Asian Language Parsing Evaluated by Hummingbird SearchServer™ at NTCIR-3

Stephen Tomlinson
Hummingbird
Ottawa, Ontario, Canada
stephen.tomlinson@hummingbird.com
November 24, 2002

Abstract

Hummingbird submitted ranked result sets for the Chinese, Japanese and Korean Single Language Information Retrieval tracks of the Cross-Language Retrieval Task of the 3rd NII-NACSIS Test Collection for IR Systems Workshop (NTCIR-3). SearchServer 5.3's segmenter for Asian text, compared to an overlapping n-gram approach, was found to modestly increase precision scores for Japanese, to have a neutral impact for Chinese, and to be detrimental for Korean. SearchServer's option to case normalize Hiragana and Katakana n-grams increased precision substantially for one Japanese query and was of neutral impact for the others. Newline suppression was found to be of only minor benefit for n-gram parsing. Normalizing Han characters to Hangul had almost no effect on the Korean test collection. **Keywords:** *Hummingbird, SearchServer, NTCIR, Chinese, Japanese, Korean, Asian, parsing, n-grams, segmenting.*

1 Introduction

Hummingbird SearchServer[1] is an indexing, search and retrieval engine for embedding in Windows and UNIX information applications. SearchServer, originally a product of Fulcrum Technologies, was acquired by Hummingbird in 1999. Founded in 1983 in Ottawa, Canada, Fulcrum produced the first commercial application program interface (API) for writing information retrieval applications, Fulcrum® Ful/Text™. The SearchServer kernel is embedded in many Hummingbird products, including SearchServer, an application toolkit used for knowledge-intensive applications that require fast access to unstructured information.

SearchServer supports a variation of the Structured Query Language (SQL), called SearchSQL™, which has extensions for text retrieval. SearchServer conforms to subsets of the Open Database Connectivity (ODBC) interface for C programming language applications and the Java Database Connectivity (JDBC) interface for Java applications. Almost 200 document formats are supported, such as Word, WordPerfect, Excel, PowerPoint, PDF and HTML.

SearchServer works in Unicode internally [2] and supports most of the world's major character sets and languages. The major conferences in text retrieval evaluation (NTCIR [4], CLEF [1] and TREC [7]) have provided opportunities to objectively evaluate SearchServer's support for a dozen languages. This paper focuses on evaluating SearchServer's parsing options for Chinese, Japanese and Korean, using the NTCIR-3 Formal Test Collections.

2 Setup

The experiments described in this paper were conducted in August, 2002 with an internal development build of SearchServer 5.3 (5.3.500.279).

2.1 Data

The document sets of the NTCIR-3 Formal Test Collections (Cross-Language Retrieval task) consist of tagged (SGML-formatted) news articles from the 1990's in Chinese, Japanese and Korean. Table 1 gives their sizes. For more information, see the NTCIR web site [4].

2.2 Indexing

A separate SearchServer table was created for each language and parsing option with a SearchSQL statement such as the following:

[1] Fulcrum® is a registered trademark, and SearchServer™, SearchSQL™, Intuitive Searching™ and Ful/Text™ are trademarks of Hummingbird Ltd. All other copyrights, trademarks and tradenames are the property of their respective owners.

Table 1. Sizes of NTCIR-3 Document Sets

Language	Text Size	#Documents
Chinese	555,285,156 bytes	381,681
Japanese	301,019,356 bytes	236,664
Korean	77,510,533 bytes	66,146

```
CREATE SCHEMA NTC3J
CREATE TABLE NTC3J
(DOCNO VARCHAR(256) 128)
STOPFILE 'japancase.stp'
CONSTRUCT_STEMS 'FALSE'
PERIODIC
BASEPATH 'e:\data';
```

The STOPFILE parameter names a file which specifies any non-default parsing options (described below) to apply at index and search-time; the file may also contain stop words, but none were used any for these tasks. The CONSTRUCT_STEMS parameter was set to 'FALSE' to disable stemming of any Latin terms that might have appeared in the text.

Into each table, one row was inserted, specifying the top directory of the library files for the language, and the text reader list. For example:

```
INSERT INTO NTC3J
(FT_SFNAME, FT_FLIST) VALUES
('NTCIR3\Japanese',
 'cTREC/E/d=128:s!
  nti/t=EUC_JP_UCS2:cTREC/p/@:s');
```

The text reader list included the nti (translation) text reader with the appropriate /t option to translate the documents to Unicode format: BIG5_UCS2 for Chinese, EUC_JP_UCS2 for Japanese, and KSC_5601_1992_UCS2 for Korean. The custom text reader called cTREC, originally written for handling TREC document sets [8], handled expansion of the library files of the NTCIR document sets. The cTREC /p option passed through all of the document content unaltered (including the SGML tags, which preferably wouldn't be indexed, but it seemed unlikely to matter for this task).

To index each table, a Validate Index statement such as the following was run:

```
VALIDATE INDEX NTC3J
VALIDATE TABLE;
```

2.3 Parsing Options

SearchServer 5.3 supports two approaches to indexing Asian text: overlapping n-grams and segmentation into words.

By default, SearchServer uses a parser named "unicode" which indexes Asian text using overlapping n-grams (usually bigrams). SearchServer's unicode parser includes Asian specific enhancements such as normalizing old forms of Kanji to new at index and search-time (this normalization is not done at fetch time so that the original form can be viewed). Asian language options of SearchServer's unicode parser include the following:

- Case normalization of Hiragana and Katakana: By default (/c=0), SearchServer treats big and little Hiragana and Katakana characters as different. Specifying /c=1 treats them the same (like upper and lower case Latin characters are normally treated the same). This option is likely to make a difference only for Japanese text.
- Normalizing Han characters to Hangul: By default (/k=0), SearchServer does not map Han characters to Hangul. Specifying /k=1 enables this mapping. This option is only recommended for Korean.
- Newline suppression: By default (/n=0), SearchServer deletes a newline character that follows an Asian character during indexing, and blanks following a deleted new line character are also deleted. Specifying /n=1 disables this feature.

SearchServer 5.3 also includes a parser named "ixasian" which segments Chinese, Japanese and Korean text into words, using Inxight LinguistX Platform 3.3.1. It has a /l option to specify the language for which to optimize.

As mentioned earlier, the parser and options are controlled by the PARSER line in the stopfile. For example, the line

```
PARSER="unicode/c=1"
```

specifies the unicode parser with case normalization of Hiragana and Katakana. Table 2 lists the parsing options evaluated in this paper.

3 Search Techniques

The NTCIR organizers created several "topics": 50 for Chinese and Japanese (which were translations of each other) and 30 for Korean (the Korean news articles covered a different time period than the Chinese and Japanese articles). Each topic contained a "Title" (subject of the topic), "Description" (a specification of the information need, typically one sentence), "Narrative" (more detailed guidelines for what a relevant document should or should not contain, often several sentences), and "Concepts" (typically a list of nouns or noun phrases separated by commas). The participants were asked to use just the Description field for at least one automatic submission per track to facilitate comparison of results.

An ODBC application called QueryToRankings.c was created (based on the example stsample.c program included with SearchServer) to parse the NTCIR topics files, construct and execute corresponding SearchSQL queries, fetch the top 1000 rows, and write out the rows in the results in the requested submission format. SELECT statements were issued with the SQLExecDirect api call. Fetches were done with SQLFetch (typically 1000 SQLFetch calls per query).

3.1 Intuitive Searching

For all runs, SearchServer's Intuitive Searching was used, i.e. the IS_ABOUT predicate of SearchSQL, which accepts unstructured text. For example, if the Title for a topic was "textbook issues in Japan", then a corresponding SearchSQL query would be:

```
SELECT RELEVANCE('V2:3') AS REL,
  DOCNO
FROM NTC3J
WHERE FT_TEXT IS_ABOUT 'textbook
  issues in Japan'
ORDER BY REL DESC;
```

This query would create a working table with the 2 columns named in the SELECT clause, a REL column containing the relevance value of the row for the query, and a DOCNO column containing the document's identifier. The ORDER BY clause specifies that the most relevant rows should be listed first. The statement "SET MAX_SEARCH_ROWS 1000" was previously executed so that the working table would contain at most 1000 rows. Also, the statement "SET VECTOR_GENERATOR '' " was previously executed to disable inflection of any Latin text that might have appeared in the query.

Of course, the actual topics (for the Single Language tracks) were in Chinese, Japanese or Korean, not English. So before running the SearchSQL statements, the appropriate "SET CHARACTER_SET" option was specified ('BIG5' for Chinese, 'EUC_JP' for Japanese, 'KSC_5601_1992' for Korean), so that the query text inside the IS_ABOUT predicate would be converted to Unicode correctly.

The same parser and options are applied to the query text as are applied to the text at index-time, though at search-time some optimizations may be made. For example, in a CONTAINS search (exact phrase matching), when n-grams are used, a minimal set of n-grams may be used for more efficient searching. Before version 5.3, SearchServer also used a minimal set of n-grams for the IS_ABOUT predicate, but as of version 5.3, overlapping n-grams are used with the IS_ABOUT predicate. The official submissions, even though they were made before SearchServer 5.3 was released, included this change. The impact of this change on the NTCIR-3 collections is shown below.

3.2 Statistical Relevance Ranking

SearchServer calculates a relevance value for a row of a table with respect to a vector of words or n-grams based on several statistics. The inverse document frequency of the word or n-gram is estimated from information in the dictionary. The term frequency (number of occurrences of the word or n-gram in the row) is determined from the reference file. The length of the row (based on the number of indexed characters in all columns of the row, which is typically dominated by the external document), is optionally incorporated. The count of the word or n-gram in the vector is also used. To synthesize this information into a relevance value, SearchServer dampens the term frequency and adjusts for document length in a manner similar to Okapi [5] and dampens the inverse document frequency in a manner similar to [6]. SearchServer's relevance values are always an integer in the range 0 to 1000.

SearchServer's RELEVANCE_METHOD setting can be used to optionally square the importance of the inverse document frequency (by choosing a RELEVANCE_METHOD of 'V2:4' instead of 'V2:3'). The importance of document length to the ranking is controlled by SearchServer's RELEVANCE_DLEN_IMP setting (scale of 0 to 1000). For the experiments described in this paper, RELEVANCE_METHOD was set to 'V2:4' and RELEVANCE_DLEN_IMP was set to 500. Different settings were used for some of the official submissions, as described later.

4 Results

The Single Language Information Retrieval tracks of the Cross-Language Retrieval Task were to run the given Chinese, Japanese and Korean queries against document collections in the same language and submit a list of the top-1000 ranked documents to NII for judging (in February, 2002). NII produced a list of relevance assessments for each track: a list of documents judged to be highly relevant, relevant, partially relevant or not relevant for each topic. NII also produced 2 "qrels" files for each track in a format compatible with Chris Buckley's trec_eval program, from which the evaluation measures can be calculated. One qrels file just counted highly relevant and relevant documents as relevant (referred to by the organizers as "rigid" mode) and the other qrels file additionally counted partially relevant documents as relevant ("relaxed" mode).

The evaluation measures are expected to be explained in an appendix of this volume. Briefly: "Precision" is the percentage of retrieved documents which are relevant. "Precision@n" is the precision after n documents have been retrieved. "Average precision" for a topic is the average of the precision after each relevant document is retrieved (using zero as

the precision for relevant documents which are not retrieved). "Recall" is the percentage of relevant documents which have been retrieved. For a set of topics, the measure is the average of the measure for each topic (i.e. all topics are weighted equally).

For topics and languages in which fewer than 3 documents were judged relevant or highly relevant, the organizers discarded the topic for that language. This critera caused 8 of the 50 topics to be discarded for Chinese and Japanese, hence the resulting Formal Test Collections for those languages just contain 42 topics. No topics were discarded for Korean, which remains a 30-topic collection.

4.1 Impact of Parsing Options

Table 2 shows the SearchServer 5.3 parsing options evaluated in this paper and the label assigned to each. The first letter of each label indicates the language (C=Chinese, J=Japanese, K=Korean), and is followed by an indicator of the experiment:

- "Base" indicates the baseline run for each language. The unicode parser (n-grams) is used for the baseline runs. For Chinese, the baseline uses the default options. For Japanese, the baseline specifies /c=1 (case normalization of Hiragana and Katakana). For Korean, the baseline specifies /k=1 (normalization of Han characters to Hangul).
- "Line" indicates that newline suppression is disabled (/n=1).
- "Case" (Japanese only) indicates that case normalization of Hiragana and Katakana is disabled (i.e. /c=1 is not specified).
- "Han" (Korean only) indicates that normalization of Han characters to Hangul is disabled (i.e. /k=1 is not specified).
- "Seg" indicates that the ixasian parser (which segments Asian text into words) is used.

3 runs were done for each parsing option, distinguished by the 3rd part of each label:

- "T" indicates just the Title field of each topic was used
- "D" indicates just the Description field was used
- "C" indicates just the Concepts field was used.

The 3 runs let us check if the parsing options have different impacts on short queries, descriptive queries, and keyword lists.

For each run, scores are shown for both qrels files. The 4th part of the label is "p" when "partially" relevant documents are counted as relevant.

Table 2. Labels for Parsing Options

Label	Parser and Options
C-Base	unicode
C-Line	unicode/n=1
C-Seg	ixasian/l=traditional-chinese
J-Base	unicode/c=1
J-Case	unicode
J-Line	unicode/c=1/n=1
J-Seg	ixasian/l=japanese
K-Base	unicode/k=1
K-Han	unicode
K-Line	unicode/k=1/n=1
K-Seg	ixasian/l=korean

Table 3 shows the precision scores for the baseline runs ("AvgP" for Average Precision, "P@10" for Precision at 10 documents).

For the other parsing options, instead of showing their absolute scores, the difference in their scores from the baseline run in the same language is shown. The "AvgDiff" column is the difference in the precision score; a positive difference means that the run had a higher score than the baseline by the difference shown. The "vs. Base" column shows the number of topics on which the precision score was higher, lower and tied with the baseline's score, respectively (these numbers should always add to 42 for the Chinese and Japanese runs, and add to 30 for the Korean runs).

Table 4 shows the impact of disabling newline suppression on the average precision measure, ordered by decreasing difference in average precision. On these test collections at least, the difference is always minor, though it appears if anything that disabling this feature has a slight negative impact (i.e. it's better to stick with the default and not specify /n=1).

Table 5 shows the impact of disabling case normalization of Hiragana and Katakana (Japanese only) on the average precision measure, ordered by decreasing difference in average precision. The "vs. Base" column shows that for the majority of topics there is no difference, which might mean that the majority of topics do not contain Hiragana or Katakana. For the remaining topics, the differences on average are negative, indicating that case normalization of Hiragana and Katakana may be modestly beneficial on average. This is mostly the result of one topic (topic 14, "computer virus"), for which case normalization helped substantially. However, no topic was substantially hurt by case normalization.

While case normalization should increase recall (more documents match), it could decrease early precision (if the original case was meaningful). Table 6 shows the impact of disabling case normalization of

Table 3. Precision of Baseline Runs

Run	AvgP	P@10
C-Base-T	0.1986	0.2643
C-Base-T-p	0.2393	0.3643
C-Base-D	0.1860	0.2762
C-Base-D-p	0.2257	0.3833
C-Base-C	0.2392	0.3286
C-Base-C-p	0.2784	0.4333
J-Base-T	0.2977	0.3500
J-Base-T-p	0.3600	0.4690
J-Base-D	0.2845	0.3452
J-Base-D-p	0.3264	0.4571
J-Base-C	0.3107	0.3667
J-Base-C-p	0.3513	0.4738
K-Base-T	0.2863	0.3333
K-Base-T-p	0.3176	0.4367
K-Base-D	0.2062	0.2733
K-Base-D-p	0.2437	0.3867
K-Base-C	0.2777	0.3200
K-Base-C-p	0.3313	0.4733

Table 4. Impact of Disabling Newline Suppression on Average Precision

Experiment	AvgDiff	vs. Base
J-Line-T-p	0.0007	4-1-37
J-Line-T	0.0006	3-1-38
C-Line-C-p	0.0003	18-18-6
K-Line-T-p	0.0003	11-16-3
J-Line-D	−0.0000	3-4-35
J-Line-D-p	−0.0000	7-3-32
J-Line-C	−0.0000	5-4-33
J-Line-C-p	−0.0001	4-5-33
C-Line-T-p	−0.0002	16-19-7
K-Line-T	−0.0004	14-11-5
C-Line-C	−0.0005	19-15-8
C-Line-T	−0.0007	20-13-9
K-Line-C	−0.0007	16-10-4
C-Line-D-p	−0.0011	21-19-2
K-Line-C-p	−0.0013	14-15-1
C-Line-D	−0.0017	18-19-5
K-Line-D-p	−0.0018	11-16-3
K-Line-D	−0.0019	12-14-4

Table 5. Impact of Disabling Case Normalization of Hiragana and Katakana on Average Precision

Experiment	AvgDiff	vs. Base
J-Case-C-p	−0.0057	8-7-27
J-Case-T	−0.0068	4-1-37
J-Case-T-p	−0.0070	4-1-37
J-Case-C	−0.0092	8-6-28
J-Case-D-p	−0.0142	4-5-33
J-Case-D	−0.0143	2-6-34

Table 6. Impact of Disabling Case Normalization of Hiragana and Katakana on Precision@10

Experiment	AvgDiff	vs. Base
J-Case-C-p	0.0000	2-1-39
J-Case-C	−0.0048	1-1-40
J-Case-T	−0.0095	0-1-41
J-Case-T-p	−0.0095	0-1-41
J-Case-D	−0.0190	0-3-39
J-Case-D-p	−0.0214	1-3-38

Hiragana and Katakana on the Precision@10 measure. It turns out Precision@10 is also negatively impacted by disabling this option (again, primarily because of topic 14).

On the whole, while it usually makes little difference, it appears there may be an occasional substantial benefit to enabling the /c=1 option for Japanese. However, the above results for average precision and Precision@10 are not statistically significant (at the 5% level by the two-sided Wilcoxon signed rank test), so a larger experiment (more topics) would be needed to be more confident of this result.

Table 7 shows the impact of disabling normalization of Han characters to Hangul (Korean only). The "vs. Base" column shows that the scores for all of the topics are the same (except one, the Concepts field of topic 20, and even the difference for it was slight). This result probably means the Korean documents and topics primarily use Hangul characters and seldom use Han characters.

Table 8 shows the impact of using the segmenter for each language (instead of n-grams) on the average precision measure. The segmenter appears to be modestly beneficial for Japanese, neutral for Chinese, and detrimental for Korean.

Table 9 shows the impact of the segmenter on the Precision@10 measure. Compared to the result for average precision, the beneficial impact for Japanese seems a little higher, the impact for Chinese seems

Table 7. Impact of Disabling Normalization of Han Characters to Hangul on Average Precision

Experiment	AvgDiff	vs. Base
K-Han-T	0.0000	0-0-30
K-Han-T-p	0.0000	0-0-30
K-Han-D	0.0000	0-0-30
K-Han-D-p	0.0000	0-0-30
K-Han-C	0.0000	1-0-29
K-Han-C-p	0.0000	1-0-29

Table 8. Impact of Segmenting on Average Precision

Experiment	AvgDiff	vs. Base
J-Seg-D-p	0.0285	28-14-0
J-Seg-C	0.0203	26-15-1
J-Seg-C-p	0.0175	27-15-0
J-Seg-T	0.0175	23-18-1
J-Seg-T-p	0.0104	23-19-0
J-Seg-D	0.0102	25-17-0
C-Seg-T-p	0.0159	23-19-0
C-Seg-D-p	0.0037	21-21-0
C-Seg-T	−0.0026	20-22-0
C-Seg-D	−0.0107	18-24-0
C-Seg-C-p	−0.0291	19-23-0
C-Seg-C	−0.0390	14-28-0
K-Seg-D	−0.0444	8-22-0
K-Seg-D-p	−0.0603	7-23-0
K-Seg-C	−0.0935	5-24-1
K-Seg-T-p	−0.1087	5-25-0
K-Seg-C-p	−0.1149	3-27-0
K-Seg-T	−0.1162	5-25-0

Table 9. Impact of Segmenting on Precision@10

Experiment	AvgDiff	vs. Base
J-Seg-C-p	0.0524	16-9-17
J-Seg-D-p	0.0524	15-10-17
J-Seg-D	0.0452	18-5-19
J-Seg-C	0.0405	15-10-17
J-Seg-T-p	0.0381	14-9-19
J-Seg-T	0.0262	13-8-21
C-Seg-T-p	0.0143	14-13-15
C-Seg-T	−0.0000	11-10-21
C-Seg-D-p	−0.0024	14-18-10
C-Seg-D	−0.0238	14-17-11
C-Seg-C-p	−0.0524	15-17-10
C-Seg-C	−0.0857	9-22-11
K-Seg-T-p	0.0000	10-13-7
K-Seg-D-p	−0.0100	13-13-4
K-Seg-D	−0.0433	8-11-11
K-Seg-C-p	−0.0467	8-11-11
K-Seg-C	−0.0567	8-12-10
K-Seg-T	−0.0733	4-14-12

more detrimental, and the impact for Korean less detrimental.

Table 10 shows estimators[2] of the impact of segmenting on the average precision measure and 95% confidence intervals[3] for the estimators. For Japanese and Chinese, 5 of the 6 results are not found to be statistically significant at the 5% level, and even in the remaining case, one end of the confidence interval is very close to zero. For Korean, all 6 results are statistically significant at the 5% level (in fact, 5 of the 6 are statistically significant at the 1% level), and in most cases, the end of the interval closest to zero would still represent a noticeable impact.

4.2 Impact of SearchServer 5.3 Change

As mentioned earlier, for the IS_ABOUT predicate, as of version 5.3 the unicode parser produces overlapping n-grams; previous versions, including version 5.0, used a minimal set of n-grams. For example, if the query was ABCD (where the letters represent Asian

[2]The estimator is the Walsh average [3] which, when subtracted from the scores of the segmenter run, maximizes the significance level by the two-sided Wilcoxon signed rank test (for all of our experiments, there was just one such value). This estimator is usually the same as the Hodges-Lehmann Estimator [3] and is less sensitive to outliers than the average difference.

[3]The listed 95% confidence intervals are derived from the range of values for which, when subtracted from the scores of the segmenter run, the significance level by the two-sided Wilcoxon signed rank test is at least 5%. The listed boundary points might not need to be in the interval; the values are rounded so as to enlarge the interval when necessary to ensure the listed interval covers the minimal one.

Table 10. Confidence Intervals for Impact of Segmenting on Average Precision

Experiment	EstDiff	95% Confidence Int.
J-Seg-D-p	0.0388	[0.0058, 0.0658]
J-Seg-C-p	0.0256	[−0.0005, 0.0495]
J-Seg-D	0.0254	[−0.0069, 0.0514]
J-Seg-C	0.0196	[−0.0079, 0.0483]
J-Seg-T	0.0149	[−0.0049, 0.0438]
J-Seg-T-p	0.0144	[−0.0083, 0.0411]
C-Seg-T-p	0.0055	[−0.0129, 0.0494]
C-Seg-D-p	0.0044	[−0.0222, 0.0346]
C-Seg-T	0.0024	[−0.0156, 0.0275]
C-Seg-D	−0.0084	[−0.0384, 0.0160]
C-Seg-C-p	−0.0180	[−0.0519, 0.0067]
C-Seg-C	−0.0280	[−0.0606,−0.0047]
K-Seg-D	−0.0416	[−0.0781,−0.0073]
K-Seg-D-p	−0.0612	[−0.1146,−0.0145]
K-Seg-C	−0.0919	[−0.1390,−0.0322]
K-Seg-T	−0.0958	[−0.1536,−0.0285]
K-Seg-T-p	−0.1093	[−0.1658,−0.0303]
K-Seg-C-p	−0.1111	[−0.1736,−0.0460]

characters), previous versions would likely have just contained AB and CD in the internal query vector, whereas SearchServer 5.3 would likely contain AB, BC and CD. This change substantially increased the scores in internal experiments on the NTCIR-2 Chinese test collection, with little impact on the scores on the NTCIR-1 Japanese test collection. How this change affects the scores on the NTCIR-3 collections is now checked.

Tables 11 and 12 show the impact of going back to minimal n-grams (SearchServer 5.0) on the average precision and Precision@10 measures, respectively. Negative differences would suggest that the change made for SearchServer 5.3 was beneficial. The tables suggest that the change was modestly beneficial for the Description-only queries in Chinese and Korean, but was modestly detrimental in most other cases, though again less so for Description-only queries in Japanese than the other query types.

Table 13 shows estimators of the impact of going back to minimal n-grams (SearchServer 5.0) on the average precision measure and 95% confidence intervals for the estimators (based on the two-sided Wilcoxon signed rank test as described in the earlier footnotes). Generally speaking, the results are consistent with the average differences shown earlier. While some of the results are statistically significant at the 5% level (the confidence interval does not contain zero), none of the estimators are large (at most a couple points).

Table 11. Impact of Minimal N-grams on Average Precision

Experiment	AvgDiff	vs. Base
J-SS5-C	0.0256	23-12-7
J-SS5-T	0.0190	14-10-18
J-SS5-C-p	0.0180	24-11-7
J-SS5-T-p	0.0173	14-9-19
J-SS5-D-p	0.0072	29-11-2
J-SS5-D	0.0008	27-13-2
K-SS5-C	0.0273	18-6-6
K-SS5-C-p	0.0230	20-5-5
K-SS5-T-p	0.0185	11-6-13
K-SS5-T	0.0120	10-7-13
K-SS5-D	−0.0176	9-18-3
K-SS5-D-p	−0.0417	9-18-3
C-SS5-T-p	0.0191	23-17-2
C-SS5-T	0.0128	22-18-2
C-SS5-C-p	0.0052	20-18-4
C-SS5-C	−0.0040	18-21-3
C-SS5-D	−0.0276	12-30-0
C-SS5-D-p	−0.0277	15-27-0

Table 12. Impact of Minimal N-grams on Precision@10

Experiment	AvgDiff	vs. Base
J-SS5-C-p	0.0357	10-6-26
J-SS5-C	0.0310	8-4-30
J-SS5-T	0.0214	7-6-29
J-SS5-T-p	0.0214	6-7-29
J-SS5-D-p	0.0214	12-7-23
J-SS5-D	0.0167	12-7-23
K-SS5-T-p	0.0400	4-4-22
K-SS5-C	0.0367	9-2-19
K-SS5-C-p	0.0333	9-5-16
K-SS5-T	0.0133	4-3-23
K-SS5-D	−0.0467	5-9-16
K-SS5-D-p	−0.0567	7-9-14
C-SS5-C-p	0.0071	15-10-17
C-SS5-T	0.0048	9-9-24
C-SS5-T-p	−0.0024	8-12-22
C-SS5-C	−0.0071	10-10-22
C-SS5-D	−0.0310	8-17-17
C-SS5-D-p	−0.0452	7-17-18

Table 13. Confidence Intervals for Impact of Minimal N-grams on Average Precision

Experiment	EstDiff	95% Confidence Int.
J-SS5-C-p	0.0150	[−0.0003, 0.0338]
J-SS5-D-p	0.0136	[0.0013, 0.0338]
J-SS5-C	0.0113	[0.0000, 0.0333]
J-SS5-D	0.0099	[−0.0050, 0.0253]
J-SS5-T	0.0003	[−0.0001, 0.0213]
J-SS5-T-p	0.0003	[−0.0020, 0.0216]
K-SS5-C	0.0171	[0.0086, 0.0315]
K-SS5-C-p	0.0160	[0.0044, 0.0283]
K-SS5-T-p	0.0007	[−0.0002, 0.0080]
K-SS5-T	0.0001	[−0.0005, 0.0161]
K-SS5-D	−0.0030	[−0.0233, 0.0052]
K-SS5-D-p	−0.0189	[−0.0591,−0.0012]
C-SS5-T-p	0.0049	[−0.0086, 0.0189]
C-SS5-T	0.0034	[−0.0080, 0.0194]
C-SS5-C-p	0.0020	[−0.0084, 0.0150]
C-SS5-C	−0.0014	[−0.0095, 0.0088]
C-SS5-D-p	−0.0208	[−0.0481,−0.0019]
C-SS5-D	−0.0216	[−0.0434,−0.0036]

4.3 Official Submissions

The official submissions in February, 2002 used an older, experimental version of SearchServer, which for Asian languages would give similar rankings as the subsequent commercial release version of SearchServer 5.3. All of the official submissions used n-gram parsing.

For the 3 Description-only submissions, the options used were the same as for the baseline D-only runs of Table 3, including the same parser options ("unicode" for Chinese, "unicode/c=1" for Japanese, "unicode/k=1" for Korean), same RELEVANCE_METHOD ('V2:4') and same RELEVANCE_DLEN_IMP (500). The official scores were almost the same as those listed in the table.

For the 3 Title+Concepts submissions, RELEVANCE_METHOD was set to 'V2:3' and RELEVANCE_DLEN_IMP was set to 0. Also, for the Japanese run, /c=1 (case normalization of Hiragana and Katakana) was not specified.

For the 3 full topic submissions, RELEVANCE_METHOD was set to 'V2:3' and RELEVANCE_DLEN_IMP was set to 1000. Again, for the Japanese run, /c=1 (case normalization of Hiragana and Katakana) was not specified.

No query expansion techniques (such as blind feedback) were applied for any of the submissions (or any of the other experiments described in this paper). All of the query text was used; no attempt was made to identify and discard Asian text corresponding to common instruction words (e.g. "Find relevant documents about"), which usually increases the scores a little.

3 monolingual English runs were donated for the benefit of the cross-language judging pools. Generally, the same topic fields and parameters were used as for the Asian runs. However, the "unicode" parser indexes Latin text using words, not n-grams. English stemming was not enabled (an oversight). Unlike for the Asian runs, stop words (e.g. "the", "by") were not indexed, and common instruction words based on past TREC topics (e.g. "find", "relevant", "documents") were discarded. The English runs are ignored elsewhere in this paper.

The scores for the official submissions are expected to be listed in an appendix of the proceedings.

References

[1] Cross-Language Evaluation Forum (CLEF) web site. http://www.clef-campaign.org/

[2] Andrew Hodgson. Converting the Fulcrum Search Engine to Unicode. In Sixteenth International Unicode Conference, Amsterdam, The Netherlands, March 2000.

[3] Myles Hollander and Douglas A. Wolfe. Nonparametric Statistical Methods. Second Edition, 1999. John Wiley & Sons.

[4] NTCIR (NII-NACSIS Test Collection for IR Systems) Home Page. http://research.nii.ac.jp/∼ntcadm/index-en.html

[5] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford. (City University.) Okapi at TREC-3. In D. K. Harman, editor, Overview of the Third Text REtrieval Conference (TREC-3). NIST Special Publication 500-226. http://trec.nist.gov/pubs/trec3/t3_proceedings.html

[6] Amit Singhal, John Choi, Donald Hindle, David Lewis and Fernando Pereira. AT&T at TREC-7. In E. M. Voorhees and D. K. Harman, editors, Proceedings of the Seventh Text REtrieval Conference (TREC-7). NIST Special Publication 500-242. http://trec.nist.gov/pubs/trec7/t7_proceedings.html

[7] Text REtrieval Conference (TREC) Home Page. http://trec.nist.gov/

[8] Stephen Tomlinson and Tom Blackwell. Hummingbird's Fulcrum SearchServer at TREC-9. In E. M. Voorhees and D. K. Harman, editors, Proceedings of the Ninth Text REtrieval Conference (TREC-9). NIST Special Publication 500-249. http://trec.nist.gov/pubs/trec9/t9_proceedings.html