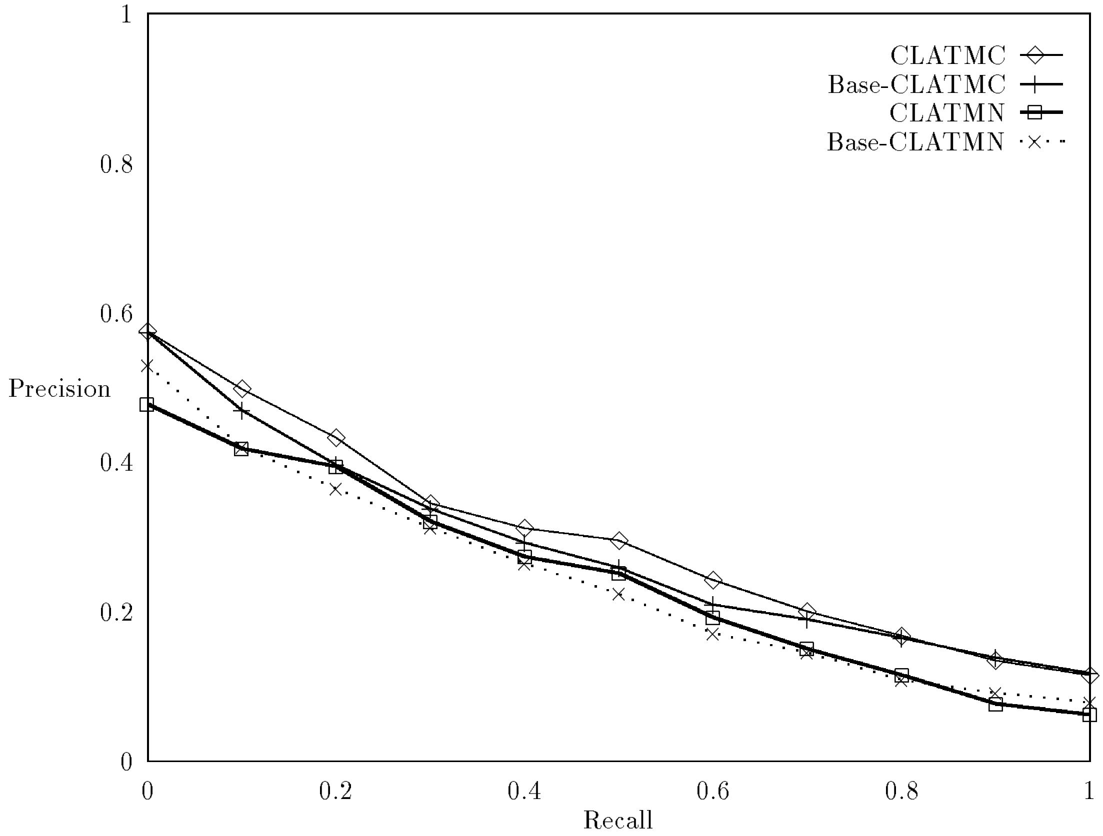


Figure 2: Precision and recall for lexical atom experiments

indexing with single words alone. On the other hand, it seems that using lexical atoms does not consistently improve recall and initial precision. In fact, it seems that it increases either recall or the initial precision.

A brief manual inspection of the identified lexical atoms shows that many, in fact, most, are not true lexical atoms. However, they are usually good terminological phrases such as "stock market" and "annual meeting". Thus, the results discussed above are in fact the results of using a very rich set of good terminology, including true lexical atoms, to replace the corresponding single words. We believe that the above observed inconsistency in the effect on initial precision and recall may be resolved by a better control over the use of these phrases. (For example, some non-lexical-atoms should supplement rather than replace single words. We should distinguish them from lexical atoms.) We would also like to point out that, in our approach, a single word contained in a lexical atom will still be used for indexing, if the word occurs separately in another non-lexical-atom phrase. We believe that in most cases, lexical atoms only reduced the frequency of the contained single words, and the single words might still be used for indexing owing to other occurrences of the words.

## 4.2 Phrase combination experiments

In TREC 5, we submitted three official experiments which use different combinations of phrases for indexing: CLPHR0, CLPHR1, and CLPHR2. CLPHR0 used only single words and serves as our

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| CLATMN-BASE | 0.71(753) | 0.5293 | 0.2270 |
| CLATMN | 0.74(786) | 0.4767 | 0.2346 |
| increase | 4% | -9% | 3% |
| CLATMC-BASE | 0.70(748) | 0.5744 | 0.2659 |
| CLATMC | 0.72(766) | 0.5752 | 0.2842 |
| increase | 3% | 0% | 7% |
| Total relevant documents: 1064 | | | |

Table 1: Lexical atom experiments

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| Auto-BASE | 0.80(944) | 0.4992 | 0.2291 |
| Auto-LEX | 0.76(901) | 0.5114 | 0.2350 |
| increase | -4% | 2% | 1% |
| Manual-BASE | 0.89(1058) | 0.7140 | 0.3631 |
| Manual-LEX | 0.90(1063) | 0.7384 | 0.3701 |
| increase | 0% | 3% | 2% |
| Total relevant documents: 1183 | | | |

Table 2: Lexical atom preliminary experiments on TREC-4 topics

baseline; CLPHR1 used single words, head modifier pairs, and the full noun phrase; CLPHR2 used adjacent subphrases in addition to those terms used by CLPHR1. We also performed the experiment CLPHR0.5 which uses single words and head modifier pairs only. In all these experiments we used the long format of the topic, i.e., all the fields in the topic description. In each experiment, the queries were processed in the same way as the documents. This means that the same combination of phrases is used for both the document and the query. Moreover, the queries were processed with one round of automatic feedback using the top 10 documents from the initial retrieval.[6]

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| CLPHR0 | 0.53(567) | 0.4235 | 0.1834 |
| CLPHR0.5 | 0.57(607) | 0.4585 | 0.2064 |
| inc over CLPHR0 | 8% | 8% | 13% |
| CLPHR1 | 0.57(609) | 0.4465 | 0.2010 |
| inc over CLPHR0 | 8% | 5% | 9% |
| CLPHR2 | 0.58(622) | 0.4316 | 0.1997 |
| inc over CLPHR0 | 9% | 2% | 9% |
| Total relevant documents: 1064 | | | |

Table 3: Effects of Phrases in the official TREC-5 runs

In Table 3 we show the comparison of the experiments that use phrases with the baseline run that uses only single words. We note that the use of phrases to supplement single words yields between 8% and 9% relative increase in recall and between 9% and 13% relative increase in average precision. Figure 3 in Appendix A provides the per-topic comparison between CLPHR0 and CLPHR1 in terms of their recall and average precision. It is clear from this figure that the general improvement in

[6] Using only 10 documents for feedback yields a very selective (and limited) set of supplemental terminology. There is no guarantee that the additional terms add relevant phrasal combinations.

performance of CLPHR1 over CLHR0 does not come from an even improvement on all topics. In fact, for some topics, the results of CLPHR1 are better, while for others, the results of CLPHR0 are better, and the difference in performance is more significant for some topics than for others.

The relative merits of different phrase sets are unclear. For example, while CLPHR2 produces a higher recall than CLPHR0.5, its precision is lower than for CLPHR0.5. Figure 4 in Appendix A shows the per-topic comparison between CLPHR1 and CLPHR2 in terms of their recall and average precision. Although the picture in this figure looks very similar to that in Figure 3, the scales in Figure 4 are all much smaller than in Figure 3, indicating that the difference between CLPHR1 and CLPHR2 is much less significant than that between CLPHR0 and CLPHR1. Since CLPHR2 uses more phrases than CLPHR1, the mixed relative improvement shown in Figure 4 indicates that adding more phrases does help some topics, but may also hurt some others.

In the official runs, we did not use the frequency of terms in the query for calculating the term weight. In subsequent experiments (CLPHR0-TF, CLPHR0.5-TF, CLPHR1-TF, and CLPHR2-TF), we found that using query term frequency consistently improves both recall and precision, as shown in Table 4.

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| CLPHR0 | 0.53(567) | 0.4235 | 0.1834 |
| CLPHR0-TF | 0.56(597) | 0.4546 | 0.2208 |
| increase | 5% | 7% | 20% |
| CLPHR0.5 | 0.57(607) | 0.4585 | 0.2064 |
| CLPHR0.5-TF | 0.60(637) | 0.5125 | 0.2398 |
| increase | 5% | 12% | 16% |
| CLPHR1 | 0.57(609) | 0.4465 | 0.2010 |
| CLPHR1-TF | 0.63(671) | 0.4845 | 0.2275 |
| increase | 10% | 9% | 13% |
| CLPHR2 | 0.58(622) | 0.4316 | 0.1997 |
| CLPHR2-TF | 0.62(661) | 0.4839 | 0.2337 |
| increase | 6% | 12% | 17% |
| Total relevant documents: 1064 | | | |

Table 4: Comparison of official and post-TREC-5 runs

The retrieval performance seems to be sensitive to the weighting of query terms. But note that the same effect of using phrases for indexing can be observed when the queries are processed with/without the term frequency, as shown in Table 5.

Our preliminary experiments with TREC-3 topics in the default mode, without using term frequency in the query, also show the positive effect of phrases. (See Table 6.) In fact, the relative improvement in average and initial precision for TREC-3 topics is much higher, between 10% and 23% for the average precision and 17% to 23% for the initial precision. Relative increase in recall is between 5% and 7%. The experiments with TREC-3 queries also show consistent improvement in performance as more phrases are added, PRE-CLPHR2 being the best of the three phrase runs. However, this trend in performance improvement was not observed in TREC-5 experiments.

In Table 5, we also include information about the NLP baseline runs for the SMART system, performed by the Xerox group. XEROX_NLP1 is pure word-based SMART, using SMART stemming; XEROX_NLP2 is pure SMART with SMART-like contiguous phrases. Therefore, XEROX_NLP1 is comparable to CLPHR0-TF; while XEROX_NLP2 is comparable to CLPHR0.5-TF, CLPHR1-TF, and CLPHR2-TF. In comparison with the SMART baseline runs, the CLARIT system generally achieves better precision and worse recall. Indeed, from the performance statistics of the two baseline runs, CLPHR0-TF and XEROX_NLP1, it seems that the selection of single words for indexing by the CLARIT system is more attuned to higher precision than recall.[7] The phrases that are used

[7] In TREC 5, the CLARIT ad hoc runs and NLP runs both found the largest number of unique relevant documents

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| CLPHR0-TF | 0.56(597) | 0.4546 | 0.2208 |
| CLPHR0.5-TF | 0.60(637) | 0.5125 | 0.2398 |
| inc over CLPHR0-TF | 7% | 13% | 9% |
| CLPHR1-TF | 0.63(671) | 0.4845 | 0.2275 |
| inc over CLPHR0-TF | 12% | 7% | 3% |
| CLPHR2-TF | 0.62(661) | 0.4839 | 0.2337 |
| inc over CLPHR0-TF | 11% | 6% | 6% |
| XEROX_NLP1 | 0.62(664) | 0.4673 | 0.2002 |
| XEROX_NLP2 | 0.63(672) | 0.4789 | 0.2147 |
| inc over XEROX_NLP1 | 1% | 2% | 7% |
| Total relevant documents: 1064 | | | |

Table 5: Phrase effect on post-TREC-5 runs

| RUNs | Recall(Ret-rel) | Init Prec | Avg Prec |
|---|---|---|---|
| PRE-CLPHR0 | 0.77(1576) | 0.6183 | 0.2782 |
| PRE-CLPHR0.5 | 0.81(1653) | 0.7234 | 0.3061 |
| inc over PRE-CLPHR0 | 5% | 17% | 10% |
| PRE-CLPHR1 | 0.82(1664) | 0.7263 | 0.3377 |
| inc over PRE-CLPHR0 | 6% | 17% | 21% |
| PRE-CLPHR2 | 0.83(1687) | 0.7633 | 0.3416 |
| inc over PRE-CLPHR0 | 7% | 23% | 23% |
| Total relevant documents: 2041 | | | |

Table 6: Effects of phrases in the preliminary experiments with TREC-3 topics

by the CLARIT system, on the other hand, lead to a relative increase in average precision that is comparable with the increase for the SMART system but much higher for the recall and the initial precision. Indeed, while phrases used by SMART help improve recall by 1% and the initial precision by 2%, CLARIT phrases result in the relative increase between 7% and 12% for recall and between 6% and 13% for the initial precision.

# 5 Conclusion

The CLARIT NLP track effort has evaluated the effectiveness of the syntactic phrases generated by a statistical NLP module based on the output from the CLARIT noun phrase extractor. The NLP module is very fast and can be used to perform the experiments on the full set of ad-hoc databases.[8] Results of our official and post-TREC-5 experiments have shown that the two hypotheses that we proposed in Section 2 are plausible.

Our lexical atom experiments have shown that exploiting lexical atoms to replace or simply reduce the frequency of the single words that form lexical atoms consistently (although only slightly) increases the average precision. The inconsistent influence of lexical atoms on the recall and initial precision may indicate a need for a better control over the selection of phrases that are used for replacing single words.

among all the participating groups. It will be important, subsequently, to investigate the reasons why the CLARIT system found so many documents that other systems missed.

[8] It takes about 1 hour to parse the whole WSJ90 on a 133-MHz DEC Alpha workstation, and the training takes about 5 hours.

Experiments in supplementing single words by various combination of syntactic phrases in the indexing process have shown a consistent and significant improvement in retrieval performance. The general conclusion that syntactic phrases have positive effects when supplementing single words for indexing is also supported by experiments conducted by other groups such as Xerox [Hull et al. 96] and GE/Lockheed Martin/Rutgers/NYU [Strzalkowski et al. 97]. In our experiments, this effect of phrases seems to hold even in combination with other techniques, such as using query term frequency for term weighting. However, the impact of adding phrases into the index space varies according to the query topic. Thus, while adding phrases helps some topics, it hurts some others. Although the results of the performed experiments are revealing, we still need to conduct more experiments to understand how to combine single words and phrases more effectively and to determine the merits of different types of phrases. We also need to analyze varying effects of phrases across different topics.

One thing we need to keep in mind when interpreting these results is that the effect of phrases can be best shown when the phrases also occur in the queries. However, all the results we have discussed came from the experiments using TREC-5 ad-hoc topics, which generally contains very few phrases. The lack of phrases in the queries makes it even more difficult to show the effect of using additional phrases for indexing, since such additional phrases (discovered by the NLP module) have even a smaller chance of occurring in the queries[9]. It is, thus, not a surprise that the GE/Lockheed Martin/Rutgers/NYU group has found that syntactic phrases seem to be more effective with longer queries [Strzalkowski et al. 97]. In fact, the additional phrases that have been used for indexing, but have not shown up in the queries, may have negative side-effects when weighted inappropriately (cf., for example, [Buckley 93]). Thus, phrase weighting is an important factor in the proper evaluation of phrase-based indexing. Based on the above observations, we believe that it would be more appropriate to evaluate phrase-based indexing under the routing task, since, with training documents available, it is possible to formulate queries with more phrases found in the training documents.

## References

[Belkin & Croft 87] Belkin, N., and Croft, B. Retrieval techniques. In: Williams, Martha E.(Ed.), *Annual Review of Information Science Technology*, Vol. 22. Amsterdam, NL: Elsevier Science Publishers. 1987. 110–145.

[Buckley 93] Buckley, Chris. The importance of proper weighting methods. In: *Human Language Technology. Proceedings of a workshop held at Plainsboro*, New Jersey, March 21-24, 1993. 349–352.

[Evans et al. 95] David A. Evans, and Robert G. Lefferts. CLARIT–TREC experiments, *Information Processing and Management*, Vol. 31, No. 3, 1995. 385–395.

[Evans et al. 91] David A. Evans, Kimberly Ginther-Webster, Mary Hart, Robert G. Lefferts, Ira A. Monarch. Automatic indexing using selective NLP and first-order thesauri. In: A. Lichnerowicz (Ed.), *Intelligent Text and Image Handling. Proceedings of a Conference, RIAO '91*. Amsterdam, NL: Elsevier. 1991. 624–644.

[Evans & Zhai 96] Evans, D. and Zhai, C. Noun-phrase analysis in unrestricted text for information retrieval. *Proceedings of the 34th Annual meeting of Association for Computational Linguistics*, Santa Cruz, University of California, June 24–28, 1996. 17–24.

[Fagan 87] Fagan, Joel L. *Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-syntactic Methods*. PhD thesis, Dept. of Computer Science, Cornell University. Sept. 1987.

[Hull et al. 96] Hull, D. et al. Xerox TREC-5 Site Report: Routing, Filtering, NLP, and Spanish Tracks, In: *TREC-5 Conference Notes*. 1996.

[9]However, our experiments with TREC-3 topics did show a consistent improvement of performance when more phrases are used for indexing.

[Lewis 91] Lewis, D. *Representation and Learning in Information Retrieval.* Ph.D thesis, COINS Technical Report 91-93, Univ. of Massachusetts. 1991.

[Milić-Frayling et al. 97] Milić-Frayling, N. Tong, X., Zhai, C., and Evans, D. A. CLARIT Compound Queries and Constraint-controlled Feedback in TREC-5 Ad Hoc Experiments. In: D. Harman (Ed.) *The Fifth Text REtrieval Conference (TREC-5).* NIST Special Publication, 1997. (This Volume.)

[Strzalkowski et al. 95] Strzalkowski, T., et al. Natural language information retrieval: TREC-3 report. In: Harman, D. (Ed.), *Overview of the Third Text REtrieval Conference (TREC-3).* NIST Special Publication, 1995. 500–225.

[Strzalkowski et al. 97] Strzalkowski, T., et al. Natural Language Information Retrieval: TREC-5 Report. In: D. Harman (Ed.), *The Fifth Text REtrieval Conference (TREC-5).* NIST Special Publication, 1997, forthcoming.

[Zhai 97] Zhai, Chengxiang. Fast statistical parsing of noun phrases for document indexing. To appear in *Proceedings of the Fifth Conference on Applied Natural Language Processing*, Washington, D.C. March 31 – April 3, 1997.

# A Per-topic comparisons

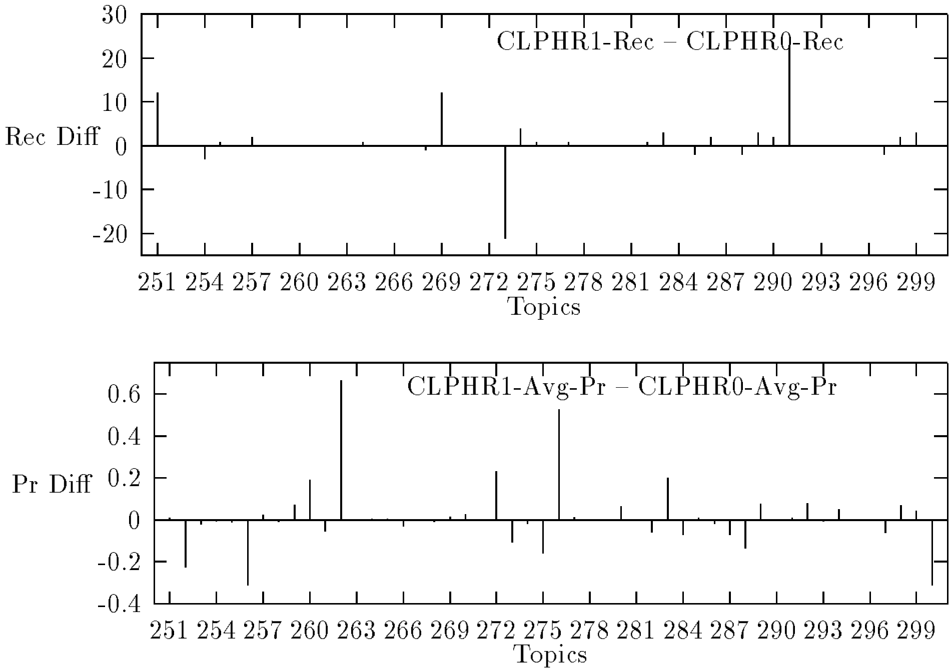


Figure 3: Per-topic Comparison of CLPHR0 and CLPHR1

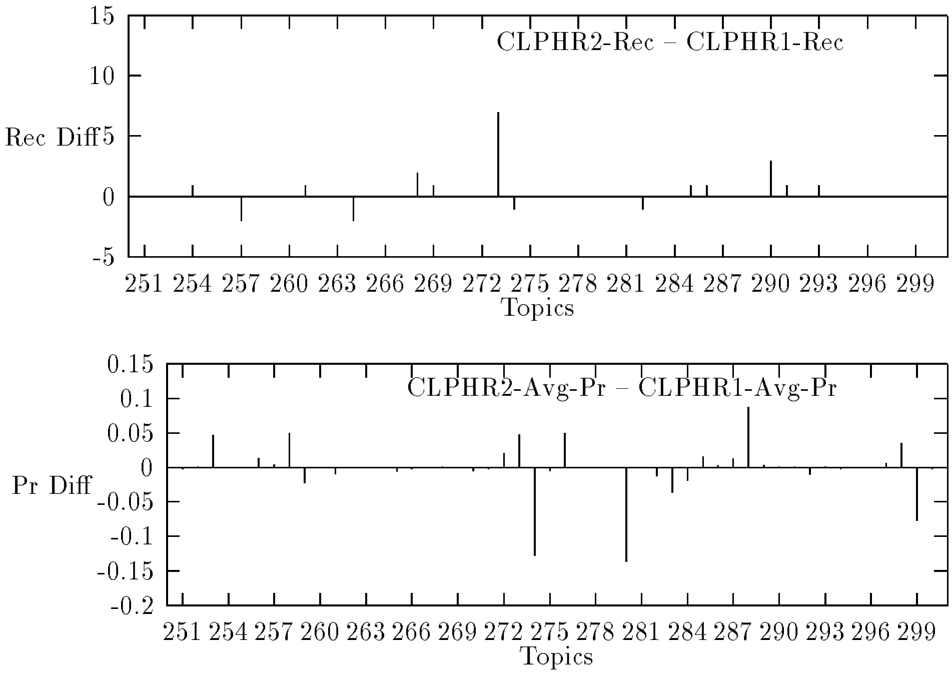


Figure 4: Per-topic Comparison of CLPHR1 and CLPHR2