The HAIRCUT System at TREC-9

Paul McNamee, James Mayfield, and Christine Piatko
The Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723-6099 USA
Paul.McNamee@jhuapl.edu
James.Mayfield@jhuapl.edu
Christine.Piatko@jhuapl.edu

Overview

The Hopkins Automated Information Retriever for Combing Unstructured Text (HAIRCUT) is a research IR system developed at the Johns Hopkins University Applied Physics Laboratory (JHU/APL). HAIRCUT benefits from a basic design decision to support flexibility throughout the system. One specific example of this is the way we represent documents and queries; words, stemmed words, character n-grams, multiword phrases are all supported as indexing terms. This year we concentrated our efforts on two of the tasks in TREC-9, the main web task and cross-language retrieval in Chinese and English.

Small Web Task

For this task we indexed documents using two types of indexing terms, unstemmed words and character n-grams using n=6. Summary information of the two indices is shown in Table 1. The difference in the number of documents is likely attributable to a few documents that contain a single short word from which no six character sequence can be formed. Note that the use of 6-grams greatly increased both the size of the dictionary and the size of the index files. No attempt was made compress our data structures and reduce the amount of disk space required although such techniques have been successful with both words [12] and n-grams [10].

Each document was processed in the following fashion. First, we ignored HTML tags and used them only to delimit portions of text. Thus no special treatment was given for sectional tags such as `<TITLE>` or `<H1>` and both tags and their attribute values were eliminated from the token stream. The text was lowercased, punctuation was removed, and diacritical marks were retained. Tokens containing digits were preserved; however only the first two of a sequence of digits were retained (e.g., 1920 became 19##). The result is a stream of blank-separated words.

When using n-grams we construct indexing terms from the same sequence of words. These n-grams may span word boundaries; an attempt is made to discover sentence boundaries so that n-grams spanning sentence boundaries are not recorded. Thus n-grams with leading, central, or trailing spaces are formed at word boundaries.

Queries were parsed in the same fashion as were documents with two exceptions. On some of our title only runs we attempted to correct the spelling of words that did not occur in our dictionary. Also, we tried to remove stop structure from the description and narrative sections of the queries using a list of about 1000 phrases constructed from past TREC topic statements.

	# docs	# terms	index size
words	1,588,374	3,019,547	2.96 GB
6-grams	1,588,169	19,209,934	36.0 GB

Table 1. Index statistics for the wt10g collection

In all our experiments we used a linguistically motivated probabilistic model. This model, described in a report by Hiemstra and de Vries [2], is essentially the same model that was used by BBN in TREC-7 [9]. The similarity calculation that is performed is:

$$Sim(q,d) = \prod_{t=terms} \left(\alpha \cdot f(t,d) + (1-\alpha) \cdot df(t)\right)^{f(t,q)}$$

Equation 1. Similarity calculation.

where *f(t,d)* is the frequency of term *t* in document *d* and *df(t)* denotes the document frequency of *t*.

After the query is parsed each term is weighted by the query term frequency and an initial retrieval is performed followed by a single round of relevance feedback.

To perform relevance feedback we first retrieve the top 1000 documents. We use the top 20 documents for positive feedback and the bottom 75 documents for negative feedback; however duplicate or near-duplicate documents are removed from these sets. We then select terms for the expanded query. After retrieval using this expanded and reweighted query, we have found a slight improvement by penalizing document scores for documents missing many highly ranked query terms. We multiply document scores by a penalty factor:

$$PF = 1.0 - \left(\frac{\text{\# of missing terms}}{\text{total number of terms in query}} \right)^{1.25}$$

Equation 2. Penalty function for missing terms.

As can be seen in Table 2, we use only about one-fifth of the terms of the expanded query for this penalty function

	# Expansion Terms	# Penalty terms
words	60	12
6-grams	400	75

Table 2. Number of expansion terms and penalty terms by indexing scheme.

Several of our official runs were formed by merging baseline ranked lists of documents, for example, merging a word-based query and a 6-gram based query. We merged separate ranked lists by first normalizing document scores and then linearly combining values from different runs, an approach that was successful for us in TREC-8 [7].

We conducted our work on a 4-node Sun Microsystems Ultra Enterprise 450 server. The workstation had 2.5 GB of physical memory and access to 100 GB of dedicated hard disk space.

Official Results

For the most part we ignored the web-nature of the documents and relied on textual content to rank documents. We did however, try two techniques to boost our content-based runs. Both techniques were motivated by the track guidelines. First, we attempted to exploit hyperlink structure and submitted two runs that used backlink frequency to rerank content-based runs. Secondly, we attempted to correct misspellings in title-only queries.

We submitted six official submissions in the small web track, four of the runs were solely based on document content and the other two were an attempt to utilize backlink frequency information to improve a content-based run.

Three of our four content-based runs differ only in the selection of which parts of the topic statements were used. Thus *apl9t*, *apl9td*, and *apl9tdn* used the title, title and description, and title, description, and narrative sections, respectively. The fourth run, *apl9all* was a combination of the three other runs. A summary of each run's performance on the task is shown in Table 3.

	avg prec	recall	# best	# ≥ median
apl9t	0.1272	1276	0	28
apl9td	0.1917	1535	2	33
apl9tdn	0.1785	1584	1	32
apl9all	0.1948	1609	0	37

Table 3. Content-based runs for the Small Web task.

We were surprised by lower than expected results in the web task. During brief post-hoc analysis of our constituent runs we observed that relevance feedback had an adverse effect on our runs; rather than the 25-30% increase in average precision that we typically find, average precision decreased by roughly 10%. It will require further analysis to discover the cause for this phenomenon. We observe that the mean number of relevant documents per query, 52.3, is lower than past ad hoc TREC tracks and it is possible that this would reduce the benefit normally associated with automated relevance feedback.

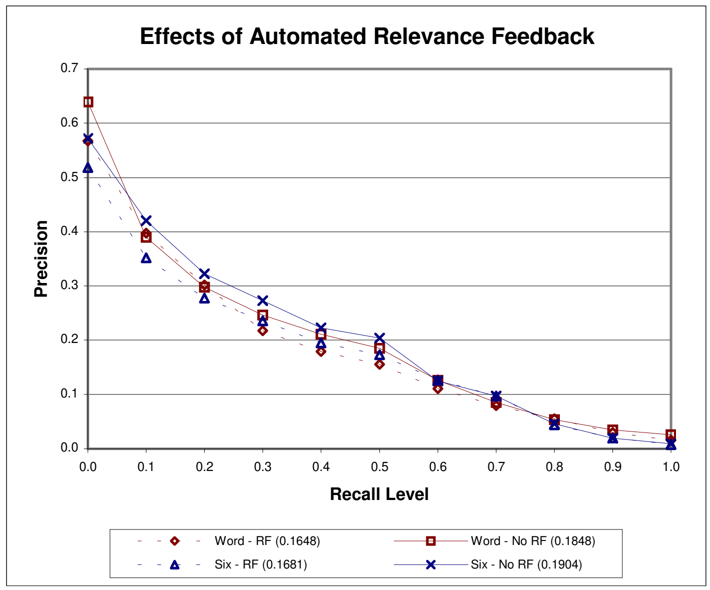

Figure 1. Adverse effects of blind relevance feedback.

Naïve Use of Backlink Frequency

We made a simple attempt to incorporate link frequencies in our results. This was done in a very simple way - we multiplied a document's score in a content-based retrieval by a multiplicative factor derived from backlink frequency and resorted the retrieved documents. The exact computation was:

$$BLFactor(d) = 0.1 + 0.9 \sqrt{\frac{backlinkcount(d)}{MaxBacklinkCount}}$$

Equation 3. *MaxBacklinkCount* is the number of documents that link to the most linked-to document.

Comparing the results in Table 3 and Table 4, it is clear that such a simple attempt to exploit backlink counts is insufficient.

	avg prec	change	# best	# ≥ median
apl9lt	0.1062	- 0.0210	0	25
apl9ltdn	0.1494	- 0.0454	0	26

Table 4. Link-influenced runs corresponding to *apl9t* and *apl9tdn*.

Use of Spelling Correction

If three or fewer documents in the TREC-8 collection contained a topic term, we attempted spelling correction on that term. First, we looked for words occurring in at least five documents that were one insertion, deletion, substitution, or transposition away from the misspelled word. If such a word was found, we used it in lieu of the misspelled word; if more than one such word was found, we selected the one that occurred most frequently (this led us to correct 'tartin' to 'martin' rather than 'tartan'). If no correction was found, we then tried to split the word into two pieces of three characters or more, each of which appeared in at least five TREC-8 documents. If no such pair was found, we left the word uncorrected.

The results of our attempts at spelling correction are shown in the following table:

Topic	Original	avg prec	Correction	avg prec	Change
463	tartin	0.0000	martin	0.0000	0.0000
464	nativityscenes	0.0000	nativity scenes	0.0000	0.0000
474	bennefits	0.0003	benefits	0.0002	-0.0001
476	aniston	0.1517	anniston	0.0062	-0.14.55
483	rosebowl	0.0108	rose bowl	0.3198	+0.3090
487	angioplast7	0.0000	angioplasty7	0.1553	+0.1553

Table 5. Impact of spelling correction.

These results reflect word-based title-only runs with relevance feedback. Spelling correction helped us dramatically on two queries, and hurt us on one.

Cross-Language Task

The TREC-9 CLIR task consisted of bilingual retrieval of Chinese newspaper articles from English queries. A monolingual Chinese-Chinese run was also permitted. This was JHU/APL's first experience with Chinese document retrieval and we learned quite a lot from the experience. Undaunted by our inability to read Chinese, we attempted the task with only an English/Chinese parallel corpus and a minimal knowledge of the Big-5 encoding. Our CLIR experiments focused on two questions, namely, "How do 2- and 3-grams compare as indexing terms in unsegmented Chinese text?" and "Does query translation with parallel corpora perform on par with an available machine translation system?"

Philosophically, we desire to maximize cross-language performance using few language-specific resources. Although segmenters and dictionaries are available for a high-density language such as Chinese, many languages lack these tools. Additionally such resources are rarely in a standard format and the quality of the resource depends greatly on the source.

Though we did perform an experiment indexing only the raw bytes of the collection, on the whole it seemed better to process the Big-5 encoded documents on a character basis. The CJKV text by Ken Lunde was an invaluable aid in our software development [6]. We did not segment the text, and instead elected to index the documents using both 2- and 3-grams. Nie and Ren have previously reported that 2-grams perform comparably with words on the TREC 5/6 Chinese collection and that a combination of both is best [11]. We wanted to assess the use of 3-grams in a straight-up comparison with 2-grams.

We tried translating the topic statements in three different ways, two using a parallel corpus and one using an online machine translation tool. In our monolingual Chinese run we attempted to remove stop structure using translations of our English stop phrases. We used the same linguistically motivated probabilistic model that was used for our English web retrieval. Most of our official runs were produced by combining individual runs using both 2- and 3-grams, an approach that as it turns out, depressed our results.

	# docs	# terms	index size
2-grams	127938	1974077	673 MB
3-grams	127938	15185076	959 MB

Table 6. Index statistics for the TREC-9 Chinese collection.

Translation Using Hong Kong Parallel Corpora

About one month before the CLIR results were due at NIST we observed that we had no in-house method for translating English to Chinese. We quickly obtained two parallel English/Chinese collections from the Linguistic Data Consortium (LDC), the Hong Kong Laws Parallel Text collection [4] and the Hong Kong News Parallel Text collection [5].

The Laws collection contains roughly 310,000 aligned sentences. The News collection contains roughly 18,000 aligned documents. Both collections are encoded in Big-5 which matches the encoding in the TREC-9 Chinese collection.

We built a hybrid collection from the Laws collection and from aligned sections of the News documents. We indexed the collection twice, both with 2-grams and 3-grams. Summary information about these two indices is shown in the following table:

	# docs	# terms	index size
English words	344,299	46,951	105 MB
Chinese 2-grams	343,714	553,358	195 MB
Chinese 3-grams	333,007	2,908,676	270 MB

Table 7. Statistics for APL's hybrid parallel collection.

Official results

We submitted four official runs for the CLIR task, *apl9xmon*, *apl9xtop*, *apl9xwrd*, and *apl9xcmb*, that are described below. Each run is produced by combining multiple base runs. All of the base runs made use of relevance feedback. The number of expansion terms varied depending on the indexing terms; 100 expansion terms were used with the 2-gram index and 400 terms were used with 3-grams.

Our only monolingual submission was *apl9xmon*. This run was produced by combining six base runs, title-only, title + description, and title + description + narrative, using both 2- and 3-grams.

Our first method for query translation followed the approach we used successfully in the CLEF-2000 evaluation [8], namely, pre-translation expansion using highly ranked documents from a document collection in the same language as the source query followed by individual term translation using our parallel collection. Using this approach, the run, *apl9xtop*, was built from two base runs that were produced from 2- and 3-grams. The base runs used queries produced by expanding full topics from documents in the TREC-8 collection.

We were concerned that using the TREC-8 collection as an expansion collection might not be a good idea since it is not contemporaneous with the Chinese collection. We therefore tried a word-by-word translation of the topic statements, also using the parallel collection. The run *apl9xwrd* was produced by combining six base runs (2-, 3-grams; T, TD, TDN queries).

The final run, *apl9xcmb*, was simply a combination of all base runs used in *apl9xtop*, *apl9xwrd*, and the unofficial machine translation run, *apl9xibm*.

	avg prec	recall	# best	# ≥ median	% mono
apl9xmon	0.3085	621	5	20	100 %
apl9xtop	0.0763	360	0	7	24.7%
apl9xwrd	0.1076	416	0	8	34.9%
apl9xcmb	0.1523	535	0	11	49.4%

Table 8. Official results for CLIR task

We wanted to compare translation using our parallel collection to available machine translation. We were not in possession of Chinese MT software in-house so we relied on a web-based translation. The first operational web-based translation service we found was the IBM AlphaWorks server [3]. We had no previous experience with this service or knowledge of its methods or quality; we decided to use it solely based on convenience. The unofficial run, *apl9xibm* was produced from six base runs (2-, 3-grams; T, TD, TDN queries).

Comparing 2-grams and 3-grams

Our decision to submit combined runs using both 2- and 3-grams was based on experience that shows benefit from a combination of multiple, reasonable quality results. As it turns out, our runs using 3-grams performed appreciably worse than those using 2-grams. Average precision and recall for the monolingual base runs used in *apl9xmon* are shown in Table 9.

It seems clear that 2-grams are preferable to 3-grams, at least on a collection of this size. This trend seems to hold both in monolingual retrieval with natural language queries and in bilingual retrieval using word-based 'translations'. We created a post-hoc monolingual run using only the 2-grams and saw average precision increase from 0.3085 in *apl9xmon* to 0.3339, an 8.2% increase.

		avg prec	recall
2-grams	T	0.2926	606
	TD	0.3154	622
	TDN	0.3333	624
3-grams	T	0.1991	572
	TD	0.2170	571
	TDN	0.2368	555

Table 9. Comparing 2- and 3-grams using monolingual queries.

A previous study by Chen et. al. [1], examined the relative merits of 1-, 2-, and 3-grams (as well as several other methods of indexing) using the TREC-5 Chinese collection. Though the data, character encoding, and retrieval model differ from this present study, the relative performance between 2-grams and 3-grams is quite similar for several metrics. On automatic long queries they report average precision of 0.3677 for 2-grams and 0.2405 for 3-grams, a performance ratio of 1.529; from values in Table 9 we compute a comparable ratio of 1.408. Looking at relevant documents retrieved we report a ratio of 1.123 to their 1.162.

Performance of Different Translation Schemes

Another thing we wanted to examine was the effect of using different query translation methods. Our three methods achieved similar performance. Rather than compare the combined runs, we instead look at the constituent base runs. The following tables reveal the performance achieved by each run and its relative performance to *apl9xmon*. For each strategy the best performance was observed when 2-grams were used on full-length topic statements.

		avg prec	recall	% mono
2-grams	TDN	0.1175	341	38.1%
3-grams	TDN	0.0261	237	8.46%

Table 10. Bilingual results using pre-translation expansion (topic expansion)

		avg prec	recall	% mono
2-grams	T	0.1036	409	33.6 %
	TD	0.1214	455	39.3 %
	TDN	0.1261	461	40.9%
3-grams	T	0.0464	254	15.0%
	TD	0.0440	309	14.3%
	TDN	0.0245	244	7.94%

Table 11. Bilingual results using individual word translation

		avg prec	recall	% mono
2-grams	T	0.0674	385	21.8%
	TD	0.1017	487	33.0%
	TDN	0.1284	517	41.6%
3-grams	T	0.0512	305	16.6%
	TD	0.0774	335	25.1%
	TDN	0.0773	374	25.1%
apl9xibm		0.1000	497	32.4%

Table 12. Bilingual results using IBM's AlphaWorks Translator

The performance achieved by each of the translation methods was very similar. The precision-recall graph in Figure 2 shows the performance of each query translation scheme using 2-gram indexing and full topic statements. The graph shows that while the average precision using each method is nearly the same, the AlphaWorks translator performs slightly better at the high-precision part of the curve.

None of the bilingual runs achieves comparable performance to the monolingual run and our best official bilingual submission, *aplxcmb* only achieves performance of 49.4% of our official monolingual run, *apl9xmon*. This is significantly lower percentage than the 70-80% we obtained in our experiments with the CLEF-2000 workshop that was devoted to European languages [8].

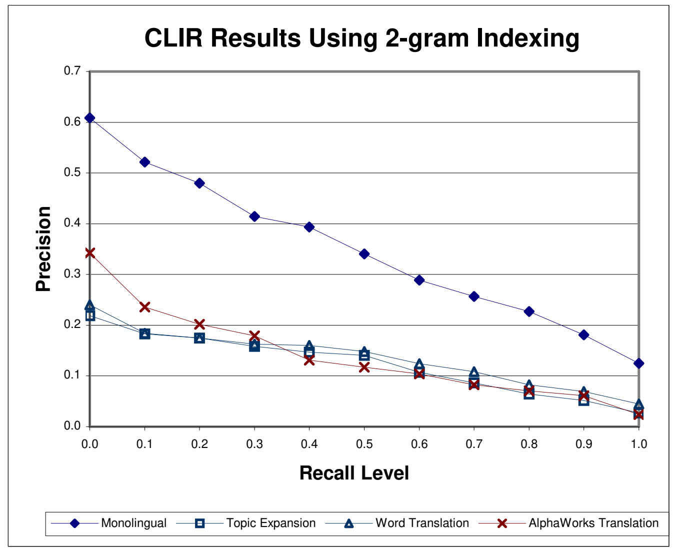

Figure 2. Precision-recall curve for CLIR runs

See the following page for an example of the query translations we used.

Topic CH73

Official English Query

<title> AIDS in China

<desc> Description:
Find documents that report on the number of cases of AIDS in China, the names and locations of AIDS research and treatment facilities in China, and the number of deaths per year attributed to AIDS in China.

<narr> Narrative:
Documents that quote specific total numbers or percentages for people diagnosed with AIDS in China are relevant. Documents containing the official names and/or locations of China's research and treatment facilities are relevant. Documents revealing China's total number of fatalities per year due to AIDS are relevant.

IBM AlphaWorks Translation

<title> 在中國

<desc>描繪中有幫助:爲檔案找到那份關於在中國中的愛滋病的情況的數目,名字和位置愛滋病研究和處理設施在中國中的,和每被把年歸於在中國中的愛滋病的死的數目的報告 .

<narr>敘事:爲用在中國中的愛滋病被診斷的人們引用特有總數或者百分比的檔案是有關.含有官方名字和或位置中國的研究和處理設施的的檔案是有關.暴露中國的總每由於愛滋病年的死亡的數目的檔案是有關.

English word	Top 2-gram	Top 3-gram
aids	愛滋	愛滋病
china	中華	華人民
cases	況下	情況下
research	研究	的研究
number	數目	的數目
treatment	治療	熱處理
hiv	滋病	愛滋病
deaths	生死	生死登
total	的總	不得超
diagnosed	診斷	生署
prevention	防止	《防止
health	生	生主
official	產管	產管理
chinese	英文	中英文
numbers	號碼	覆:
carriers	散貨	／散貨
infected	感染	受感染
provinces	轄市	直轄市
intravenous	二年	大卑斯
disease	疾病	傳染病
county	縣級	縣級以
beijing	北京	在北京
education	教育	教育
reclassification	none	none
virus	病毒	病毒
spread	蔓延	生署
patient	病人	生署
adolescents	青少	青少年
dept	線協	理受影
yunnan	雲南	雲南省
tracy	none	none
ivdu	none	none
mainland	內地	國內地
infection	感染	生署
indicates	顯示	本標誌
foreigners	外籍	外籍人
regions	地區	自治區
spreading	擴散	（星期
risk	風險	的風險
reported	星期	（星期
publicity	宣傳	傳活動
angeles	洛杉	洛杉磯
adults	成人	成年人
los	洛杉	洛杉磯
characteristics	特徵	統計調
facilities	設施	的設施
drug	藥物	危險藥
monitoring	監察	（星期
thomas	星期	日（星
medical	醫生	冊醫生
negative	負面	影響
control	控制	或控制
table	省覽	交立法
discovered	發現	聞公布
ministry	交部	外交部
causes	導致	或導致
december	二月	2 月
cities	城市	香港的
chen	教授	（星期
minzhang	none	none

Figure 3. Two query translation methods are compared. The original English version of topic CH73 is shown along with the results of the IBM AlphaWorks translator. In the table on the right the query used in apl9xtop is partially displayed. The first column contains the best sixty terms produced by searching the TREC-8 ad hoc English documents using the official English version of topic CH73. The second column contains the top-ranked 2-gram extracted from our parallel collection; the third column contains the top-ranked 3-gram. During retrieval the top three 2-grams and the top 10 3-grams were used; however, only the top term is shown here due to space constraints.

Conclusions

This year we participated in two tracks that each presented new challenges.

In the small web task, we focused on content-based methods and tried two techniques to 'accommodate' the web-nature of the task. The first technique was a rudimentary use of backlink counts that proved too simplistic to be beneficial. The second technique, spell correcting misspelled short queries was generally beneficial, however it backfired in certain instances. We found automated relevance feedback to have a deleterious effect on our performance, a finding that warrants further investigation.

Though our team is experienced in cross-language retrieval, we had no experience in Asian language retrieval. We started the Chinese task with no ability to read Chinese and no language resources such as segmenters or dictionaries to draw on. Due to time constraints we were unable to make use of the TREC-5/6 training data and thus we entered the task relatively unprepared. We relied on our general experience using n-grams as indexing terms, a quickly acquired knowledge of the Big-5 encoding, and an English/Chinese parallel collection.

From our experience in the CLIR track we draw the following lessons. First, 2-grams are preferable to 3-grams for indexing Chinese. We remain open to the possibility that other techniques may be better still – for example, using both 2-grams and 3-grams, or 2-grams and segmented words. Our second observation is that corpus-based translation is a viable alternative to extant machine translation software. However, our present results in English to Chinese, bilingual retrieval seem to fall well short of Chinese monolingual retrieval. Now that we have some experience in Chinese text retrieval and a training collection to draw from, we will endeavor to refine our methods to narrow this gap.

References

[1] A. Chen, J. He, L. Xu, F. C. Gey, and J. Meggs, 'Chinese Text Retrieval Without Using a Dictionary'. In the Proceedings of the 20th International Conference on Research and Development in Information Retrieval (SIGIR-97), pp. 42-49, July 1997.

[2] D. Hiemstra and A. de Vries, 'Relating the new language models of information retrieval to the traditional retrieval models.' CTIT Technical Report TR-CTIT-00-09, May 2000.

[3] IBM's AlphaWorks Translation Service, http://www.alphaworks.ibm.com/aw.nsf/html/mt

[4] Linguistic Data Consortium (LDC), Hong Kong Laws Parallel Text, described at http://www.ldc.upenn.edu/Catalog/LDC2000T47.html

[5] Linguistic Data Consortium (LDC), Hong Kong News Parallel Text, described at http://www.ldc.upenn/Catalog/LDC2000T46.html

[6] Ken Lunde, *CJKV Information Processing*, O'Reilly & Associates, January 1999.

[7] J. Mayfield, P. McNamee, and C. Piatko, 'The JHU/APL HAIRCUT System at TREC-8.' In E. M. Voorhees and D. K. Harman, eds., *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*. To appear.

[8] P. McNamee, J. Mayfield, and C. Piatko, 'A Language-Independent Approach to European Text Retrieval.' Draft version in the *Working Notes of the CLEF-2000 Workshop*, Lisbon, Portugal, September 2000.

[9] D. R. H. Miller, T. Leek, and R. M. Schwartz, 'A Hidden Markov Model Information Retrieval System.' In the Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR-99), pp. 214-221, August 1999.

[10] E. Miller, D. Shen, J. Liu, and C. Nicholas, 'Performance and Scalability of a Large-Scale N-gram Based Information Retrieval System.' In the *Journal of Digital Information*, 1(5), January 2000.

[11] J.-Y. Nie and F. Ren, 'Chinese Information Retrieval: using characters or words?'. In *Information Processing and Management*, 35(4), 1999.

[12] I. Witten, A. Moffat, and T. Bell, Managing Gigabytes: Compressing and Indexing Documents and Images 2nd edition, Academic Press, 1999.