RMIT and Gunma University at NTCIR-9 GeoTime Task

Michiko Yasukawa* J. Shane Culpepper† Falk Scholer† Matthias Petri†

*Gunma University, Kiryu, Japan
michi@cs.gunma-u.ac.jp

†RMIT University, Melbourne, Australia
{shane.culpepper, falk.scholer, matthias.petri}
@rmit.edu.au

ABSTRACT

In this report, we describe our experimental approach for the NTCIR-9 GeoTime task. For our experiments, we use our experimental search engine, NeWT. NeWT is a ranked self-index capable of supporting multiple languages by deferring linguistic decisions until query time. To our knowledge, this is the first application of ranked self-indexing to a multilingual information retrieval task at NTCIR.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*indexing methods*; H.3.2 [**Information Storage and Retrieval**]: Information Storage—*file organization*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*query formulation, retrieval models, search process*; I.7.3 [**Document and Text Processing**]: Text Processing—*index generation*

General Terms

Text Indexing, Text Compression, Language Independent Text Indexing, Data Storage Representations, Experimentation, Measurement, Performance

1. INTRODUCTION

In this report, we describe our experimental approach for the NTCIR-9 GeoTime task. Our group participated in the English - English and Japanese - Japanese subtasks. Our search engine, NeWT, uses backwards search in a Burrows-Wheeler transform similar to the FM-*index* of Ferragina and Manzini [3] and a *wavelet tree* [5] for occurrence counting in a document array [7]. To our knowledge, this is the first application of ranked self-indexes for multilingual tasks at NTCIR.

In the NTCIR-9 GeoTime task, the document collections consisted the six different newswires shown in Table 1. This collection is consistent with previous NTCIR workshops. Newspaper articles generally exhibit a standardized writing style, proper vocabulary usage, and consistent orthography, in contrast to web pages and blog entries. In the NTCIR-9 GeoTime task, documents and topics were provided in English and Japanese, and each topic contained a translation in both languages. Our primary goal for NTCIR-9 was implement and test our new class of indexing algorithms for multilingual tasks. The indexing approach can also be applied to cross-lingual IR tasks, and we plan to investigate this further in future work.

Collection	Year	Language
Mainichi Daily	1998-2001 (4 years)	English
Korea Times	1998-2001 (4 years)	English
New York Times	2002-2005 (4 years)	English
Xinhua English	1998-2001 (4 years)	English
Mainichi	1998-2005 (8 years)	Japanese

Table 1: NTCIR-9 GeoTime Document Collections.

2. BACKGROUND

For brevity, we do not reproduce our comprehensive discussion of document parsing of CJKV languages using inverted indexes and self-indexes. The necessary background and discussion of NeWT for NTCIR-9 can be found in our NTCIR-9 INTENT task report [9]. Further details about the self-indexing algorithms and parsing techniques used by NeWT can be found in [1, 2, 9].

2.1 Geographic and Temporal Search

In last year's NTCIR workshop, the geographical and temporal Information Retrieval task was introduced [4]. The task was to search a multilingual collection for specific thematic, geographic, or temporal events.

In the NTCIR-8 GeoTime task, the vocabulary-based re-ranking approach [6] achieved balanced effectiveness for precision and recall, while the question decomposition and question answering approach [8] accomplished high precision among the participants. Another promising approach was the Boolean query reformulation for Information Retrieval (ABRIR) [10]. ABRIR formulates a probabilistic Boolean query, and uses the five top-ranked documents for each query to improve effectiveness using pseudo-relevance feedback. The system selects the 300 terms with the highest mutual-information from relevant documents for query expansion. The final query is formulated using the selected terms, and documents are ranked using BM25. ABRIR uses "The EDR Electronic Dictionary" [1] to obtain synonyms for Boolean query construction. In order to overcome problems with NEs (Named Entities) in Japanese Katakana, specific rules for the macron and small letters of Katakana are introduced.

[1] http://www2.nict.go.jp/r/r312/EDR/index.html

NTCIR-9 Workshop Meeting, 2011, Tokyo, Japan.

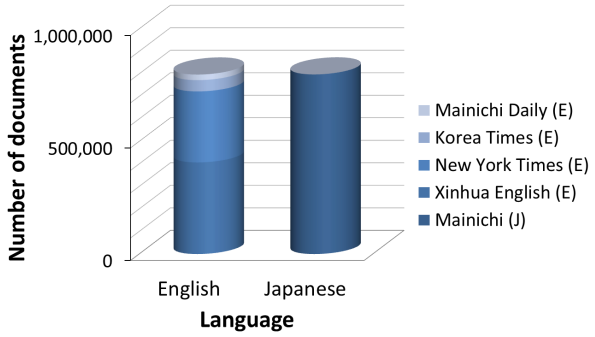

Figure 1: A comparison of the total number of documents in the English and Japanese collections. All documents are combined into a single index using NEWT.

For the NTCIR-9 GeoTime task, we formulate a Boolean query that limits documents by using the timestamp of documents. To gather synonyms for each Boolean query, we use Japanese Wikipedia[2] and Japanese WordNet[3]. We also account for a variety of writing forms in Japanese Katakana NEs, and use regular expressions rather than manually constructed rules. For query expansion, we use 100 n-suffixes that are obtained from the document text using query term pattern matching.

3. EXPERIMENTAL FRAMEWORK

In this section, we explore the experimental setup used for the NTCIR-9 GeoTime task.

3.1 Collection Processing and Indexing

Table 1 shows the document collections that we used in the task, and Figure 1 shows the total number of documents in the English and Japanese collection. All documents were stored in a single index. The primary goal of our participation was to investigate the viability of a language independent index.

In order to compare the effectiveness of our approach with current state-of-the-art, we selected Indri (Indri Search Engine)[4] as a baseline for our experiments. Then, we tested basic ranking and querying in NEWT for the English collection. After the series of experiments in English, we incorporated the Japanese collection into the index and did sanity testing. Collection processing and indexing proceeded as follows.

English documents for Indri: Each document from the English collection was converted to lowercase and written in TREC format, which is understood by the indexing process of Indri[5]. Figure 2 shows an example document in the TREC SGML format. We then indexed the collection using the vanilla Indri settings, including Krovetz stemming and stopword removal.

[2]http://ja.wikipedia.org/
[3]http://nlpwww.nict.go.jp/wn-ja/index.en.html
[4]http://sourceforge.net/projects/lemur/
[5]http://sourceforge.net/apps/trac/lemur/

```
<DOC>
<DOCNO>EN-00001</DOCNO>
<TEXT>
A TREC SGML text file contains one or more documents.
Each document has a unique document number. The
text of the document is contained within TEXT tags.
</TEXT>
</DOC>
```

Figure 2: The TREC SGML document format.

English documents for NEWT: Preprocessing of the four English text collections in Table 1 was minimal for NEWT. Each document from the collection was converted to lowercase. All non-alphanumeric characters and spaces were left unchanged, with no stemming, followed by a distinct `end of document marker`. In principle, our index can also be used to match keywords not normally supported in an inverted index. The preprocessed English document collections were then merged into a single monolithic index.

Japanese documents for Indri: Each document from the Japanese collection was extracted and converted to UTF8 character codes. Then, word segmentation of documents was performed by using ChaSen. After the word segmentation, documents were converted into TREC format in the same way as English collections for Indri. Japanese morphemes in documents were separated with spaces and tokenised into terms within Indri.

Japanese documents for NEWT: Preprocessing for the Japanese text collection in Table 1 was minimal for NEWT. Each document from the collection was extracted and converted to UTF8 character codes. Next, all whitespace was removed from each document to create a contiguous UTF8 string, followed by a distinct `end of document` identifier. The Japanese documents for NEWT are not word segmented. Since our approach is a character-based self-index, we do not need to insert white spaces or alter the original Japanese text. The preprocessed Japanese document collection was then merged into the English index and all Japanese and English queries were ran against the same index. This is in contrast to our Indri baseline, which used two separate indexes. In principle, it should be possible to combine multiple languages in a single Indri index, but we did not explore this alternative.

3.2 Topic Processing

Topics in GeoTime were provided in XML format. The XML file contained a DESCRIPTION and a NARRATIVE field for both English and Japanese topics. The NARRATIVE field is a detailed summary of the information need presented in the DESCRIPTION. In the task, submission of at least one run using only the DESCRIPTION field was mandatory. Runs using both DESCRIPTION and NARRATIVE fields were optional. In the task, we used only the DESCRIPTION field for both English and Japanese monolingual runs.

English queries for Indri: English queries for Indri were treated as a bag-of-words for all terms from the DESCRIPTION field in English, and used the vanilla defaults in Indri including Krovetz stemming and default stopword removal.

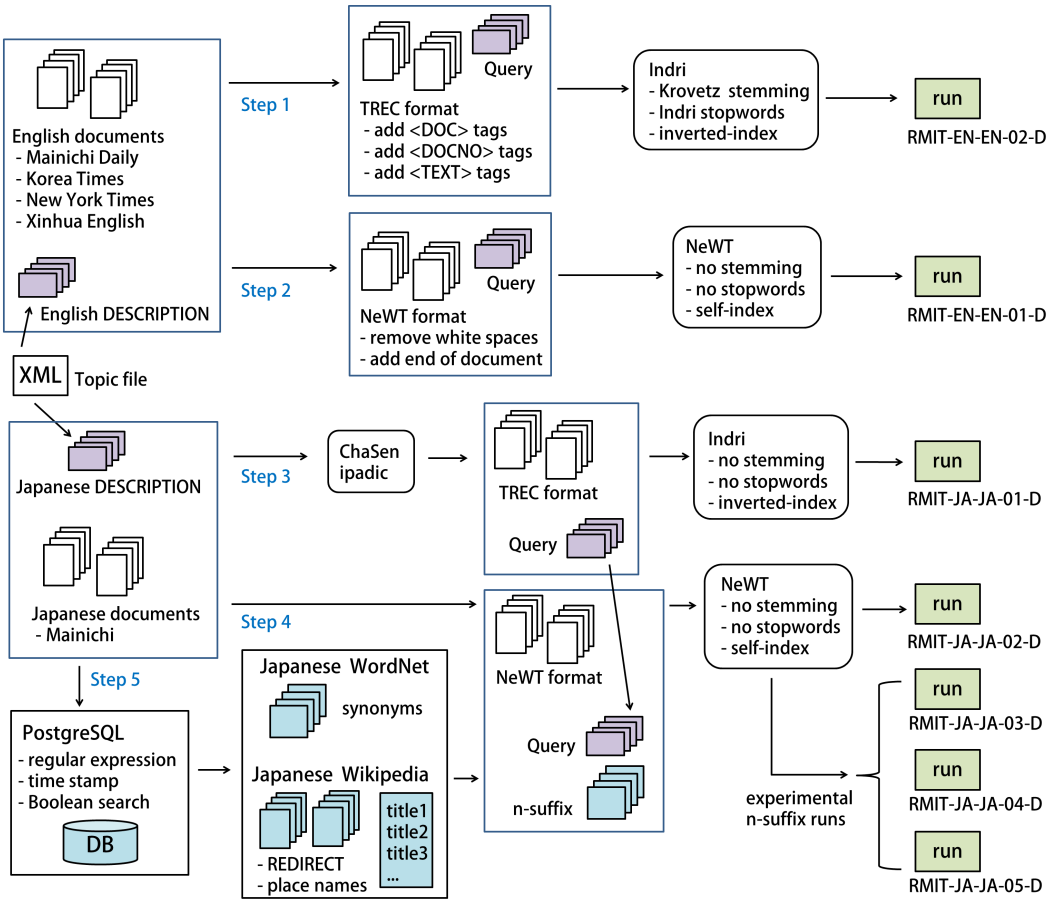

Figure 3: RMIT runs at a glance.

English queries for NeWT: English queries for NeWT were treated as a bag-of-strings with no stemming or stopword removal. Each word was treated as a single string, with the exception of proper nouns, which were treated as a phrase. For example, NeWT treats "steve fosset" as a single term, instead of "steve, fosset". Also, acronyms have spaces added before and after the term to avoid overly aggressive substring matching. For example, the term "ana" is treated as "␣ana␣".

Japanese queries for Indri: Japanese queries used in Indri were composed of nouns and substantive non-nouns extracted from the DESCRIPTION field. Specifically, substantive non-nouns were independent words such as verbs, adjectives, and adverbs. In order to remove the conjugational suffixes of Japanese non-nouns, only Kanji characters in the words were used. To identify parts-of-speech for morphemes, morphological analysis by ChaSen was performed. All particles and affixes were removed during the topic processing. In the Japanese topic processing, all interrogative expressions in the DESCRIPTION field, such as "いつ、どこで" meaning "when and where" were removed using a simple pattern matching `sed`[6] script. This year, Topic #37 required spatial reasoning. In order to include the place name in the query, the geographic coordinates in the topic was associated with the place name. This process is called as "reverse geocoding." For the reverse geocoding, we used the map API provided by the Yahoo! Japan Developer Network[7].

Japanese queries for NeWT: We generated Japanese queries for NeWT in an empirical manner. Unlike the English queries for NeWT, terms from the DESCRIPTION field were not segmented. In order to identify distinct terms for each query, a generic process to parse each sentence into terms was required. In our preliminary experimental study, the same queries used for Indri were with NeWT to see if they work. Unfortunately, the results were not effective from the following reasons:

1. **Accidental short substrings:** Terms in the DESCRIPTION field in Japanese were often short words that caused accidental mismatches among queries and documents. The same problem was recognized in our preliminary testing with English queries. For example, the word "lion" that is an animal also matches the word "billionaire". Since there is no white space in Japanese collection used in NeWT, word boundaries are particularly problematic in the Japanese collection. If the English phrase "the art" is written without

[6]http://sed.sourceforge.net/

[7]http://developer.yahoo.co.jp/webapi/map/

Run	Language	System	Ranking	Preprocess	Query Set
RMIT-EN-EN-01-D	English	Newt	BM25	None	EN-D
RMIT-EN-EN-02-D	English	Indri	Dirichlet LM	Krovetz	EN-D
RMIT-JA-JA-01-D	Japanese	Indri	Dirichlet LM	ChaSen	JA-D
RMIT-JA-JA-02-D	Japanese	Newt	BM25	None	JA-D
RMIT-JA-JA-03-D	Japanese	Newt	BM25	None	JA-D with 2-suffixes
RMIT-JA-JA-04-D	Japanese	Newt	BM25	None	JA-D with 3-suffixes
RMIT-JA-JA-05-D	Japanese	Newt	BM25	None	JA-D with 4-suffixes

Table 2: NTCIR-9 GeoTime English and Japanese runs.

white space, it becomes "theart", creating substring mismatches such as "lightheart ed" that span multiple terms simultaneously. When the short query terms are used in the Japanese collection, the overly aggressive substring matching pollutes the ranking results.

2. **Unexpected document ranking:** In addition to the accidental substring problem, ambiguous or polysemous words were used in the DESCRIPTION field. For example, a general noun word in Japanese meaning "accident" is used in newspaper articles that deal with different "accident" in different contexts. Therefore, the Japanese word meaning "accident" in the queries was ambiguous and useless to filter out irrelevant documents. In other cases, proper nouns in queries were polysemous. A proper noun that clearly means the name of a country in some documents, also announces the name of a product in other documents. Such words were useless to restrict documents within the intended topic. Because the Japanese query terms for Newt were treated as a bag-of-strings, ambiguous or polysemous word resolution was not performed for this task, leading to less discriminative query terms. As a result, the document rankings contained overly general results for the GeoTime task.

In order to avoid overly aggressive substring matching, we performed an n-character suffix expansion to short query terms. For example, the query term "ナス" ("eggplant"), also matches "バナナスムージー" ("banana-smoothie"). However, if the original query term is expanded to include "と" and "の" ("and/of/with"), depending on the context, the false matches no longer occur. The expanded query term "ナスと" and "ナスの" correctly find documents containing "eggplant". For example, a document containing "ナス と リコッタ の ラザニア" ("eggplant lasagna with ricotta cheese") or a document containing "ナス の スパゲティ・ボロネーゼ" ("spaghetti Bolognese with eggplants") match correctly, while the document containing "バナナスムージー" ("banana-smoothie") does not.

In order to identify n-character suffixes for query terms, we first perform a Boolean search to gather a fixed number of high ranking documents containing all of the original query terms. For most topics, the conjunctive queries generated search results with an appropriate number of documents.

However, some of the Boolean conjunctive queries were too specific, and did not produce enough documents for further processing. For these queries, we did query expansion with synonyms and reran the queries in disjunctive form. Synonyms were generated using Japanese Wikipedia and Japanese WordNet. From Japanese Wikipedia, we used definitions of "REDIRECT" that presents an alternative title of a title in Wikipedia. For example, if you access to the article for "UK", you will be sent to a redirect page and its target article, "United Kingdom". We used these alternate titles as synonyms in disjunctive form, e.g., "UK OR United Kingdom". We also used definitions for major place names in Japanese Wikipedia. Specifically, we used the lists in tabular format in the Japanese articles for "South America", "U.S. state", "Europe", "Africa", and "Prefectures of Japan". From Japanese WordNet, the synset for terms were extracted, and synsets with a higher term frequency in the Japanese collection were used in the query expansion. For example, Japanese WordNet identifies several synonyms for the query term "プリンセス" ("princess"), such as "妃殿下", "王女", "皇女", "お姫様", "姫御前", "姫君". If the first two synonyms "妃殿下" and "王女" have a higher term frequency than the query term "プリンセス" in the document collection, they are both used as synonyms in disjunctive form, e.g., "プリンセス OR 妃殿下 OR 王女".

While some topics are overly specific, other topics are overly general, and return too many documents from the initial Boolean search. In order to extract a set of relevant documents, we used the timestamp of newspaper articles. In general, parts of previous newspaper articles are reused in postdated articles to give a summary. Article containing post-analysis of previous events tend to be more informative and reflective. Therefore, the documents most likely to contain "when and where" often appear later in the collection. To exclude less informative documents in the search results, we processed documents in reverse chronological order of the timestamp. For the hybrid search approach using Boolean query expansion and the timestamp ordering, we used PostgreSQL[8]. For each document, the filename, timestamp and document text were extracted. Then, all documents are inserted into a relational table. After all documents were stored in the table, SQL statements for each topic were ran against the table to perform the hybrid search.

Once search results for the Boolean query were obtained, we identified n-character suffixes for each query. We refer to these n-character suffixes as n-suffixes henceforth. If the number of n-suffixes is not limited, the expansion process is intractable. So, we gather a maximum of 100 n-suffixes per topic. If the n-suffixes are too long, the expanded query terms become overly specific. In order to find the best value of n, we generated fixed n-suffix query sets using $n = 2, 3, 4$, and ran each expanded query set separately.

[8]http://www.postgresql.org/

Metacharacter	Matches
.	Any single character.
[chars]	Any of the characters in the brackets.
x	Only the character x.
*	A sequence of 0 or more matches of the atom.
+	A sequence of 1 or more matches of the atom.
?	A sequence of 0 or 1 matches of the atom.
{m}	A sequence of exactly m matches of the atom.

Table 3: Regular Expression Metacharacters

Regular Expression	n-suffixes
regexp_matches('peter piper', 'p.{1}', 'g')	{pe}, {pi}, {pe}
regexp_matches('peter piper', 'p.{2}', 'g')	{pet}, {pip}
regexp_matches('peter piper', 'p.{3}', 'g')	{pete}, {pipe}
regexp_matches('peter piper', 'p.{4}', 'g')	{peter}, {piper}
regexp_matches('peter piper', 'p.{8}', 'g')	{peter pip}

Table 4: Regular Expression Examples

To extract matching substrings from the document text, we used POSIX regular expressions provided by PostgreSQL. Table 3 provides the regular expression atoms and quantifiers used. To gather n-suffixes for queries, we used the *regexp_matches* function that returns all of the substrings resulting from matching the POSIX regular expression pattern. Table 4 provides examples for $n = 1, 2, 3, 4, 8$. The third parameter 'g' causes the function to all matches in the string, not only the first one.

When applying regular expression patterns to Katakana words, orthographical variants were taken into account. To be more specific, interpunct (a punctuation mark consisting of a dot, "・") and macron (Katakana-Hiragana prolonged sound mark, "ー") were the atoms that occur in 0 or 1 matches. Also, small letters [アイウエオカケツヤユヨワ] and their corresponding normal letters [アイウエオカケツヤユヨワ] were matched. For example, when a query term has the written form "パターン・マッチ" ("pattern match"), alternative written forms, such as "パターンマッチ" and "パタンマツチ" are also matched. The processing effort of obtaining orthographical variants was minimized in our GeoTime experiments. Finding the best expansion is the subject in future work.

3.3 Description of Runs

Table 2 provides a brief description for each of the seven runs submitted by the collaboration of RMIT and Gunma University. In Table 2, EN-D and JA-D corresponds English and Japanese DESCRIPTION field respectively. We used traditional term segmentation in Indri as a baseline for the language independent runs of NeWT.

No manual intervention was used for our query runs with the exception of reverse geocoding for Topic 37 of the Japanese run. Creation of Boolean queries was done automatically for the Japanese runs. We did not use reverse geocoding for the English run as our primary goal was to objectively evaluate character-aligned self-indexes with traditional term-based inverted indexes. We did not attempt to directly compare our Japanese and English runs, so the decision not to use reverse geocoding in English does not affect our evaluation.

For the RMIT-EN-EN-01-D run, we used NeWT with bag-of-string queries without stemming or stopword removal, and BM25 ranking as described in [9]. In our second English run, RMIT-EN-EN-02-D, we used Indri with bag-of-word queries with default stopword removal, Krovetz stemming, and the Dirichlet LM ranking function. For the RMIT-JA-JA-01-D, we used Indri, with bag-of-word queries, and the Dirichlet LM ranking function. Text and query preprocessing used the Japanese morphological analyzer ChaSen. For the remaining Japanese runs, we experimented used NeWT with bag-of-string queries and the BM25 ranking function described in [9]. For each of the RMIT-JA-JA-03-D, RMIT-JA-JA-04-D and RMIT-JA-JA-05-D runs, query expansion with n-suffixes of length 3, 4, and 5 respectively was used.

Among the NeWT runs in Japanese, the run with the same query as Indri was given a higher priority than runs with experimentally obtained n-suffix queries. Figure 3 shows a diagrammatic summary of how the runs were prepared.

4. EVALUATION AND RESULTS

In this section we present the evaluation results of our experiments for the GeoTime task.

4.1 Evaluation Metrics

The official effectiveness performance measures for the GeoTime task are mean average precision (MAP), Q-measure (Q), and normalised discounted cumulative gain at cutoffs 10, 100 and 1000 (nDCG@10, nDCG@100, nDCG@1000). Our team submitted seven runs for the GeoTime task.

4.2 English Runs

The results for the two monolingual English runs are shown in Table 5. Compared to the baseline Indri run RMIT-EN-EN-02-D, the NeWT run RMIT-EN-EN-01-D shows higher performance towards the top of the ranked list (nDCG@10), but more poorly on metrics that take longer segments of the ranked results list into account. Overall, there is no statistically significant difference between the runs on any of the five reported metrics (paired t-test).

4.3 Japanese Runs

Evaluation results for the five submitted monolingual Japanese runs are shown in Table 6. The Indri run RMIT-JA-JA-01-D is treated as a baseline submission. The NeWT run RMIT-JA-JA-02-D performed worse than the baseline across all five evaluation measures (significant at the 0.05 level for MAP, Q and nDCG@10, and at the 0.01 level for nDCG@100 and nDCG@1000). The run RMIT-JA-JA-03-D, using 2-suffixes, performed slightly worse than the baseline, but the difference was not statistically significant. The runs RMIT-JA-JA-04-D and RMIT-JA-JA-05-D, which used 3- and 4-suffixes respectively, were more effective than the baseline towards the top of the result list (nDCG@10), but the differences were not statistically significant.

5. CONCLUSIONS

Run	MAP	Q	nDCG@10	nDCG@100	nDCG@1000
RMIT-EN-EN-01-D	0.2477	0.2524	0.4282	0.3691	0.4232
RMIT-EN-EN-02-D	0.2830	0.3057	0.3531	0.3763	0.4992

Table 5: Effectiveness results based on mean average precision, Q-measure, and nDCG at cutoffs 10, 100 and 1000 for the English runs. † and ‡ indicate statistical significance relative to the baseline run RMIT-EN-EN-01-D at the 0.05 and 0.001 levels respectively, based on a paired t-test.

Run	MAP	Q	nDCG@10	nDCG@100	nDCG@1000
RMIT-JA-JA-01-D	0.3779	0.4119	0.4769	0.5109	0.6000
RMIT-JA-JA-02-D	0.3084†	0.3239†	0.3510†	0.3936‡	0.4539‡
RMIT-JA-JA-03-D	0.3282	0.3349	0.4768	0.4653	0.5352
RMIT-JA-JA-04-D	0.3671	0.3714	0.5230	0.5211	0.5809
RMIT-JA-JA-05-D	0.3376	0.3398	0.4988	0.4841	0.5457

Table 6: Effectiveness results based on mean average precision, Q-measure, and nDCG at cutoffs 10, 100 and 1000 for the Japanese runs. † and ‡ indicate statistical significance relative to the baseline run, RMIT-JA-JA-01-D, at the 0.05 and 0.001 levels respectively, based on a paired t-test.

In this report, we have presented results for our new experimental ranked self-index NeWT. While overall effectiveness for document ranking in our first attempt with the GeoTime topics is not significantly better than comparable inverted indexing approaches, we have shown that ranked self-indexes are a viable alternative to these approaches for multilingual document collections. Our method does not require any domain knowledge about the underlying text being indexed, and multiple languages can easily be incorporated into a single index. Furthermore, all domain and language decisions can be deferred until query time, creating new opportunities to improve effectiveness using query expansion and relevance feedback. In future work, we will investigate new approaches to improve system effectiveness, and explore other alternatives to circumvent the substring rank pollution problem identified in this work.

6. ACKNOWLEDGMENTS

The second author was supported by the Australian Research Council.

7. REFERENCES

[1] J. S. Culpepper, G. Navarro, S. J. Puglisi, and A. Turpin. Top-k ranked document search in general text databases. In M. de Berg and U. Meyer, editors, *Proceedings of the 18th Annual European Symposium on Algorithms (ESA 2010), Part II*, volume 6347 of *LNCS*, pages 194–205. Springer, 2010.

[2] J. S. Culpepper, M. Yasukawa, and F. Scholer. Language independent ranked retrieval with NeWT. In *Proceedings of the 16th Australasian Document Computing Symposium (ADCS 2011)*, pages 18–25, December 2011.

[3] P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Proceedings of the 41st IEEE Annual Symposium on Foundatations of Computer Science (FOCS 2000)*, pages 390–398. IEEE Computer Society Press, November 2000.

[4] F. Gey, R. Larson, N. Kando, J. Machado, and T. Sakai. NTCIR-GeoTime overview: Evaluating geographic and temporal search. In *Proceedings of NTCIR-8 Workshop Meeting*, pages 147–153, June 2010.

[5] R. Grossi, A. Gupta, and J. S. Vitter. Higher-order entropy-compressed text indexes. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2003)*, pages 841–850, January 2003.

[6] K. Kishida. Vocabulary-based re-ranking for geographic and temporal searching at NTCIRGeoTime task. In *Proceedings of NTCIR-8 Workshop Meeting*, pages 181–184, June 2010.

[7] S. Mithukrishnan. Efficient algorithms for document retrieval problems. In D. Eppstein, editor, *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2002)*, pages 657–666, January 2002.

[8] T. Mori. A method for GeoTime information retrieval based on question decomposition and question answering. In *Proceedings of NTCIR-8 Workshop Meeting*, pages 167–172, June 2010.

[9] M. Yasukawa, J. S. Culpepper, F. Scholer, and M. Petri. Rmit and gunma university at ntcir-9 intent task. In *Proceedings of the NTCIR-9 Workshop Meeting*, December 2011.

[10] M. Yoshioka. On a combination of probabilistic and boolean IR models for question answering. In *Proceedings of AIRS2010*, pages 588–598, 2010.