DCU at the NTCIR-12 SpokenQuery&Doc-2 Task

David N. Racca
ADAPT Centre
School of Computing
Dublin City University
Dublin 9, Ireland
dracca@computing.dcu.ie

Gareth J. F. Jones
ADAPT Centre
School of Computing
Dublin City University
Dublin 9, Ireland
gjones@computing.dcu.ie

ABSTRACT

We describe DCU's participation in the NTCIR-12 SpokenQuery&Doc (SQD-2) task. In the context of the slide-group retrieval sub-task, we experiment with a passage retrieval method that re-scores each passage according to the relevance score of the document from which the passage is taken. This is performed by linearly interpolating their relevance scores which are calculated using the Okapi BM25 model of probabilistic retrieval for passages and documents independently. In conjunction with this, we assess the benefits of using pseudo-relevance feedback for expanding the textual representation of the spoken queries with terms found in the top-ranked documents and passages, and experiment with a general multidimensional optimisation method to jointly tune the BM25 and query expansion parameters with queries and relevance data from the NTCIR-11 SQD-1 task. Retrieval experiments performed over the SQD-1 and SQD-2 queries confirm previous findings which affirm that integrating document information when ranking passages can lead to improved passage retrieval effectiveness. Furthermore, results indicate that no significant gains in retrieval effectiveness can be obtained by using query expansion in combination with our retrieval models over these two query sets.

Team Name

DCU

Subtasks

SQ-SCR-SGS (Japanese)

Keywords

Spoken content retrieval, SCR, contextualisation, query expansion

1. INTRODUCTION

The SQ-SCR task at the NTCIR-12 SpokenQuery&Doc-2 (SQD-2) task [4] evaluated the effectiveness of spoken content retrieval (SCR) systems in the task of ranking spoken passages from a collection of speech recordings in order of relevance to a spoken query. For the SQD-2 task, the speech collection consisted of the Spoken Document Processing Workshop (SDPWS) dataset [1], a corpus of 98 oral presentations and lectures in the Japanese language which were also used in previous editions of this task [2, 3]. As in previous years, two subtasks were offered to participants which imposed different constraints in the boundaries of the spoken passages that systems were required to retrieve in response to a query. In the slide-group or known-boundary retrieval subtask, retrieval units were determined by the times when slide transitions were made in a presentation to convey a coherent topic or idea. In the passage or unknown-boundary retrieval subtask, retrieval units must be inferred by the retrieval system and were restricted to sequences of contiguous utterances or inter-pausal units (IPUs) within a single presentation.

In this paper, we describe DCU's participation in the slide-group retrieval subtask. We performed retrieval with a rather simple but effective contextualisation technique that re-ranks slide-group passages based on the relevance scores of documents from which they are taken [8, 10, 6]. This approach has been repeatedly demonstrated to be effective in previous editions of this task [14] as well as in other tasks that involve ranking text elements that are part of larger textual documents such as in traditional XML and passage retrieval tasks [5]. Another technique that has been proven effective multiple times in SCR [14, 16] is pseudo relevance feedback (PRF), in particular, when it is used to expand the original query transcription with terms from the top-ranked elements obtained after a first-pass retrieval. In this respect, our approach differentiates from previous work in that we performed two independent query expansion (QE) processes, one to improve the first-pass retrieval of documents and another to improve the first-pass retrieval of slide-group passages prior to performing passage re-ranking from the document scores. Also, we compute relevance scores for documents and passages using the Okapi BM25 function of probabilistic retrieval [15] into which we incorporate an additional parameter that provides better estimates of inverse document frequency (IDF) scores for the SDPWS collection. In addition, we optimise BM25 and QE parameters jointly to maximise mean average precision (MAP) on the queries and relevance assessments from the SQD-1 by using a non-gradient based optimisation method.

The remainder of the paper is structured as follows. Section 2 describes how we process the text from the manual and automatic speech recognition (ASR) transcripts of the spoken queries and documents. Section 3 overviews the retrieval and PRF models that we use to generate our submission runs, while Section 4 describes the optimisation method we use to tune model parameters. Finally, Section 5 presents the results obtained by our retrieval models in the SQD-2 task and Section 6 concludes.

2. DATA PROCESSING

Table 1: List of manual and ASR transcripts.

SpokenQuery&Doc ID	Short ID
MANUAL	M
REF-WORD-MATCH	A1

This section overviews the speech transcripts that we use to generate our submissions and the procedures we adopt to process the recognised text prior to building search indexes.

2.1 Transcripts and Segmentation

The task organisers provided manual and ASR transcripts of the spoken queries and documents. Furthermore, ASR transcripts of varying quality were made available for participants. These were created with the Julius[1] and Kaldi[2] large vocabulary continuous speech recognition (LVCSR) systems by using different combinations of acoustic and language models [4]. Table 1 lists the transcripts that we use in our experiments. The second column of the table shows short identifiers that we use to refer to each type of transcript throughout this paper. In all our experiments, we use the word-level ASR transcripts and represent a spoken document with the sequence of words contained in its 1-best recognition hypothesis. To achieve this, we extract the words from the 1-best recognition hypothesis of each IPU and concatenate them to form the textual representation of a spoken document. The resulting text documents are then segmented into passages of varying length according to the slide-group annotations provided by the organisers. As result, each text passage is represented by the words that were hypothetically said during the presentation of a specific slide-group, and its boundaries are assumed to correspond to those of a speech fragment uttered to convey a particular topic in a presentation.

2.2 Text processing

In order to obtain terms that could be used to index the text documents and passages, we process Japanese text with the morphological analyser MeCab[3] v0.996 and the Ipadic dictionary v2.7.0. Considering that manual transcripts only contain untokenised text, we process these with MeCab to obtain tokens and to also obtain the base (root) form and part-of-speech tag of each identified word. We subsequently use the base form of words as indexing terms. Although the word-level ASR transcripts already contain suitable tokens that can be used as indexing terms, these have been shown to be less effective in previous editions of this task than tokens recognised by MeCab. For this reason, we re-tokenise the text from the 1-best ASR hypothesis with MeCab by feeding the tool with the string that results from concatenating the surface form of the recognised words and removing white-space delimiters. In addition, previous research has demonstrated that lemmas of nouns and verbs are more effective indexing features in Japanese SCR than character or phone n-grams [17]. We therefore further remove from the transcripts all tokens not tagged as verbs or nouns by MeCab. Additionally, we remove words contained in a stop word list with 44 frequent prepositions and determiners after observing that MeCab repeatedly misclassified certain function words as nouns or verbs. By filtering text this way the length of each query and document transcript is reduced to about 50% of its original length. After processing text, we use the Terrier platform v4.0[4] to generate an index for each transcription type. Table 2 shows term statistics of the collection of documents and slide-group passages that result from our text processing pipeline. Differences in the number of passages across transcripts are possible since we do not index passages that become empty after processing the text.

[1] `http://julius.sourceforge.jp/`
[2] `http://kaldi-asr.org/`
[3] `http://mecab.sourceforge.net/`

Table 2: Collection statistics of documents and passages.

ID	#Terms	Documents		Passages		
		Ave.len.	S.D.len.	N	Ave.len.	S.D.len.
M	6230	1769.17	276.15	2329	74.48	67.61
A1	6131	1752.15	262.81	2334	73.62	67.56

Table 3: Error rates of document transcripts.

ID	WER	TER	BIA
M	0%	0%	100%
A1	43.7%	70.0%	42.8%

Table 3 reports word error rate (WER), term error rate (TER) [11] and binary index accuracy (BIA) [20] figures for each type of document transcript. As before, these values are computed from the text that results from applying our text processors and filters, as an attempt to obtain figures that could better reflect the potential impact of ASR errors on retrieval effectiveness. WER values are computed at the IPU level with the NIST sclite tool v2.3 while TER and BIA values are computed and averaged at the passage level. In contrast to WER, the TER and BIA measures disregard word ordering and consider a substitution error as two errors: an insertion plus one deletion. These metrics estimate differences between the reference and ASR indexes and are thus better suited to measure the quality of ASR transcripts in SCR applications. In particular, TER [11] is the sum of the term frequency differences between the reference and hypothesised documents divided by the length of the reference document. BIA [20] disregards term counts and is computed as the product between the fraction of unique terms from the reference found in the hypothesis document (recall) and the fraction of unique terms from the hypothesis found in the reference document (precision).

Table 4 shows term statistics and recognition rates for the post-processed transcripts of the spoken queries. We present statistics for both the query sets used at the SQD-1 and SQD-2 tasks, since we use the SQD-1 queries for tuning the parameters of our retrieval models. With respect to SQD-1, SQD-2 queries are longer on average and contain less ASR errors for nouns and verbs. By comparing the values from Tables 4 and 3, it can be seen that spoken queries are harder to recognise than the presentation speeches.

[4] `http://terrier.org`

Table 4: Error rates and statistics of query transcripts.

Queries	N	ID	WER	TER	BIA	Terms	Ave.len.	S.D.len.
SQD-1	37	M	0%	0%	100%	373	24.13	11.61
		A1	51.6%	80.6%	41.6%	490	29.64	15.14
SQD-2	80	M	0%	0%	100%	714	30.77	12.06
		A1	49.2%	68.3%	47.3%	961	36.85	16.65

3. RETRIEVAL MODELS

We use a conventional document retrieval technique to rank documents or passages in order of relevance to a spoken query. Our ranking function for computing an element's relevance score is based on the Okapi BM25 function of probabilistic retrieval [15]. Besides the well known k_1, b, and k_3 parameters, we include a fourth parameter, namely $d \geq 1$, as the exponent of the inverse document frequency weight [21]. This parameter can be adjusted to increase the relative difference between weights assigned to frequent and rare terms. In our experiments with the SDPWS, setting $d > 1$ results in improved effectiveness as this may provide better estimates of IDF weights for terms that are underrepresented in the collection. This particular issue is explored in more detail by Murata et. al. [13] who propose an alternative IDF function that compensates for underrepresented terms that are assigned unusually high IDF scores. In our preliminary experiments with the SDPWS collection, we observed that Murata et. al.'s function is less effective than the modification that we propose above. Consequently, our term weighting function calculates the weight of a term t occurring in element e and query Q as follows:

$$w_e(t) = \frac{(k_1+1)tf(t)}{tf(t)+k_1(1-b+b\frac{\text{docl}}{\text{avel}})} \frac{(k_3+1)qf(t)}{qf(t)+k_3} w_1(t)^d \quad (1)$$

where $tf(t)$ and $qf(t)$ denote the number of occurrences of t in e and Q respectively, *docl* is the length of e, *avel* is the average length of all the elements of the same type in the collection and b, k_1, k_3, and d are tuning parameters. In Equation 1, $w_1(t)$ is given by the Robertson Spärck Jones Relevance Weight (RW), defined as follows:

$$RW(t) = \log \frac{(r_t+0.5)(N-R-n_t+r_t+0.5)}{(n_t-r_t+0.5)(R-r_t+0.5)} \quad (2)$$

where n_t is the number of elements containing term t, N is the number of elements in the collection and $R = r_t = 0$. Given this term scoring function, we rank elements according to their relevance score with respect to Q, defined by:

$$S_{BM25}(Q,e) = \sum_{t \in Q \cap e} w_e(t) \quad (3)$$

We also experiment with PRF and QE for improving the ranking of documents and passages. In this case, we perform an initial retrieval with $S_{BM25}(Q,e)$ and expand the query with terms found in the top R ranked elements, which are then assumed to be relevant to the query. The criteria for selecting an expansion term t from the set of pseudo-relevant elements is based on the term's Offer Weight (OW) [18], defined as:

$$OW(t) = r_t RW(t) \quad (4)$$

where r_t denotes the number of elements assumed relevant that contain the term t. All terms occurring in the pseudo-relevant elements are then ranked by their OW and the top T terms, which are not already in the original query, are included in the expanded query Q'. In our implementation, we do not re-calculate weights for query terms that are already in the original query. Finally, documents or passages are retrieved again and ranked according to $S_{BM25}(Q',e)$. The resulting scoring function is then defined as:

$$S_{BM25\text{-}QE}(Q,e) = S_{BM25}(Q',e) \quad (5)$$

3.1 Document Score Interpolation (DSI

Traditional IR models assume that the relevance of a document is independent of the relevance of other documents from the collection. Although this assumption may seem reasonable in document retrieval applications, it certainly seems less justifiable in the case of passage retrieval where many of the elements to be ranked may belong to a single document. Passages that belong to the same document are more likely to be about similar topics and, therefore, more likely to condition the probability of relevance of other passages that also occur in that document. In lectures or academic presentations, for example, it is normal for a presenter to provide an introduction at the beginning of the talk which, even though it may occur some minutes before the full presentation of a particular topic, it may still be of importance for this topic and possibly contain some useful terms which may not be mentioned later in the presentation. In addition, very short passages may not contain enough terms to be retrieved effectively. In this case, it seems logical to consider a longer informational unit where the target passage could be contextualised in order to improve its retrievability.

A simple approach to contextualising a passage with information from its document is to combine the passage's relevance score with the score of the source document. Firstly, passages and documents are scored independently to form two separate ranked-lists of results. Secondly, passages retrieved initially are re-ranked according to the relevance scores of their documents. Among the methods that exist for score combination, we adopt a simple weighted linear interpolation of scores or CombSUM [8, 6, 10]. The relevance score of a passage p within document D is then given by:

$$S_{DSI}(Q,p) = \lambda S_{BM25}(Q,D) + (1-\lambda) S_{BM25}(Q,p) \quad (6)$$

where the interpolation parameter λ adjusts the influence of the document score over the combined score. In other words, the λ parameter controls the amount of context that is considered to compute the relevance score of the passage. As it is normally recommended, we normalise the document and passage scores between 0 and 1 before calculating their interpolation.

Additionally, we experiment with the document interpolation method and the QE method described previously. In this case, QE is performed independently to improve the ranking of documents and passages. This means that we generate two expanded queries, one from the first-pass retrieval performed at the document level Q_d and another from the first-pass retrieval performed at the passage level Q_p. These expanded queries are then used to perform a second-pass retrieval for documents and passages and the resulting scores are combined with the CombSUM strategy. This is expressed as:

$$S_{DSI\text{-}QE}(Q,p) = \lambda S_{BM25}(Q_d,D) + (1-\lambda) S_{BM25}(Q_p,p) \quad (7)$$

where Q_d and Q_p denote the expanded queries obtained from the documents and passages first-pass retrieval steps respectively.

4. PARAMETER OPTIMISATION

Most retrieval models contain one or more free parameters that control diverse aspects of their term weighting

scheme, such as document length normalisation, sub-linear saturation of term frequencies, or interpolation weights in language modelling approaches. Adjusting these parameters appropriately for the particular task and test collection at hand can often provide increased retrieval effectiveness in comparison to using the recommended parameter settings of the model. This is particularly the case when the set of recommended parameters has been estimated for tasks and collections that are different to the ones being tested [9].

In the Okapi BM25 model presented in Section 3, there are four parameters that need to be estimated: b that adjusts the degree of length normalisation; k_1 and k_3 which control the rate of increase of the TF factor as the raw term frequency of a term increase in the element and query respectively; and the newly incorporated parameter d which controls the rate of decrease of the IDF factor as the collection frequency of a term increase. Moreover, there are two additional parameters that need to be estimated when performing QE: the number of elements R that are assumed relevant from the first-pass retrieval step; and the number of terms T that are sampled to expand the original query. In addition to this, in the DSI method the optimal BM25 parameters for ranking the documents may be different from those for ranking the passages. Therefore, we consider these as independent parameters that need to be optimised. By counting the interpolation parameter λ, the total number of parameters that need to be estimated for the DSI and DSI-QE models adds up to 9 and 13 respectively. The large number of existing parameter configurations discards any possibility of adopting a grid search approach to find effective parameter settings and so we must seek an alternative optimisation method.

Existing approaches to optimising multiple parameters of retrieval models can be classified into two broad categories. Those that try to maximise retrieval effectiveness metrics that are defined over the ranks of the relevant documents [19], such as MAP or normalised discounted cumulative gain (NDCG), and those that try to optimise alternative objective functions which are commonly designed to correlate well with rank-dependent metrics and to permit, at the same time, the application of gradient-descent methods [7] to solve the optimisation problem. For our purposes, we implement a general optimisation method which seeks to maximise MAP directly on a given set of queries. This method can be considered a more efficient alternative to exhaustive search since it selectively explores different regions in the search space that seem more likely to contain a global or local optima. The optimisation method we implement belongs to the family of unconstrained line search optimisation methods [12], which has been already used to optimise BM25 parameters in previous research [19].

4.1 Line Search

Let $\mathbf{p} = \langle p_1, p_2, \ldots, p_n \rangle$ be the vector of parameters that we want to optimise and $\boldsymbol{\theta} = \langle \theta_1, \theta_2, \ldots, \theta_n \rangle$ a initial parameter configuration. For a particular parameter p_i that can accept values in some interval $\alpha = [x, y]$, a line search is performed by evaluating the objective function at M distinct values of p_i while the values of the rest of the parameters are kept fixed. The M values are sampled equidistant in α and initially centred around θ_i. At each subsequent iteration of the algorithm, the size of the search interval α is reduced by a factor $0 < r < 1$ and the value of p_i that best maximises the objective function so far is chosen as the next point for centring the following M samples that are taken from α. This procedure is repeated for p_i until: (i) the size of α becomes smaller than some ϵ; (ii) a maximum number $maxit$ of iterations have been performed; or (iii) the optimal value of p_i remains the same after $minit$ iterations. In our implementation of line search, we set $M = 20$, $maxit = 30$ and $minit = 5$. Additionally, we set $\epsilon = 0.01$ and $r = 0.8$ for parameters that can take values in $\mathbb{R}$ while for those that can only take values in $\mathbb{N}$ we set $\epsilon = 1$ and reduce the size of α by 1 at every iteration. In order to reduce the size of the search space for parameters in $\mathbb{R}$ we truncate their values to two decimal positions. Values for α are set differently for each parameter as shown in Table 5.

Table 5: Parameter ranges and initial values.

Param	Range (α)	Initial value (θ_i)
b	$[0, 1]$	0.75
k_1	$[0, 5]$	1.20
k_3	$[0, 1000]$	1000
d	$[1, 4]$	1.00
R	$[1, 50]$	0
T	$[1, 30]$	0
λ	$[0, 1]$	0.50

4.2 Promising Directions

We can perform a line search for every parameter in $\mathbf{p}$ to obtain an optimal configuration of values $\boldsymbol{\theta}^*$. The vector from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}^*$ suggests a "promising" direction in the multidimensional parameter space, so we further perform an additional line search on this direction by modifying the values of all the parameters linearly from θ_i to θ_i^*. By doing this, we hope to explore interesting regions of the parameter space which may led us to find even better parameter configurations. The process of performing n one-dimensional line searches plus one final multi-dimensional line search in the promising direction is commonly referred to as an epoch. In our implementation, we perform up to a maximum of 10 epochs and stop searching when the process results in the same parameter configuration in two consecutive epochs.

5. EXPERIMENTS AND RESULTS

We first use the queries from the NTCIR-11 SQD-1 task to learn parameters for the DSI-QE model by using the optimisation method described in Section 4. We perform three optimisations in total: one by using the M transcripts of queries and documents; one by using the M transcripts of queries and the A1 transcripts of documents; and one by using A1 transcripts for both the queries and documents. The optimal parameters found for these three combinations of transcripts are shown in Table 6. Although the optimal values differ depending on which transcripts are used, most parameters tend to converge to similar values and some trends can be thus identified. For example, length normalisation (b) seems to be beneficial when scoring passages, but not so important when ranking documents. Also, high values of k_3 suggest that the contribution of query frequencies to passage scores should be modelled as a linear function and not as a log-shaped function. Differences in k_1 indicate that the contribution of term counts onto relevance scores should saturate faster when scoring passages than when scoring documents. Additionally, the optimal values for λ indicate that

Table 6: Optimal parameter settings for the DSI-QE model and the SQD-1 queries.

Query	Doc.	Level	b	k_1	k_3	d	R	T	λ
M	M	doc.	0.00	4.30	1.27	1.02	4	7	0.70
		pass.	0.38	1.84	254.82	1.03	4	30	
M	A1	doc.	0.02	3.31	1.51	1.00	5	26	0.63
		pass.	0.38	2.52	948.95	1.05	2	5	
A1	A1	doc.	0.38	3.48	4.89	1.03	3	28	0.63
		pass.	0.47	2.10	225.00	1.11	2	12	

Table 7: MAP scores obtained on the SQD-1 queries.

Query	Doc.	Model	QE	MAP
M	M	BM25	-	.241
		BM25	✓	.262
		DSI	-	**.330**
		DSI	✓	**.387***
M	A1	BM25	-	.190
		BM25	✓	.179
		DSI	-	**.240***
		DSI	✓	**.278***
A1	A1	BM25	-	.178
		BM25	✓	.183
		DSI	-	**.253***
		DSI	✓	**.299***

document scores should be given more significance than passage scores in the DSI model.

Table 7 shows MAP scores obtained using the BM25 and DSI models with and without QE for the SQD-1 query set and the three combinations of transcript types. Figures in bold and those marked with * show statistically significant differences with respect to BM25 and BM25-QE respectively according to a paired t-test with 95% confidence. In experiments with the BM25 model, parameter values are set to those found to be optimal for ranking passages with the DSI-QE model, depicted in Table 6. The results show that the DSI model with and without QE significantly outperforms the baseline when tested on the training queries. This re-validates previous findings that have demonstrated the benefits of considering passages in the context of documents when performing passage retrieval [14]. Although the figures suggest that QE provides further gains in retrieval performance in the DSI model, the differences are not significant.

We run the same set of retrieval experiments on the queries of the SQD-2 task to generate our final submissions and evaluate the models on queries that were not used as training data. Table 8 presents the results. The second column in the table shows the ID suffix of our runs as they are reported in [4]. In addition, the † symbol marks statistically significant differences with respect to the DSI-QE model. The results on the test queries are consistent with those obtained with the training queries and provide further evidence of the advantage of considering the global context of a passage to compute its relevance score. The results also show that QE does not provide significant gains in performance. The minor improvements obtained with QE models in the experiments with the SQD-1 queries disappear when models are tested on SQD-2 queries. We suspect this is due to overfitting. If this is the case, the results may further suggest that it is difficult to find values for the QE parameters R and T that can provide gains in retrieval effectiveness for both training and test queries.

Table 8: MAP scores obtained on the SQD-2 queries.

Query	Doc.	SQ-SCR ID	Model	QE	MAP
M	M	TXT-9	BM25	-	.278
		TXT-8	BM25	✓	.293
		TXT-1	DSI	-	**.343***
		TXT-7	DSI	✓	**.342***
M	A1	TXT-10	BM25	-	.212
		TXT-11	BM25	✓	.217
		TXT-6	DSI	-	**.279***†
		TXT-2	DSI	✓	.238
A1	A1	SPK-9	BM25	-	.188
		SPK-7	BM25	✓	.183
		SPK-8	DSI	-	**.250***
		SPK-1	DSI	✓	**.240***

6. CONCLUSIONS

This paper has described DCU's participation at the NTCIR-12 SQD-2 task. We participated in the SQ-SCR task over slide-group segments (SGS) and submitted runs for three transcript combinations of spoken queries and documents. The goal of our submissions was to compare the retrieval effectiveness of two retrieval models for the task of ranking spoken passages in order of relevance to a spoken query: a baseline retrieval model based on the Okapi BM25 function; and a document interpolation model that additionally considers document-level relevance scores to determine the final rank of a passage. We have also experimented with PRF to expand the transcripts of the spoken queries with terms found in top-ranked elements and with an heuristic search optimisation technique that permits tuning of BM25 and QE parameters jointly to a given set of queries and relevance assessment data.

Results of retrieval experiments performed on queries from the SQD-1 and SQD-2 tasks re-affirm previous findings that indicate that the rank of relevant passages can be improved if passages are considered within the context of their documents and that a simple way to achieve this is by interpolating their relevance scores. Furthermore, the experimental results confirm that the PRF and QE techniques that we apply provide no significant gains in retrieval performance overall and that it is hard to find good optimal settings for QE parameters that generalise well to test queries.

In future work, we will experiment with retrieval models that not only utilise information from the full document to estimate the relevance score of a passage but that also exploit local context from a passage's neighbouring elements.

7. ACKNOWLEDGEMENTS

This work is supported by Science Foundation Ireland through the CNGL Programme (Grant No: 12/CE/I2267) in the ADAPT Centre at Dublin City University.

8. REFERENCES

[1] T. Akiba, K. Aikawa, Y. Itoh, T. Kawahara, H. Nanjo, H. Nishizaki, N. Yasuda, Y. Yamashita, and K. Itou. Test collections for spoken document retrieval from lecture audio data. In *Proceedings of LREC'08*, pages 1572–1577, Marrakech, Morocco, 2008.

[2] T. Akiba, H. Nishizaki, K. Aikawa, X. Hu, Y. Itoh, T. Kawahara, S. Nakagawa, and H. Nanjo. Overview of the NTCIR-10 SpokenDoc-2 Task. In *Proceedings of NTCIR-10*, pages 573–587, Tokyo, Japan, 2013.

[3] T. Akiba, H. Nishizaki, H. Nanjo, and G. J. F. Jones. Overview of the NTCIR-11 SpokenQuery&Doc task. In *Proceedings of NTCIR-11*, pages 350–364, Tokyo, Japan, 2014.

[4] T. Akiba, H. Nishizaki, H. Nanjo, and G. J. F. Jones. Overview of the NTCIR-12 SpokenQuery&Doc-2 task. In *Proceedings of NTCIR-12*, Tokyo, Japan, 2016.

[5] P. Arvola, J. Kekäläinen, and M. Junkkari. Contextualization models for XML retrieval. *Information Processing & Management*, 47(5):762–776, 2011.

[6] B. T. Bartell, G. W. Cottrell, and R. K. Belew. Automatic combination of multiple ranked retrieval systems. In *Proceedings of SIGIR'94*, pages 173–181, New York, NY, USA, 1994. Springer.

[7] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of ICML'05*, pages 89–96, New York, NY, USA, 2005. ACM.

[8] J. P. Callan. Passage-level evidence in document retrieval. In *Proceedings of SIGIR'94*, pages 302–310, New York, NY, USA, 1994. Springer.

[9] A. Chowdhury, M. C. McCabe, D. Grossman, and O. Frieder. Document normalization revisited. In *Proceedings of SIGIR'02*, pages 381–382, New York, NY, USA, 2002. ACM.

[10] E. A. Fox and J. A. Shaw. Combination of multiple searches. In *Proceedings of TREC-2*, pages 243–252. NIST, 1994.

[11] S. E. Johnson, P. Jourlin, G. Moore, K. Spärck Jones, and P. C. Woodland. The Cambridge University spoken document retrieval system. In *Proceedings of ICASSP'99*, pages 49–52, Phoenix, AZ, USA, 1999.

[12] D. G. Luenberger and Y. Ye. *Linear and nonlinear programming*, volume 2. Springer, 1984.

[13] M. Murata, H. Nagano, R. Mukai, K. Kashino, and S. Satoh. BM25 with exponential IDF for instance search. *IEEE Transactions on Multimedia*, 16(6):1690–1699, 2014.

[14] H. Nanjo, T. Yoshimi, S. Maeda, and T. Nishio. Spoken document retrieval experiments for SpokenQuery&Doc at Ryukoku University (RYSDT). In *Proceedings of NTCIR-11*, pages 365–370, Tokyo, Japan, 2014.

[15] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proceedings of TREC-3*, pages 109–126. NIST Special Publication 500-225, 1995.

[16] S.-R. Shiang, P.-W. Chou, and L.-C. Yu. Spoken term detection and spoken content retrieval: Evaluations on NTCIR 11 SpokenQuery&Doc task. In *Proceedings of NTCIR-11*, pages 371–375, Tokyo, Japan, 2014.

[17] K. Shigeyasu, H. Nanjo, and T. Yoshimi. A study of indexing units for Japanese spoken document retrieval. In *Proceedings of WESPAC'09*, Beijing, China, 2009.

[18] K. Spärck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing & Management*, 36(6):779–808, 2000.

[19] M. Taylor, H. Zaragoza, N. Craswell, S. Robertson, and C. Burges. Optimisation methods for ranking functions with multiple parameters. In *Proceedings of CIKM'06*, pages 585–593, New York, NY, USA, 2006. ACM.

[20] L. van der Werff and W. Heeren. Evaluating ASR output for information retrieval. In *Proceedings of the SIGIR'07 Workshop on Searching Spontaneous Conversational Speech*, pages 13–20, Amsterdam, The Netherlands, 2007.

[21] C. Zhai. Notes on the Lemur TFIDF model. Technical report, 2001.