TREC2003 Robust, HARD and QA Track Experiments using PIRCS

L. Grunfeld, K.L. Kwok, N. Dinstl and P. Deng
Computer Science Department, Queens College, CUNY
Flushing, NY 11367

1 Introduction

We participated in the Robust, HARD and part of the QA tracks in TREC2003. For Robust track, a new way of doing ad-hoc retrieval based on web assistance was introduced. For HARD track, we followed the guideline to generate clarification forms for each topic so as to experiment with user feedback and metadata. In QA, we only did the factoid experiment. The approach to QA was similar to what we have used before, except that WWW searching was added as a front-end processing. These experiments are described in Sections 2, 3 and 4 respectively.

2 Robust Track

Combining the results of a number of different retrieval outcome generally improves the overall performance [1,2]. The intuitive explanation for this phenomenon is that the different retrievals are more likely to rank the same relevant documents early, than the same non-relevant documents. Consequently, combining retrieval methods that differ greatly often yields better results. Paradoxically we can obtain robust retrieval by adding the results of non-robust methods.

We start with our high performance PIRCS retrieval engine, which is considered robust since it is based on statistical methods, and combine it with retrievals for which the queries were generated based on returned web pages by the Google search engine operating on WWW data. Google queries are of Boolean type and returned results may be less stable. The addition of a single word can dramatically alter retrieval lists, and hence the queries defined by them.

For each robust task topic, our approach is to employ the 60 best-weighted words (except for common words on a stop list) contained in the top 20 web pages (returned by Google) as a reformulated query for our PIRCS engine. The rationale is that because the web is so huge and rich in content, there is a good chance that relevant pages containing the content terms of the original topic exist in the web. These pages will probably rank near the top by the Google, and may be rich in content terms related to the topic. These terms can therefore define, for our ad-hoc processing, useful alternate queries that can lead to different retrievals, and which could be useful for combining with the original retrieval list that is based on a query generated directly from the topic statement. The next section describes how we form Google queries from the description section of the original topic statement.

2.1 Generating Google Queries from Topic Statements

The Google search engine (http://www.google.com) accepts queries in a form similar to simplified Boolean expressions. It allows one to specify conjunctive clauses by having terms placed adjacent to each other, disjunctive clauses by placing the string OR between terms, and negated terms with a '-' prefixing them. Phrase matching is allowed by having words surrounded by double quotes. Un-stemmed words are used.

We employ three different strategies to create queries for Google retrieval. The queries are formed using only the Description section of a topic. Since previous experience has shown that retrievals combine better

if they are dissimilar, our aim is to make the queries as different as possible. The three query formation strategies (identified by the names qds, qdp and qdt) are described below:

qds queries:

This simplest approach just employs sequentially the first six content words from a topic in a logical AND fashion. Caution is needed to avoid using too many words; otherwise nothing is retrieved. As an example consider the original Query 378 and the generated Google query G378:

Q378 - Identify documents that discuss opposition to the introduction of the euro, the European currency.

G378 - opposition introduction euro, European currency.

This method works fairly well except when queries are long. Consider the following:

Q610 - Find claims made by U.S. small businesses regarding the adverse impact on their businesses of raising the minimum wage.

G610 - claims U.S. small businesses adverse impact

The generated query G610 does not include important terms like: “raising”, “minimum”, “wage”, and the returned pages are not satisfactory.

qdp queries:

This and the following qdt method attempt to create Google queries by identifying important words in the topic based on Dekang Lin's MINIPAR parser [3] available at http://www.cs.ualberta.ca/~lindek/minipar.htm. MINIPAR is a general-purpose parser for an English sentence, identifies phrases, and generates a dependency structure where each word modifies at most one head word. Our strategy is to select the six best nouns based on the following order of priority: nouns appearing in phrases, nouns that designate a person, country, location, corporation, language or title, followed by other nouns. As an example, Q610 above generates the following:

G610 - minimum wage U.S. businesses impact claims

where “minimum”, “wage” come first since it is part of a phrase, followed by “U.S.”, a country, then followed by other nouns. The exclusion of verbs and adjectives sometimes harms performance. Consider Query 644:

Q644 - Identify documents that discuss exotic species of animals that are imported into the U.S. or animals that are imported into the U.S. or U.K.

G644 - U.S. U.K. species animals

The resultant G644 misses out the important verb “imported” and adjective “exotic”. Another example is:

Q362 - Identify incidents of human smuggling.

G362 - incidents smuggling

which misses the important adjective “human”.

qdt queries:

This strategy first selects the phrases identified by MINIPAR. If there are none, other phrases defined by patterns (N1 gov N2), (N1 N2) (N2 N3) (if the 3-word phrase N1 N2 N3 are defined), and (A gov N) are then used to select words in this order. If query is < 6 words, nouns and verbs are added (AND'ed) until query has 6 words. For example, Q362 becomes:

G362 - "human smuggling" incidents

where the quotes tell Google that “human smuggling” needs to be adjacent. Another example is:

Q643 - What harm have power dams in the Pacific northwest caused to salmon fisheries?

G643 - "Pacific northwest" "salmon fisheries" harm dams

which includes most of the content terms. A problem with this method is that some queries become too specific and no web pages are returned.

As an example, we include in the following the output of Q643 after analyzed by MINIPAR:

```
> (
E1      (()     fin C   *       )
1       (What   ~ N     E1      whn     (gov fin))
2       (harm   ~ N     3       s       (gov have))
3       (have   ~ V     E1      i       (gov fin))
E3      (()     harm N  3       subj    (gov have)      (antecedent 2))
E0      (()     fin C   3       fc      (gov have))
4       (power  ~ A     5       mod     (gov dam))
5       (dams   dam N   10      s       (gov cause))
6       (in     ~ Prep  5       mod     (gov dam))
7       (the    ~ Det   9       det     (gov northwest))
8       (Pacific ~ N 9 nn (gov northwest)(atts (sem (+location))))
9       (northwest      ~ N     6       pcomp-n(gov in))
10      (caused cause V E0      i       (gov fin))
E4      (()     what N  10      obj     (gov cause)     (antecedent 1))
E5      (()     dam N   10      subj    (gov cause)     (antecedent 5))
11      (to     ~ Prep  10      mod     (gov cause))
12      (salmon ~ N     13 nn   (gov fishery)(atts (sem (+gname +male))))
13      (fisheries      fishery N       11      pcomp-n(gov to))
)
>
```

2.2 Generating PIRCS Queries from Retrieved Web Pages

Each of the Google queries in the previous sub-section was used to retrieve the top 20 web pages. From these html tags, non-text items and some common words are removed. A query is then created using the 60 best-weighted words. Weight of a word is defined as sum (over all web pages in which it occurs) of its frequency divided by html text length. Exceptionally long pages are skipped. The alternate query for our target retrieval is composed of these 60 words normalized by the least weight. For example, G643 in the previous qds section leads to the following alternative query for our PIRCS retrieval:

SALMON 10 RIVER 9 DAMS 8 WATER 6 COLUMBIA 6 FISH 5
POWER 5 DAM 4 NORTHWEST 3 FEDERAL 3 SNAKE 3
BASIN 2 SPECIES 2 OREGON 2 .. + 46 other single terms

Note that this alternative query includes important geographical information such as “Columbia River”, “Snake River”, “Columbia Basin” and “Oregon” that are absent in the original topic description section. This query has good performance.

2.3 Retrieval based on Data Fusion

The final robust retrieval submission is based on combination of retrieval lists: using our normal query qd (description) or qa (all sections) obtained from a topic statement, and from alternative queries as discussed in the previous sub-section. All these query types undergo retrieval using our PIRCS engine on the given collection. Experiments have been performed using the description section of a topic only (pircRBd?, where ?=1 means normal PIRCS retrieval with PRF (pseudo-relevance feedback), ?=2 means web-assisted retrieval using combination strategy (i), while ?=3 means combination strategy (ii). The combination strategies are: (i) (qd 0.4) $\oplus$ (qds 0.2) $\oplus$ (qdp 0.2) $\oplus$ (qdt 0.2), and (ii) (qd 0.5) $\oplus$ (qdt 0.5). The symbol $\oplus$ is used to denote ranked list combination, and each retrieval list is weighted by the given factors. Experiments using all sections were also submitted pircRBa?: ?=1 means normal PIRCS retrieval, and ?=2 means (qa 0.3) $\oplus$ (pircRBd1 0.7). Note that pircRBa2 not only make use of web-assistance, but also combine description with all-section query results.

2.4 Robust Track Results and Discussions

Results of our submissions are shown in Table 1: split into 50 old topics, 50 new topics and all 100 merged. The 50 old topics may be considered as training topics since their relevant answers are known from

previous TREC experiments, and the 50 new as testing set. Evaluation measures shown are the standard ones used in TREC: Rel.Ret = total relevant items in the 1000 retrieved documents, MAP = mean average precision, R-Pre = average precision value at the exact number of available relevant documents for each query, and Pnn = average precision at nn documents retrieved, where nn = 10, 20 or 30. Two new measures for Robust track are: number of topics without relevant documents at 10 retrieved '# no-relv-@10' and the 'area' measure which is a weighted sum of the precision for the worst-25% of the topics. An immediate observation is that all effectiveness values are much lower for the old topics than for the new, showing that the 50 old topics are much more difficult for retrieval with this TREC-8 collection of documents. In particular, the 'area' values are less than .01 for old 'description' queries, while they vary from .05 to .08 for new queries.

In Table 1, we also show results of our PIRCS retrieval with PRF (pircRBd1 for description query and pircRBa1 for all-section queries) as basis for comparison. Of the two web-assisted description runs pircRBd2 and pircRBd3, the former has better performance. The latter makes use of qdt queries only and is not sufficiently robust, and more combination of retrievals appears useful. Using the 100 query results, one sees that our method of web-assisted retrieval brings substantial improvements for the Robust track measures: reducing the '# no-relv-@10' from 16 in pircRBd1(100) to 8 for pircRBd2(100), while the 'area' value increases from .0122 to .0219, an 80% boost. There are also smaller improvements in the other measures such as MAP, P10, etc. Similar improvements are also observed for the all-section queries.

Table 2 shows percentage improvements of certain measures of the web-assisted runs (for both description and all section queries) compared to their respective basis runs and separated into old and new queries. For the training set (Old-50), the web-assisted retrievals have double-digit percentage improvements compared to basis PIRCS retrieval for the description queries. For the testing (New-50) set, only pircRBd2 has slight improvements. We might have over-trained and the strategy does not carry over to testing set well, or that it is difficult to attain increases for the better performing queries of the New-50. For the long queries, however, except for slight decrease of 1% in two measures, other measures show good improvements over the Basis run in both old training and new testing sets.

Run ID	Rel.Ret	MAP	R.Pre	P10	P20	P30	# no-relv-@10	area
Total No. of Relevant Documents: old50 set=4416, new50 set=1658, all100=6074								
Query size: description section only								
pircRBd1 (old)	2216=50%	.1526	.1887	.3220	.2810	.2393	14/50=28%	.0045
pircRBd1 (new)	*1534=93%*	*.4022*	*.3963*	*.5200*	*.4230*	*.3500*	*2/50=4%*	*.0804*
pircRBd1 (100)	**3750=62%**	**.2774**	**.2925**	**.4210**	**.3520**	**.2947**	**16/100=16%**	**.0122**
Web-assisted runs: (qd 0.4) ⊕ (qds 0.2) ⊕ (qdp 0.2) ⊕ (qdt 0.2)								
pircRBd2 (old)	2377=54%	.1772	.2148	.3820	.3240	.2787	6/50=12%	.0091
pircRBd2 (new)	*1565=94%*	*.4029*	*.3845*	*.5320*	*.4170*	*.3467*	*2/50=4%*	*.0819*
pircRBd2 (100)	**3942=65%**	**.2900**	**.2996**	**.4570**	**.3705**	**.3127**	**8/100=8%**	**.0219**
Web-assisted runs: (qd 0.5) ⊕ (qdt 0.5)								
pircRBd3 (old)	2287=52%	.1754	.2106	.3760	.3250	.2727	7/50=14%	.0065
pircRBd3 (new)	*1444=87%*	*.3878*	*.3781*	*.5220*	*.4080*	*.3273*	*2/50=4%*	*.0540*
pircRBd3 (100)	3731=61%	.2816	.2944	.4490	.3665	.3000	9/100=9%	.0165
Query size: all sections of topic								
pircRBa1 (old)	2562=58%	.1796	.2282	.3640	.3230	.2867	6/50=12%	.0136
PircRBa1 (new)	*1494=90%*	*.4405*	*.4150*	*.5440*	*.4550*	*.3800*	*3/50=6%*	*.0716*
pircRBa1 (100)	**4056=67%**	**.3101**	**.3216**	**.4540**	**.3890**	**.3333**	**9/100=9%**	**.0203**
Web-assisted runs: (qa 0.3) ⊕ (pircRBd1 0.7)								
pircRBa2 (old)	2641=60%	.1854	.2234	.4000	.3340	.2907	5/50=10%	.0135
pircRBa2 (new)	*1575=95%*	*.4369*	*.4159*	*.5760*	*.4520*	*.3760*	*1/50=2%*	*.1062*
pircRBa2 (100)	**4217=69%**	**.3111**	**.3197**	**.4880**	**.3930**	**.3333**	**6/100=6%**	**.0290**

Table 1: Robust Retrieval - Summary for All Submitted Runs -- Lenient Evaluation

	Old-50 training set						New-50 testing set					
	MAP	**%imp**	**P10**	**%imp**	**area**	**%imp**	**MAP**	**%imp**	**P10**	**%imp**	**area**	**%imp**
Query size: description section only												
pircRBd1	.1526	*	.3220	*	.0045	*	.4022	*	.5200	*	.0804	*
Web-assisted runs												
pircRBd2	.1772	+16	.3820	+19	.0091	+102	.4029	+0	.5320	+2	.0819	+2
pircRBd3	.1754	+15	.3760	+17	.0065	+44	.3878	-4	.5220	+0	.0540	-33
Query size: all sections of topic												
pircRBa1	.1796	*	.3640	*	.0136	*	.4405	*	.5440	*	.0716	*
Web-assisted runs												
pircRBa2	.1854	+3	.4000	+10	.0135	-1	.4369	-1	.5760	+6	.1062	+48

Table 2: Comparing Web-Assisted to Basis Retrieval – Training and Testing Sets

Run ID	Median AP Best	(>/=/<)	Worst	MAP	% no-relv-@10	area	Worst25% MAP
pircRBd1	2	64/3/33	1	0.2774	16%	0.0122	0.0310
pircRBd2	**1**	**74/2/24**	**0**	**0.2900**	**8%**	**0.0219**	**0.0478**
pircRBd3	4	73/1/26	2	0.2816	9%	0.0165	0.0418
pircRBa1	11	79/0/21	0	0.3101	9%	0.0203	0.0467
pircRBa2	**7**	**86/2/12**	**0**	**0.3111**	**6%**	**0.0290**	**0.0622**

Table 3: Comparing PIRCS 100-Topic Results with Median

Compared to all submissions, our results perform very favorably. Table 3 shows the comparison with median AP values. For example, the web-assisted pircRBd2 average precision has 74 topics better than median, 2 equal and 24 worse. One of the 74 has best average precision and none has worst. PircRBa2 is even better.

2 HARD Track

'HARD' (High Accuracy Retrieval from Documents) is a new 2003 extension to previous ad-hoc retrieval experiments. Its purpose is to study the effects of user feedback and metadata on retrieval effectiveness. After a first round of retrieval by a search engine ('Basis Retrieval'), the system is allowed to solicit user feedback by creating a 'Clarification Form' concerning the topic. Users are allowed three minutes time per topic to answer questions presented in the form. Afterwards, the system is able to make use of the form data, as well as further on-topic metadata that is provided, in order to improve on the Basis Retrieval.

2.1 Basis Retrieval

We employ our standard PIRCS ad-hoc processing and retrieval to provide first-round results for the user. This involves an initial retrieval plus a pseudo-relevance feedback (PRF) processing using 20 top documents and 60 best terms (20d60t). This 2-stage retrieval is called pircHDBt1, and is our 'Basis Retrieval' results in this HARD track environment. This involves only the title section of a topic as query. Another basis retrieval using both title and description sections as query is also submitted and denoted as: pircHDBtd1. We have also captured the above ad-hoc processing using only the first stage retrieval. Their retrieval results: pircHDBt0 and pircHDBtd0 are not submitted. An alternative is to use the first stage retrieval as basis retrieval. This saves second stage retrieval time, provides data faster for the user, but the basis would generally not be as high since PRF usually brings higher average precision values.

2.2 Clarification Form Design

After the Basis Retrieval of the previous section, three clarification forms are generated automatically and denoted as: C1 (submitted name QCSU1), C2 (QCSU2) and C3 (QCSU3). The general layout of our clarification form consists of three sections for users to make relevance judgment: i) candidate *related terms*, and ii) candidate *related document* titles or first sentences; iii) user *keyword input*. Each related term or document is associated with a radio button for clicking 'yes ', i.e. relevant. C1 makes use of WordNet [4] to obtain related terms to a query and display them in the clarification form. Clarification form C1 also does not display any related documents, and hence it does not rely on any retrieval and therefore less costly. C2 needs retrieval processing to define top documents and best terms for display. C2 makes use of the title section of a topic only for the initial retrieval. C3 is similar to C2, but uses both title and description sections of a topic for retrieval. The keyword input section is a scrollable. Since a user has only 3 minutes to complete a form, he or she would not have much time left for keyword input even though the scrollable window allows for the space.

For C1, we employ the title section of a topic and define each consecutive two-word as a phrase. Each phrase is passed to WordNet to pick up synonyms, and then the single words. The synonyms obtained are displayed in the 'related term' section of C1 for the user to judge. Some topics may fail to pick up synonyms; for these the related term section may be blank or the query words themselves. Users moreover can type in any words they deem important for a topic in the keyword input section of the clarification form. An example is Topic Hard-044 "Amusement Park Safety". The phrase "amusement park" gets Wordnet to return the following synonyms: "amusement park, funfair, pleasure ground". The last two phrases would not be obtained if words were used individually. Thus, the single words "amusement" gets synonyms "entertainment, amusement", and "park" gets "park, parkland, commons, common, green, ballpark, Mungo park, parking lot, car park, parking area". Since a user is present to judge these terms, presence of noise terms is tolerable.

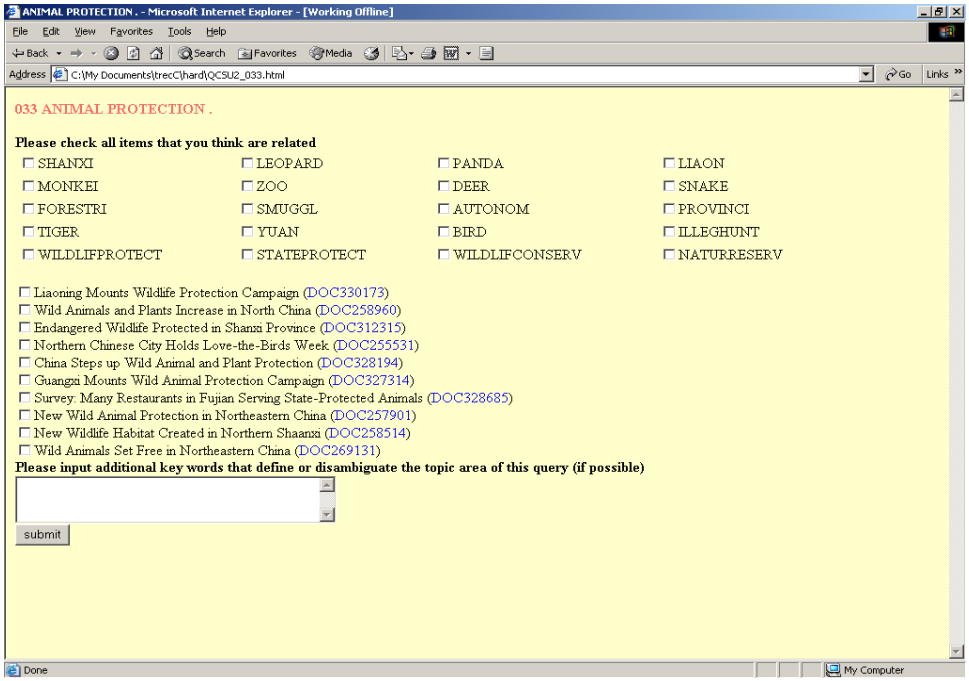

Fig.1: C2 Clarification Form for Query Hard-033

For C2 (title) and C3 (title + description), the related document section comes from the title or first sentence of the 10 top-ranked documents of a 2-stage ad-hoc retrieval. (Initially our design was to use the top documents from an initial retrieval. However, after the conference we discovered that an un-intended

mix-up of files actually led to the use of the 2nd stage results instead). 20 top-ranked terms are also displayed in the related term section. An undesirable situation arises because some of the feedback terms are actually stems not regular words. Porter's stemming algorithm has been employed, and the process is irreversible. Some other terms may be a combination of two stems into one string, which are the result of our adjacent two-word phrase indexing. These may be useful as indexing terms but not suitable for user browsing. We hope to remedy these situations in future enhancements. The user can click on whichever term or document they think is useful for a topic. It is possible that, for some difficult queries, none of the suggested terms or documents is related. However, the keyword input section is available for the user to type in additional words so as not to get frustrated. We believe that a user can complete the clarification form – click through 30 items and input some key words -- in three minutes time. An example of a C2 form is shown in Fig.1.

2.3 Final Retrieval – Document Level

The system has to decide on how to make use of the clarification form data to do further processing. Our strategy is to employ the 'user-clicked' related terms and the 'keywords' typed by the user to expand on the original query (either the title section or the title + description of each topic). Typed words can have typographic errors. We use Google's spell-check facility to remedy the situation. Afterwards, these keywords have to be stemmed to be compatible with other existing index terms. Repeated mention of the same term is kept to provide higher weight. Each expanded query is used for a fresh full 2-stage retrieval. However, during pseudo-relevance feedback (except for C1 form), the 'user-clicked' documents are guaranteed to be among the 20 feedback documents used. Term expansion is still kept at 60. These procedures provide the submissions pircHDC1t1, pircHDC2t1 and pircHDC3td1. It is to be noted that because of time constraints on the part of the assessors, 4 topics in C3 (036, 048, 053, 105) were not filled in. Results of these queries default back to those of the Basis Retrieval.

For clarification forms C2 and C3, we have two further submissions: pircHDC2t2 and pircHDC3td2. In many queries, the related document section receives very few or no 'clicks'. We used a threshold of less than 3. This suggests evidence that the Basis Retrieval results are not good (assuming the user can do correct judgment using the title or first sentence of the retrieved document), and may imply that the topic is a difficult one. For these topics (16 for C2: 36, 59, 87, 115, 117, 124, 154, 177, 180, 186, 187, 220, 226, 228, 231, 235 and 13 for C3: 59, 87, 117, 154, 177, 194, 203, 215, 217, 220, 228, 231, 235), we disable the 2nd stage retrieval during final retrieval and used the initial retrieval results instead. The idea is that quite often for difficult topics, 2nd stage retrieval can lead to worse results compared to initial retrieval.

2.4 Final Retrieval – Phrase Level

After the clarification data has been filled in, additional information in the form of metadata such as: purpose of retrieval, document genre wanted, user familiarity with topic, granularity of result, sample of relevant texts are also released concerning each topic. We only focus on the granularity metadata which can have values: document, passage, sentence and any. There are 16 (18-2 removed) queries that have the requirement of granularity = passage or sentence. For these, each of their retrieval lists is passed to our PIRCS-QA system (see also Section 3) for further processing to try to isolate a small text extent as answer for the topic. Our QA system can be summarized as follows [5]:

1) returning n top-ranked subdocuments from PIRCS retrieval using a query with stemming and stop-word removal;
2) scoring and returning top-ranked sentences from the subdocuments with respect to the general context of the question keywords using a set of eleven heuristics -- both raw and stemmed words were taken into account;
3) analyzing specific properties of the question to obtain its expected answer types, and assigning one of four functional modules that use keywords, meta-keywords and patterns to detect possible answers and add bonus weights to top-ranked sentences for selection purposes;
4) extracting answer strings of required size from top candidate sentences based on the previous question analysis with rules and heuristics for entity definition or identification.

Instead of evaluating sentences we evaluate paragraphs. They are detected by the </p> tag or blank line or an indented line. Since more than one result was submitted and the order is important, we increased the retrieval score bonus. A document-offset bug which was present in our QA track this year was also fixed. The run-id's with the phrase processing are identified with a 'p' at the end like: pircHDC2tp.

2.5 HARD Track Results and Discussion

2.5.1 Results of Document-Level Evaluation

Table 4 below shows results of our submitted and some of the un-submitted runs. Each column has "%imp" denoting "% improvement" from the Basis Retrieval. Relevance judgment is of type 'hard' where partially relevant documents are excluded as relevant. 'Soft' type judgment results that include partially relevant documents are shown in Table 5. The following TREC measures are tabulated: Rel.Ret (number of relevant documents within the top-ranked 1000), MAP (mean average precision over the 48 queries), Pnn (average precision at nn retrieved documents where nn = 10, 20 and 30), and RPre (average precision at rr retrieved where rr = the exact number of relevant documents for each query).

Concerning the short, title only query (average 2.8 terms) results, Table 4 shows that our submitted Basis Retrieval (pircHDBt1) is substantially better than the un-submitted first stage retrieval (pircHDBt0) without PRF, and provides a much higher basis to compare with the final retrievals using clarification or metadata. The queries using Wordnet-based clarification forms (C1) for expansion have average 11.0 terms, and the final retrieval pircHDC1t1 improves over basis 3 to 11% in various measures except for P30 with a decrease of 1%. Using forms C2, the queries have average 12.6 terms and the pircHDC2t1 result improve over basis from 7 to 15% except for Rel.Ret with a decrease of 2%. Looking at MAP values, pircHDC1t1 employing the less costly C1 forms leads to slightly higher performance compared to pircHDC2t1. However, pircHDC2t1 has better effectiveness in low-recall high-precision retrieval region, achieving double-digit improvements for P10-30 over the basis; pircHDC1t1 has more erratic performance: from 10% increase in P10 to 1% decrease in P30. C1 Wordnet only suggests synonyms of the different senses of a word and for 21 queries did not suggest new word. C2 always have some suggested terms that may be related, not necessarily synonyms. It seems that the three minutes spent by a 'user' can bring out significant precision improvements (>5%) over the basis retrieval.

Since there are two sources (related terms and keyword input) of user feedback to augment the original title query, we investigate to see which source is more useful and whether both are necessary. The un-submitted runs ~pircHDC1t1term and ~pircHDC1t1key show results using either the clicked related terms or the typed keywords only. Each leads to an average of 4.8 and 9.2 query terms respectively. Similarly for C2 runs ~pirHDC2t1term (7.5 terms) and ~pircHDC2t2key (8.3 terms). Thus, related terms from Wordnet provide on average only 4.8-2.8=2 relevant words while PRF provides 7.5-2.8=4.7. This reinforces previous experience that general purpose thesauri often miss the query words or may not contain the right sense of query terms (for a user to click). In both C1 and C2 cases, use of only one source of feedback data performs worse than using both, and often worse than the basis values. When both sources are used, the original title, input keywords and clicked related terms often overlap. This is equivalent to user weighting some good terms higher, and may contribute to better results. Comparing pircHDC1t1term with pircHDC1t1key rows, the former is uniformly worse, leading us to conclude that Wordnet supplied terms are less useful than typed keywords. On the other hand, comparing pircHDC2t1term and pircHDC2t1key, the former has better results except for Rel.Ret, leading us to believe that PRF supplied terms are more useful than typed keywords. Different forms have different keywords typed, probably by different users.

Our submitted run pircHDC2t2 that disables PRF when user clicks fewer than 3 relevant documents, was in error due to a use of wrong files. The corrected run is shown as ~pircHDC2t2. It is a bit worse compared to pircHDC2t1 and the procedure is not effective. Of the 15 queries affected, only 5 is better to ignore PRF processing or little change. The additional user-typed keywords may make a query better. It is also qualitatively similar in the case of using title+description queries (pircHDC3td2).

Overall, results for the longer title+description queries (9.2 terms average) improves slightly over the title only run: e.g. the basis run pircHDBtd1 MAP value of .3277 is better by 2% compared to pircHDBt1 title

basis value of .3219. The run with clarification form C3, pircHDC3td1 with average of 18 query terms, improves over the basis pircHDBtd1 between 8 to 15% in all measures except for a 1% decrease in Rel.Ret. As in title runs, the effect of using either the clicked related terms (un-submitted ~pircHDC3td1term, average 15.5 terms) or the typed keywords (~pircHDC3td1key, average 14.3 terms) is to depress performance compared to using both. Also, clicked terms are preferred over user input keywords as in the title only results. This run provides the best overall MAP value of 0.3604 and R-Pre of 0.3875 for all submitted runs.

Our official title run pircHDC2t1 compares favorably with other submitted runs, with 37 queries (1 best) above median, 10 below median (1 worst) and 1 equal to median average precision. Table 5 provides results of soft evaluation, i.e. when partially relevant documents are also treated as relevant. Behavior is similar to that of hard evaluation in Table 4.

Run-ID	Rel.Ret %imp	MAP %imp	R-Prec %imp	P10 %imp	P20 %imp	P30 %imp
hard criteria evaluation (48 queries**;** total 5123 relevant documents**)**						
Query size = title						
No clarification form used (average query size 2.8 terms)						
~pircHDBt0	3482 -11	.2170 -33	.2558 -26	.3500 -21	.3094 -27	.2917 -29
pircHDBt1	3893 *	.3219 *	.3460 *	.4417 *	.4229 *	.4132 *
Clarification form with Wordnet (average query size 11.0 terms)						
pircHDC1t1	3999 +3	.3583 +11	.3740 +8	.4854 +10	.4344 +3	.4111 -1
Using clicked Wordnet terms only (average query size 4.8 terms)						
~pircHDC1t1term	3766 -3	.2995 -7	.3206 -7	.4292 -3	.3792 -10	.3625 -12
Using input keywords only (average query size 9.2 terms)						
~pircHDC1t1key	3802 -2	.3197 -1	.3437 -1	.4604 +4	.4083 -3	.3958 -4
Clarification form with PRF data (average query size 12.6 terms)						
pircHDC2t1	3812 -2	.3536 +10	.3717 +7	.5083 +15	.4802 +14	.4535 +10
Using clicked PRF terms only (average query size 7.5 terms)						
~pircHDC2t1term	3507 -10	.3021 -6	.3242 -6	.5042 +14	.4635 +10	.4236 +3
Using input keywords only (average query size 8.3 terms)						
~pircHDC2t1key	3564 -8	.2900 -10	.3079 -11	.4437 +0	.4083 -3	.3785 -8
pircHDC2t2	3589 -8	.3048 -5	.3191 -8	.4542 +3	.4354 +3	.4125 -0
~pircHDC2t2	3791 -3	.3469 +8	.3648 +5	.5063 +15	.4812 +14	.4535 +10
Query size = title + description						
No clarification form used (average query size 9.2 terms)						
~pircHDBtd0	3606 -9	.2719 -17	.3075 -8	.4417 -9	.3875 -7	.3424 -9
pircHDBtd1	3958 *	.3277 *	.3360 *	.4875 *	.4167 *	.3743 *
Clarification form with PRF data (average query size 18.0 terms)						
pircHDC3td1	3915 -1	.3589 +10	.3813 +13	.5271 +8	.4802 +15	.4306 +15
Using clicked PRF terms only (average query size 15.5 terms)						
~pircHDC3td1--term	3766 -5	.3388 +3	.3574 +6	.4875 +0	.4323 +4	.4049 +9
Using input keywords only (average query size 14.3 terms)						
~pircHDC3td1--key	3699 -7	.3033 -7	.3169 -6	.4604 -6	.3937 -6	.3632 -3
pircHDC3td2	3901 -2	.3604 +10	.3875 +15	.5146 +6	.4740 +14	.4236 +13

Table 4: 'HARD' Retrieval with Hard Evaluation (~ denotes un-submitted data)

Run-ID	Rel.Ret %imp	MAP %imp	R-Prec %imp	P10 %imp	P20 %imp	P30 %imp
soft criteria evaluation (48 queries; total 7576 relevant documents**)**						
Query size = title						
~pircHDBt0	4938 -8	.2548 -30	.2893 -25	.4460 -17	.4140 -21	.3833 -26
pircHDBt1	5372 *	.3650 *	.3857 *	.5396 *	.5260 *	.5167 *
pircHDC1t1	5533 +3	.4069 +11	.4250 +10	.5979 +11	.5469 +4	.5299 +3
pircHDC1t1term	5251 -2	.3449 -6	.3703 -4	.5375 +0	.4958 -6	.4764 -8
pircHDC1t1key	5202 -3	.3585 -2	.3882 +1	.5646 +5	.5094 -3	.4937 -4
pircHDC2t1	5215 -3	.3986 +9	.4242 +10	.6500 +20	.6104 +16	.5799 +12
pircHDC2t2term	4726 -12	.3188 -13	.3512 -9	.6021 +12	.5479 +4	.5035 -3
pircHDC2t2key	4857 -10	.3191 -13	.3501 -9	.5438 +1	.5208 -1	.4868 -6
pircHDC2t2	4891 -9	.3314 -9	.3454 -10	.5583 +3	.5323 +1	.5062 -2
~pircHDC2t2	5201 -3	.3902 +7	.4156 +8	.6479 +20	.6094 +16	.5771 +12
Query size = title + description						
~pircHDBtd0	5069 -7	.3037 -17	.3387 -14	.5600 -5	.4900 -9	.4473 -7
pircHDBtd1	5430 *	.3656 *	.3932 *	.5875 *	.5365 *	.4826 *
pircHDC3td1	5470 +1	.3934 +8	.4131 +5	.6271 +7	.5896 +10	.5403 +12
PircHDC3td1term	5203 -4	.3667 +0	.3926 -0	.5771 -2	.5281 -2	.4979 +3
PircHDC3td1key	5118 -6	.3363 -8	.3532 -10	.5667 -4	.5010 -7	.4708 –2
pircHDC3td2	5445 +0	.3937 +8	.4133 +5	.6167 +5	.5865 +9	.5319 +10

Table 5: 'HARD' Retrieval with Soft Evaluation (~ denotes un-submitted data)

2.5.2 Results of Phrase-Level Evaluation

Table 6 shows results of our passage retrieval. The official evaluation measures are P10, R-Pre and F-measure at 30 documents retrieved (F30). As discussed before, run-id's that end with 'p' undergo special passage processing and return a passage list, i.e. document id with a text extent. Other runs without 'p' return document id lists only. We may consider them as document id with a text extent equal to the whole document length, and each document id contributes one retrieval result only. Errors were also discovered for the pircHDC2tp and pircHDC3tdp runs: there were 18 queries out of 42 that went through our QA

Run-ID	P10 %imp	R-Prec %imp	R30 %imp	P30 %imp	F(30) %imp
hard criteria evaluation (42 queries)					
Query size = title					
pircHDBt1	.2809 *	.1810 *	.2359 *	.2491 *	.1491 *
pircHDC1t1	.3152 +12	.2335 +29	.2724 +15	.2369 -5	.1479 -1
pircHDC1tp	.3770 +34	.3195 +77	.1839 -22	.3081 +24	.1403 -6
pircHDC2t1	.3209 +14	.2145 +19	.2501 +6	.2766 +11	.1549 +4
pircHDC2tp	.3754 +34	.2508 +39	.1426 -40	.3191 +28	.1269 -15
~pircHDC2tp	.3829 +36	.2595 +43	.1762 -25	.3336 +34	.1423 -5
Query size = title+description					
pircHDBtd1	.3186 *	.1699 *	.2404 *	.2316 *	.1280 *
pircHDC3td1	.3359 +5	.2141 +26	.2922 +22	.2374 +3	.1452 +13
pircHDC3tdp	.3353 +5	.2555 +50	.1746 -27	.2772 +20	.1283 +0
~ pircHDC3tdp	.3438 +8	.2575 +52	.1812 -25	.2797 +21	.1294 +1

Table 6: 'HARD' Passage Retrieval with Hard Evaluation (~ denotes un-submitted data)

processing. The rest were supposed to be the document-level retrieval from the basis pircHDC1t1 and pircHDC3td1 respectively; however we erroneously used the first stage retrievals pircHDC2t0 and pircHDC3td0 instead. The corrected runs are shown in Table 6 as ~pircHDC2tp and ~pircHDC3tdp.

It is seen that for all passage runs with run-id 'p', precision values are high, but recall and F(30) values are low compared to the basis. One reason precision values are high is that precision calculation favors shorter passages than whole documents. One reason recall values are low is that relevant documents always cover all relevant materials, especially if it has multiple relevant passages. Comparing pircHDC1t1 (which has Wordnet form C1 feedback) with the basis pircHDBt1, we see P10 improves but F(30) decreases by 1% due to individual precision and recall values at 30 retrieved. Looking at pircHDC1tp where the retrieved list are passage-level, precision values improve over basis probably also because relevant passages are promoted earlier. However, recall values at 30 retrieved are low leading to decreases in F(30).

3 Question-Answering Track

The Internet is a great storehouse of information and facts. Millions search it daily and use it to solve their information needs. It is not surprising therefore that a number of participants in the QA track make use of it as a source of knowledge. This year we join this trend and extend our QA system to use results of the Google search engine.

We extract answers from Google retrievals in two different ways. For certain question types we developed reliable patterns, which identify the answer term. For other questions we use the most frequent word from the snippets returned by Google. For questions that require named entity recognition we make use of Minipar's NE capability.

We submitted three runs for the passage track. pircsQA1 is virtually identical to our QA system for the 2001 QA track. It uses the top 100 documents retrieved by our PIRCS search engine. It combines probabilistic IR methods with search pattern recognition to select the highest-ranking sentence.

PIRCS does not always return documents with the answer and sometimes the document with the correct answer is ranked very low. To remedy this situation we have merged the original query submitted to PIRCS with possible answer extracted from a Google answer snippets. pircsQA2 uses the top 100 documents created by this retrieval. pircsA3 makes use of a different strategy to utilize suggested answers from Google: extra bonus is added to sentences that contain them.

The official results are quite low, due to a bug in the system, which caused the document offsets to be calculated incorrectly. Here we report our own unofficial evaluation, for the 382 queries which had answers in the document collection.

	Score	% improve
pircsQA1	0.249	
pircsQA2	0.264	6.32%
pircsQA3	0.338	35.79%

Table 7: Unofficial results for 382 queries with answers.

pircsQA1 performed worse than in 2001, which indicates that the queries are getting harder. pircsQA2 shows that results can be improved by enhancing the query submitted to the front end search engine, but not by much. The greatest improvement comes from searching the test collection for answers found in the Web. This makes the task somewhat unrealistic, sometimes the answer is available, but we don't tell the user, because it is not in the document collection. Our systems performance can be improved by developing improved patterns.

4 Conclusions

A method of exploiting the WWW to improve ad-hoc retrieval from a target collection was introduced for the Robust Track. This involves forming Boolean-type Google queries from TREC description queries to perform web retrieval, defining alternate queries from returned web pages, and data fusion to define final results. This approach was successful and has improved the worst-query measures substantially from 30% to 80%.

For the HARD Track, clarification forms for each query were designed to solicit relevant term and document information from a user. One type makes use of Wordnet to suggest synonyms to query terms and asks the user to 'click' the relevant ones for query expansion purposes. In addition, users can type in more keywords. Data for this form does not rely on a retrieval and is less costly. A second type of form makes use of retrieval results of the original query, with the top retrieved documents and top related terms presented to the user for feedback. In both cases, results seem to indicate that two inputs, user keywords and 'clicked' terms, are necessary to get improvements compared to results not using these forms. Although document-level results show that forms that rely on a retrieval has a slight edge over the Wordnet forms, passage-level results indicate otherwise.

Acknowledgment

This work was partially supported the Space and Naval Warfare Systems Center San Diego, under grant No. N66001-1-8912, and by a U.S. Govt. DST/ATP contract 2003*H532600*000.

References

[1] Vogt, C.C. and Cottrell, G.W. (1999) Fusion via a linear combination of scores. Information Retrieval, 1(3):151-173.

[2] Kwok, K.L., Grunfeld, L. & Chan, M. (2000) TREC-8 ad-hoc, query and filtering experiments using PIRCS. In: Information Technology: The Eighth Text REtrieval Conference (TREC-8), E.M. Voorhees & D.K. Harman, eds. NIST Special Publication 500-246, US GPO: Washington, DC. pp.217-227.

[3] Lin, D. (1994) PRINCIPAR – an efficient, broad-coverage, principle-based parser. *Proc of COLING-94*. pp.482-488

[4] Miller, G. (1990) Wordnet: an online lexical database. *International Journal of Lexicography*. 3(4)

[5] Kwok, K.L., Grunfeld, L. Dinstl, N & Chan, M. (2001) TREC 2001 Question-answering, web and cross language track experiments using PIRCS. In: Information Technology: The Tenth Text Retrieval Conference, TREC 2001. E.M. Voorhees & D.K. Harman, eds. NIST Special Publication 500-250, US GPO: Washington, DC. pp.452-456.