Vol.2009-DBS-148 No.9
Vol.2009-FI-95 No.9
2009/7/28

Revisiting NTCIR ACLIA IR4QA with Additional Relevance Assessments

Tetsuya Sakai,[†1] Noriko Kando,[†2] Chuan-Jie Lin,[†3]
Ruihua Song,[†1] Hideki Shima[†4]
and Teruko Mitamura[†4]

At the NTCIR-7 Workshop Meeting held in December 2008, participating systems of the ACLIA IR4QA task were evaluated based on "qrels version 1," which covered the depth-30 pool for every topic and went further down the pool for a limited number of topics, due to time constraints. This paper reports on revised results based on "qrels version 2" which covers the depth-100 pool for every topic. While the version 1 and version 2 results are generally in agreement, some differences in system rankings and significance test results suggest that the additional effort was worthwhile. This paper also reports on a set of additional experiments with new "pseudo-qrels," which mimics the qrels without relying on any manual relevance assessments. Our pseudo-qrels experiments are surprisingly successful: the Pearson correlation coefficients between performances based on our "size-100" pseudo-qrels and those based on qrels version 2 are over 0.9, and even the Kendall rank correlations are 0.58-0.86. Hence, for the next round of IR4QA at NTCIR-8, we may be able to predict system rankings with reasonable accuracy using size-100 pseudo-qrels, right after the run submission deadline.

1. Introduction

At the NTCIR-7 Workshop Meeting held in December 2008[4)], participating systems of the ACLIA IR4QA[⋆1] task[5)] were evaluated based on "qrels version 1," which covered the depth-30 pool for every topic and went further down the pool for a limited number of topics, due to time constraints[10)]. This paper reports on revised results based on "qrels version 2" which covers the depth-100 pool for every topic. More detailed results based on qrels version 2 can be found in our online publication[9)]. For the basics of information retrieval evaluation using test collections formed through *pooling*, the reader is referred to 7).

Having completed the additional relevance assessments to cover all depth-100 pools, for CS (Simplified Chinese), the average number of relevant documents per topic has increased from 9488/97=97.8 to 16475/97=169.8 (+73%); For CT (Traditional Chinese), it has increased from 5262/95=55.4 to 8620/95=90.7 (+61%); For JA (Japanese), it has increased from 8506/98=86.8 to 10785/98=110.1 (+79%)[⋆2]. To avoid confusion, we stick to the original IR4QA topics sets (97 CS, 95 CT and 98 JA) topics for evaluation (See Section 2 in the IR4QA Overview paper[10)], hereafter referred to as "the Overview"), even though our additional relevance assessments did find a small number of relevant documents for a few of the "deleted" topics. As with qrels version 1, all evaluation metric values were computed using Sakai's `ir4qa_eval`[⋆3].

This paper also reports on a set of additional experiments with new "pseudo-qrels," which mimics the qrels without relying on any manual relevance assessments. The pseudo-qrels files we used at NTCIR-7 treated the top 10 documents in each "sorted pool" as relevant, where a sorted pool was a list of documents sorted by "popularity" (See Section 3)[9)]. Our additional experiments use two new variants of the pseudo-qrels, called "size-100" and "size-R," in addition to the original one "size-10." These experiments are surprisingly successful: the Pearson correlation coefficients between performances based on our size-100 pseudo-qrels and those based on qrels version 2 are over 0.9, and even the Kendall rank correlations are 0.58-0.86. Hence, for the next round of IR4QA at NTCIR-8, we may be able to predict system rankings with reasonable accuracy using size-100 pseudo-qrels, right after the run submission deadline.

The remainder of this paper is organised as follows. Section 2 reports on our new system ranking results using qrels version 2. Section 3 discusses the

[†1] Microsoft Research Asia
[†2] National Institute of Informatics
[†3] National Taiwan Ocean University
[†4] Carnegie Mellon University

[⋆1] Advanced Crosslingual Information Access - Information Retrieval for Question Answering

[⋆2] The CS, CT and JA relevance assessments were coordinated independently by Ruihua Song, Chuan-Jie Lin and Noriko Kando, respectively, but they all used the SEPIA (formerly known as EPAN) relevance assessment interface[5)].

[⋆3] Available at `http://research.nii.ac.jp/ntcir/tools/ir4qa_eval-en`.

Table 1 Performance based on qrels version 2: CS runs; Mean over 97 topics.

run	AP	run	Q	run	nDCG
OT-CS-CS-04-T	0.6184	OT-CS-CS-04-T	0.6192	OT-CS-CS-04-T	0.8086
OT-CS-CS-02-T	0.6028	OT-CS-CS-02-T	0.6010	OT-CS-CS-02-T	0.7895
CMUJAV-CS-CS-02-T	0.5733	CMUJAV-CS-CS-02-T	0.5714	CMUJAV-CS-CS-02-T	0.7680
CMUJAV-CS-CS-01-T	0.5704	MITEL-EN-CS-03-T	0.5693	CMUJAV-CS-CS-01-T	0.7673
MITEL-EN-CS-03-T	0.5670	CMUJAV-CS-CS-01-T	0.5690	MITEL-EN-CS-05-TD	0.7667
MITEL-EN-CS-05-TD	0.5606	MITEL-EN-CS-05-TD	0.5613	MITEL-EN-CS-03-T	0.7619
HIT-EN-CS-01-DN	0.5598	HIT-EN-CS-01-DN	0.5596	MITEL-EN-CS-01-T	0.7616
MITEL-EN-CS-01-T	0.5550	MITEL-EN-CS-01-T	0.5558	OT-CS-CS-03-T	0.7591
HIT-EN-CS-02-T	0.5538	OT-CS-CS-03-T	0.5538	OT-CS-CS-05-T	0.7575
MITEL-EN-CS-04-D	0.5499	OT-CS-CS-05-T	0.5535	MITEL-EN-CS-04-D	0.7571
OT-CS-CS-03-T	0.5482	HIT-EN-CS-02-T	0.5535	MITEL-EN-CS-02-T	0.7507
OT-CS-CS-05-T	0.5478	MITEL-EN-CS-04-D	0.5514	HIT-EN-CS-01-DN	0.7397
MITEL-EN-CS-02-T	0.5394	MITEL-EN-CS-02-T	0.5414	HIT-EN-CS-02-T	0.7337
CMUJAV-EN-CS-01-T	0.5233	CMUJAV-EN-CS-01-T	0.5207	RALI-CS-CS-04-T	0.7293
HIT-EN-CS-02-D	0.5073	HIT-EN-CS-02-D	0.5123	RALI-CS-CS-03-T	0.7268
CMUJAV-EN-CS-02-T	0.5044	CMUJAV-EN-CS-02-T	0.5019	RALI-CS-CS-05-T	0.7242
RALI-CS-CS-05-T	0.4852	RALI-CS-CS-05-T	0.4887	RALI-CS-CS-01-T	0.7182
RALI-CS-CS-03-T	0.4843	RALI-CS-CS-03-T	0.4876	RALI-CS-CS-02-T	0.7144
RALI-CS-CS-01-T	0.4834	RALI-CS-CS-01-T	0.4863	CMUJAV-EN-CS-01-T	0.7140
RALI-CS-CS-04-T	0.4786	RALI-CS-CS-04-T	0.4832	HIT-EN-CS-02-D	0.7016
RALI-CS-CS-02-T	0.4768	RALI-CS-CS-02-T	0.4776	OT-CS-CS-01-T	0.6999
HIT-EN-CS-02-DN	0.4477	HIT-EN-CS-02-DN	0.4542	CMUJAV-EN-CS-02-T	0.6987
KECIR-CS-CS-01-T	0.4424	RALI-EN-CS-04-T	0.4293	RALI-EN-CS-04-T	0.6713
RALI-EN-CS-04-T	0.4208	RALI-EN-CS-05-T	0.4255	HIT-EN-CS-02-DN	0.6638
RALI-EN-CS-05-T	0.4176	RALI-EN-CS-01-T	0.4236	RALI-EN-CS-05-T	0.6563
RALI-EN-CS-02-T	0.4165	RALI-EN-CS-02-T	0.4223	RALI-EN-CS-02-T	0.6551
RALI-EN-CS-01-T	0.4156	OT-CS-CS-01-T	0.4198	RALI-EN-CS-01-T	0.6508
CYUT-EN-CS-03-DN	0.4018	KECIR-CS-CS-01-T	0.4125	CYUT-EN-CS-03-DN	0.6182
OT-CS-CS-01-T	0.3830	CYUT-EN-CS-03-DN	0.4013	CYUT-EN-CS-01-T	0.5749
KECIR-CS-CS-02-DN	0.3753	CYUT-EN-CS-01-T	0.3608	KECIR-CS-CS-01-T	0.5744
CYUT-EN-CS-01-T	0.3586	CYUT-EN-CS-02-D	0.3549	CYUT-EN-CS-02-D	0.5684
KECIR-CS-CS-03-DN	0.3558	KECIR-CS-CS-02-DN	0.3498	KECIR-CS-CS-02-DN	0.5060
CYUT-EN-CS-02-D	0.3519	KECIR-CS-CS-03-DN	0.3380	KECIR-CS-CS-03-DN	0.4993
WHUCC-CS-CS-02-T†	0.2837	WHUCC-CS-CS-02-T†	0.2675	WHUCC-CS-CS-02-T†	0.4054
WHUCC-CS-CS-01-T†	0.2837	WHUCC-CS-CS-01-T†	0.2675	WHUCC-CS-CS-01-T†	0.4054
NLPAI-CS-CS-02-T	0.0990	NLPAI-CS-CS-02-T	0.0924	NLPAI-CS-CS-02-T	0.1966
NLPAI-CS-CS-05-DN	0.0979	NLPAI-CS-CS-05-DN	0.0914	NLPAI-CS-CS-05-DN	0.1934
NLPAI-CS-CS-01-T	0.0917	NLPAI-CS-CS-01-T	0.0841	NLPAI-CS-CS-01-T	0.1865
NLPAI-CS-CS-03-T	0.0882	NLPAI-CS-CS-03-T	0.0811	NLPAI-CS-CS-03-T	0.1786
NLPAI-CS-CS-04-T	0.0845	NLPAI-CS-CS-04-T	0.0768	NLPAI-CS-CS-04-T	0.1716

†These two runs are in fact identical: they contain the same ranked document lists for every topic.

Table 2 Performance based on the qrels version 2: CT runs; Mean over 95 topics.

run	AP	run	Q	run	nDCG
MITEL-CT-CT-03-D	0.5561	MITEL-CT-CT-03-D	0.5715	MITEL-CT-CT-03-D	0.7705
MITEL-CT-CT-02-T	0.5547	MITEL-CT-CT-02-T	0.5700	MITEL-CT-CT-02-T	0.7684
MITEL-CT-CT-01-T	0.5507	MITEL-CT-CT-01-T	0.5653	MITEL-CT-CT-01-T	0.7646
MITEL-CT-CT-04-T	0.5432	MITEL-CT-CT-04-T	0.5545	OT-CT-CT-04-T	0.7531
OT-CT-CT-04-T	0.5304	OT-CT-CT-04-T	0.5488	MITEL-CT-CT-04-T	0.7471
OT-CT-CT-03-T	0.4793	OT-CT-CT-02-T	0.4978	OT-CT-CT-02-T	0.7204
OT-CT-CT-02-T	0.4768	OT-CT-CT-03-T	0.4973	OT-CT-CT-03-T	0.7134
OT-CT-CT-05-T	0.4562	OT-CT-CT-05-T	0.4770	OT-CT-CT-05-T	0.7031
RALI-CT-CT-05-T	0.4051	RALI-CT-CT-05-T	0.4186	RALI-CT-CT-03-T	0.6673
RALI-CT-CT-01-T	0.4030	RALI-CT-CT-01-T	0.4162	RALI-CT-CT-04-T	0.6652
RALI-CT-CT-03-T	0.3874	RALI-CT-CT-03-T	0.4056	RALI-CT-CT-05-T	0.6586
RALI-CT-CT-04-T	0.3861	RALI-CT-CT-04-T	0.4034	RALI-CT-CT-01-T	0.6528
RALI-CT-CT-02-T	0.3850	RALI-CT-CT-02-T	0.3993	RALI-CT-CT-02-T	0.6496
NTUBROWS-CT-CT-01-T	0.3415	NTUBROWS-CT-CT-01-T	0.3574	OT-CT-CT-01-T	0.6424
OT-CT-CT-01-T	0.3077	OT-CT-CT-01-T	0.3533	NTUBROWS-CT-CT-01-T	0.5804
RALI-EN-CT-01-T	0.2759	RALI-EN-CT-05-T	0.2904	NTUBROWS-CT-CT-02-T	0.5115
RALI-EN-CT-05-T	0.2745	RALI-EN-CT-01-T	0.2904	NTUBROWS-CT-CT-03-T	0.4925
RALI-EN-CT-04-T	0.2628	RALI-EN-CT-04-T	0.2808	RALI-EN-CT-04-T	0.4894
RALI-EN-CT-02-T	0.2626	RALI-EN-CT-02-T	0.2769	RALI-EN-CT-05-T	0.4791
CYUT-EN-CT-01-T	0.2469	NTUBROWS-CT-CT-02-T	0.2639	RALI-EN-CT-01-T	0.4769
CYUT-EN-CT-03-DN	0.2362	CYUT-EN-CT-01-T	0.2596	RALI-EN-CT-02-T	0.4757
CYUT-EN-CT-02-D	0.2352	NTUBROWS-CT-CT-03-T	0.2577	NTUBROWS-CT-CT-04-T	0.4739
NTUBROWS-CT-CT-03-T	0.2267	CYUT-EN-CT-02-D	0.2483	CYUT-EN-CT-01-T	0.4596
NTUBROWS-CT-CT-02-T	0.2208	CYUT-EN-CT-03-DN	0.2474	CYUT-EN-CT-02-D	0.4454
NTUBROWS-CT-CT-04-T	0.2102	NTUBROWS-CT-CT-04-T	0.2411	CYUT-EN-CT-03-DN	0.4448
NTUBROWS-CT-CT-05-T	0.1780	NTUBROWS-CT-CT-05-T	0.2090	NTUBROWS-CT-CT-05-T	0.4078

∗The documentIDs in these two runs were all illegal: Their evaluation scores are computed here after a bug fix, even though the pools were created using the original runs.

correlation of the system ranking based on qrels version 2 and that based on pseudo-qrels. Section 4 discusses related work, and Section 5 concludes this paper.

2. Ranking Systems with Qrels Version 2

Table 1, **Table 2** and **Table 3** show, for each document language, the Mean *Average Precision (AP), Q-measure and nDCG*[7] values for each run based on qrels version 2. These correspond to Tables 6-8 in the Overview[10]. With qrels version 1 for CT, Q and nDCG disagreed with AP by ranking MITEL-CT-CT-02-T at the top (Table 7 in the Overview[10]). With version 2, however, all three metrics agree that MITEL-CT-CT-03-D is the best CT run on average.

Table 4, **Table 5** and **Table 6** show the "best" T-runs from each team in terms of average performance. These correspond to Tables 9-11 in the Overview[10]. For each adjacent pair of runs shown in this table, we conducted a two-sided, paired bootstrap test using 1000 bootstrap samples of topics[6]. Our additional relevance assessments have resulted in more system pairs with significant performance differences. For example, with qrels version 1, OT-CS-CS-04-T and MITEL-EN-CS-03-T were not significantly different from each other in terms of AP and Q (Table 9 in the Overview[10]), but with version 2, the difference between these two runs are statistically significant at $\alpha = 0.01$ in terms of AP, Q and nDCG. This suggests that the additional relevance assessments were worthwhile.

Table 3 Performance based on qrels version 2: JA runs; Mean over 98 topics.

run	AP	run	Q	run	nDCG
OT-JA-JA-04-T	0.6999	OT-JA-JA-04-T	0.7068	OT-JA-JA-04-T	0.8632
OT-JA-JA-02-T	0.6682	OT-JA-JA-02-T	0.6748	OT-JA-JA-02-T	0.8439
BRKLY-JA-JA-01-DN	0.6376	BRKLY-JA-JA-01-DN	0.6470	BRKLY-JA-JA-01-DN	0.8192
CMUJAV-JA-JA-01-T	0.5969	BRKLY-JA-JA-02-T	0.6029	BRKLY-JA-JA-02-T	0.7854
CMUJAV-JA-JA-03-T	0.5932	CMUJAV-JA-JA-01-T	0.5987	CMUJAV-JA-JA-01-T	0.7803
CMUJAV-JA-JA-04-T	0.5925	CMUJAV-JA-JA-03-T	0.5954	CMUJAV-JA-JA-04-T	0.7786
BRKLY-JA-JA-02-T	0.5903	CMUJAV-JA-JA-04-T	0.5945	CMUJAV-JA-JA-03-T	0.7766
CMUJAV-JA-JA-02-T	0.5834	CMUJAV-JA-JA-02-T	0.5875	BRKLY-JA-JA-02-DN	0.7757
CMUJAV-JA-JA-05-T	0.5831	BRKLY-JA-JA-02-DN	0.5863	OT-JA-JA-05-T	0.7731
BRKLY-JA-JA-02-DN	0.5810	CMUJAV-JA-JA-05-T	0.5853	CMUJAV-JA-JA-02-T	0.7716
OT-JA-JA-05-T	0.5596	OT-JA-JA-05-T	0.5742	CMUJAV-JA-JA-05-T	0.7696
BRKLY-JA-JA-03-T	0.5469	BRKLY-JA-JA-03-T	0.5540	BRKLY-JA-JA-03-T	0.7484
CMUJAV-EN-JA-01-T	0.4311	OT-JA-JA-03-T	0.4417	OT-JA-JA-01-T	0.7111
CMUJAV-EN-JA-03-T	0.4283	OT-JA-JA-01-T	0.4349	OT-JA-JA-03-T	0.6571
CMUJAV-EN-JA-04-T	0.4269	CMUJAV-EN-JA-01-T	0.4342	CMUJAV-EN-JA-01-T	0.5999
OT-JA-JA-03-T	0.4234	CMUJAV-EN-JA-03-T	0.4312	CMUJAV-EN-JA-03-T	0.5979
CMUJAV-EN-JA-05-T	0.4227	CMUJAV-EN-JA-04-T	0.4297	CMUJAV-EN-JA-04-T	0.5965
CMUJAV-EN-JA-02-T	0.4227	CMUJAV-EN-JA-05-T	0.4257	CMUJAV-EN-JA-05-T	0.5943
OT-JA-JA-01-T	0.3911	CMUJAV-EN-JA-02-T	0.4255	CMUJAV-EN-JA-02-T	0.5928
CYUT-EN-JA-01-T	0.2552	CYUT-EN-JA-01-T	0.2500	CYUT-EN-JA-03-DN	0.4288
CYUT-EN-JA-03-DN	0.2543	CYUT-EN-JA-03-DN	0.2486	CYUT-EN-JA-01-T	0.4218
CYUT-EN-JA-02-D	0.2277	CYUT-EN-JA-02-D	0.2253	CYUT-EN-JA-02-D	0.4058
TA-EN-JA-02-D	0.0154	TA-EN-JA-02-D	0.0167	TA-EN-JA-03-T	0.0446
TA-EN-JA-03-T	0.0128	TA-EN-JA-03-T	0.0157	TA-EN-JA-02-D	0.0349
TA-EN-JA-01-D	0.0119	TA-EN-JA-01-D	0.0118	TA-EN-JA-01-D	0.0261

Table 4 The best T-run from each CS team: "*" and "**" indicate that a run significantly outperforms (at $\alpha = 0.05$ and $\alpha = 0.01$, respectively) than one shown immediately below according to a two-sided paired bootstrap test. Note, however, that pairwise statistical significance is not transitive.

run	AP	run	Q	run	nDCG
OT-CS-CS-04-T	0.6184**	OT-CS-CS-04-T	0.6192**	OT-CS-CS-04-T	0.8086**
CMUJAV-CS-CS-02-T	0.5733	CMUJAV-CS-CS-02-T	0.5714	CMUJAV-CS-CS-02-T	0.7680
MITEL-EN-CS-03-T	0.5670	MITEL-EN-CS-03-T	0.5693	MITEL-EN-CS-05-TD	0.7667
HIT-EN-CS-02-T	0.5538**	HIT-EN-CS-02-T	0.5535**	HIT-EN-CS-02-T	0.7337
RALI-CS-CS-05-T	0.4852*	RALI-CS-CS-05-T	0.4887 **	RALI-CS-CS-04-T	0.7293**
KECIR-CS-CS-01-T	0.4424**	KECIR-CS-CS-01-T	0.4125	CYUT-EN-CS-01-T	0.5749
CYUT-EN-CS-01-T	0.3586*	CYUT-EN-CS-01-T	0.3608**	KECIR-CS-CS-01-T	0.5744**
WHUCC-CS-CS-01-T	0.2837**	WHUCC-CS-CS-01-T	0.2675**	WHUCC-CS-CS-01-T	0.4054**
NLPAI-CS-CS-02-T	0.0990	NLPAI-CS-CS-02-T	0.0924	NLPAI-CS-CS-02-T	0.1966

Table 5 The best T-run from each CT team: "*" and "**" indicate that a run significantly outperforms (at $\alpha = 0.05$ and $\alpha = 0.01$, respectively) than one shown immediately below according to a two-sided paired bootstrap test. Note, however, that pairwise statistical significance is not transitive.

run	AP	run	Q	run	nDCG
MITEL-CT-CT-02-T	0.5547	MITEL-CT-CT-02-T	0.5700	MITEL-CT-CT-02-T	0.7684
OT-CT-CT-04-T	0.5304**	OT-CT-CT-04-T	0.5488**	OT-CT-CT-04-T	0.7531**
RALI-CT-CT-05-T	0.4051*	RALI-CT-CT-05-T	0.4186*	RALI-CT-CT-03-T	0.6673**
NTUBROWS-CT-CT-01-T	0.3415**	NTUBROWS-CT-CT-01-T	0.3574**	NTUBROWS-CT-CT-01-T	0.5804**
CYUT-EN-CT-01-T	0.2469	CYUT-EN-CT-01-T	0.2596	CYUT-EN-CT-01-T	0.4596

Table 6 The best T-run from each JA team: "*" and "**" indicate that a run significantly outperforms (at $\alpha = 0.05$ and $\alpha = 0.01$, respectively) than one shown immediately below according to a two-sided paired bootstrap test. Note, however, that pairwise statistical significance is not transitive.

run	AP	run	Q	run	nDCG
OT-JA-JA-04-T	0.6999**	OT-JA-JA-04-T	0.7068**	OT-JA-JA-04-T	0.8632**
CMUJAV-JA-JA-01-T	0.5969	BRKLY-JA-JA-02-T	0.6029	BRKLY-JA-JA-02-T	0.7854
BRKLY-JA-JA-02-T	0.5903**	CMUJAV-JA-JA-01-T	0.5987**	CMUJAV-JA-JA-01-T	0.7803**
CYUT-EN-JA-01-T	0.2552**	CYUT-EN-JA-01-T	0.2500**	CYUT-EN-JA-01-T	0.4218**
TA-EN-JA-03-T	0.0128	TA-EN-JA-03-T	0.0157	TA-EN-JA-03-T	0.0446

3. Correlation between Qrels Version 2 and Pseudo-Qrels

At NTCIR-7, we investigated whether simple pseudo-qrels files that do not rely on manual relevance assessments can mimic real qrels, since relevance assessments are costly. The pseudo-qrels files were created by taking the top 10 documents from each *sorted pool*, in which each document was ranked by the number of runs containing it (the larger the better) and then by the sum of ranks of the document within those runs (the smaller the better)[10]. Hence, our original pseudo-qrels assumed that the number of relevant document was 10 for every topic[⋆1]. We refer to this as *size-10 pseudo-qrels* hereafter.

However, it turns out that, according to qrels version 2, the average number of relevant documents per topic is 16475/97=169.8 for CS, 8620/95=90.7 for CT, and 10785/98=110.1 for JA[9]. Hence our size-10 pseudo-qrels underestimated the number of relevant documents substantially. In this paper, we report on a set of additional experiments with *size-100 pseudo-qrels* which treated the top 100 documents in the depth-30 pool as relevant[⋆2], and with *size-R pseudo-qrels*, which uses the true number of relevant documents (R) according to qrels version 2 for each topic. Note that the size-R experiments rely on an oracle, i.e., they assume knowledge of R, and is not applicable in practice. This is similar in spirit to Soboroff's "exact-fraction sampling" experiment in 11).

[⋆1] The IR4QA relevance assessments had two relevance levels: $L2$ (relevant) and $L1$ (partially) relevant[10]. The pseudo-qrels treated all of the top ranked documents in the sorted pool as $L1$-relevant.

[⋆2] The depth-30 pool contained more than 100 documents for all topics (CS, CT, JA), except for CS-T68 (96 documents), CS-T329 (98 documents) and CS-T370 (85 documents). Thus, the size-100 pseudo-qrels file for CS in fact contain a little less than 100 documents for these three topics.

Table 7 Pearson correlation, Kendall and Yilmaz/Aslam/Robertson rank correlation: System ranking by size-10 pseudo-qrels vs qrels version 2 for each metric.

	AP	Q	nDCG
CS runs	.682/.621/.610	.734/.646/.608	.832.700/.675
CT runs	.918/.760/.628	.921/.748/.660	.954/.686/.610
JA runs	.940/.753/.530	.945/.693/.470	.975/.753/.508

Table 8 Kendall and Yilmaz/Aslam/Robertson rank correlation: System ranking by size-100 pseudo-qrels vs qrels version 2 for each metric.

	AP	Q	nDCG
CS runs	.989/.846/.677	.991/.862/.684	.995/.859/.737
CT runs	.935/.803/.756	.926/.797/.766	.943/.846/.750
JA runs	.924/.633/.584	.923/.580/.561	.963/.727/.681

Table 9 Kendall and Yilmaz/Aslam/Robertson rank correlation: System ranking by size-R pseudo-qrels vs qrels version 2 for each metric.

	AP	Q	nDCG
CS runs	.994/.900/.791	.994/.897/.793	.997/.897/.817
CT runs	.967/.914/.830	.961/.902/.837	.976/.883/.810
JA runs	.962/.800/.727	.962/.720/.664	.980/.793/.667

Table 7 compares, for each metric (Mean AP, Q or nDCG), performances and rankings according to size-10 pseudo-qrels and those according to qrels version 2. In each cell, the first number is the Pearson correlation coefficient that compares the actual metric values; the second number is Kendall's rank correlation that compares two system rankings; and the third number is Yilmaz/Aslam/Robertson (YAR) rank correlation[13], which is similar to Kendall, but weights the swaps near the top ranks more heavily. The YAR correlation values are computed by treating the version 2 ranking as the ground truth, since YAR correlation is not symmetrical. This table corresponds to Table 27 in the Overview[10].

Table 8 shows a similar set of results for size-100 pseudo-qrels. By comparing it with Table 7, it can be observed that size-100 is a much better choice than size-10, since the correlation values are much higher. The Pearson correlation values are 0.923-0.995, and even the Kendall rank correlation values are 0.580-0.862. The YAR rank correlation values are lower than the Kendall ones, suggesting that size-100 pseudo-qrels are not too accurate at predicting the top ranks, as we shall verify later[⋆1].

Similarly, **Table 9** shows the results for size-R (i.e., oracle) pseudo-qrels. The Pearson correlation values are now 0.961-0.997, and the Kendall rank correlation values are 0.720-0.914: slightly higher than the size-100 results. This suggests that estimating the number of relevant documents per topic may be a good approach to constructing accurate pseudo-qrels[⋆2]. However, since the size-100 pseudo-qrels results are surprisingly impressive, the size-100 pseudo-qrels approach may be adopted for the next round of IR4QA at NTCIR-8: Right after forming the sorted pools using the runs submitted by the participants, we could release an early report, by ranking systems based on the size-100 pseudo-qrels, to predict the official ranking based on the real qrels. If the predictions are sufficiently accurate, an early report like this may accelerate research progress, since this will enable researchers to roughly identify which IR strategies are good or bad without having to wait for the release of qrels files.

Finally, we take a closer look at the correlation results. **Fig. 1**, **Fig. 2** and **Fig. 3** are scatterplots that visualise the relationship between size-100 pseudo-qrels and qrels version 2, in terms of Mean Q. (The graphs for AP and nDCG are extremely similar to the Q graphs and are omitted here.)

Fig. 4 and **Fig. 5** compare the ranking of CS runs according to pseudo-qrels and that according to qrels version 2, in terms of Mean AP and Q, respectively. The runs have been sorted by the version 2 performance. It can be observed that while the original size-10 pseudo qrels file was not very useful, size-100 and size-R pseudo-qrels predict the "true" system ranking relatively well.

Fig. 6 and **Fig. 7** show similar results for the CT runs. **Fig. 8** and **Fig. 9** show similar results for the JA runs. The JA pseudo-qrels files are considerably less accurate than the other two, in that they (even size-100 and size-R pseudo-qrels) are not good at predicting the top ranks. This is also represented by the low YAR correlation values for JA in Table 7, Table 8 and Table 9. However, even for the JA runs, it can be observed that the ranking of the low performers is fairly

[⋆1] When pairwise swaps are uniformly distributed over the ranked list being examined, YAR rank correlation is equivalent to Kendall rank correlation[13].

[⋆2] A crude variable-size approach, that treated the entire depth-30 pool as pseudo-relevant for each topic, was not as successful as the simpler size-100 approach in our experiments.

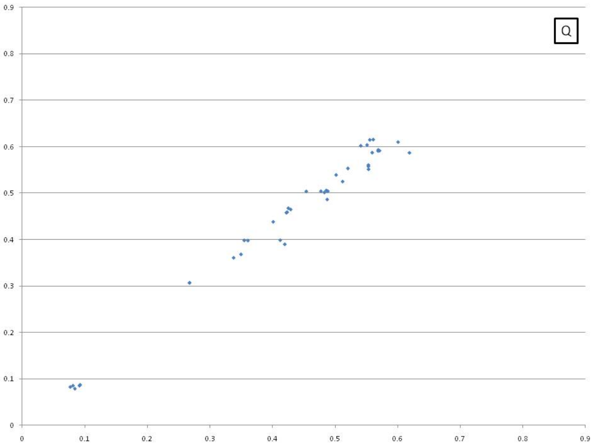

Fig. 1 Qrels version 2 vs. size-100 pseudo-qrels: Scatterplot of Mean Q for CS runs.

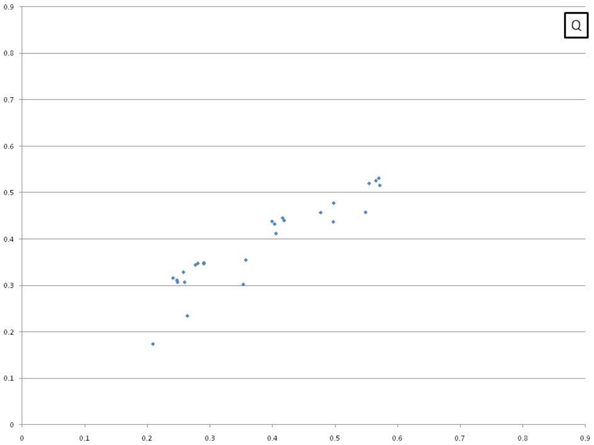

Fig. 2 Qrels version 2 vs. size-100 pseudo-qrels: Scatterplot of Mean Q for CT runs.

accurate. These results do not contradict with our original observation in the Overview[10]: "systems that retrieve popular documents are not necessarily good; however, systems that do not retrieve popular documents are probably bad"[10].

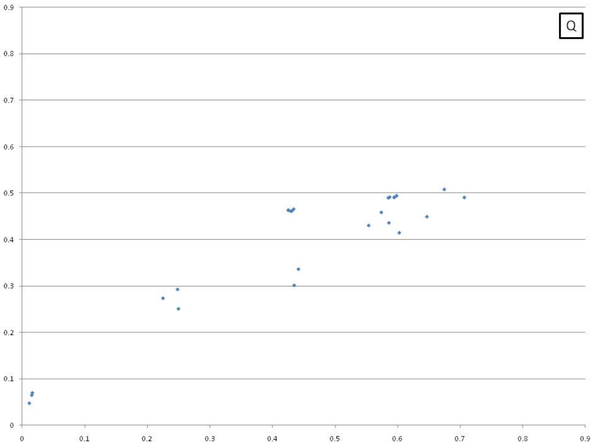

Fig. 3 Qrels version 2 vs. size-100 pseudo-qrels: Scatterplot of Mean Q for JA runs.

4. Related Work

Soboroff, Nicholas and Cahan[11] were probably the first who attempted to rank systems without conducting manual relevance assessments of the pooled documents. *Their* pseudo-qrels were formed by automatically assigning relevance assessments *at random* within the pool. Subsequently, Aslam and Savel[1] showed that the Soboroff method ranks systems in terms of "popularity" rather than performance, by showing that the method behaves very similarly to a simple method that ranks systems according to *average similarity* to other systems. In their experiments using the TREC data, the Pearson correlation coefficients between the "true" system ranking and the pseudo-ranking based on either of the two methods were less than 0.8. The Kendall rank correlations were less than 0.6. Recall that our correlation results are much higher.

At TREC, the standard practice has been to take the top k documents from all submitted runs to form a pool of unique documents, sorted by document IDs[12]. However, we argue that this methodology throws away two important pieces of information, namely, (a) the probability of relevance of each document, as estimated by each system; and (b) the popularity of each document, i.e., how many systems agree that the document is relevant. Recall that our sorted pool directly utilises these clues, by using the number of systems that returned the document as the first sort key and the sum of ranks of the returned document as the second key.

More than a decade ago, Cormack, Palmer and Clarke[3] proposed an efficient relevance assessment method: They suggested preserving the rank order within each run (i.e., use (a) mentioned above) and judging more documents from runs that have returned more relevant documents recently and fewer documents from runs that have returned fewer relevant documents recently. Voorhees[12] claims that this biases the pools toward early-precision systems. In contrast, our sorted pool approach does not favour any particular runs – it just lets popular documents judged first, and the set of documents judged is identical to that with "sort-by-document-ID" pools.

Zobel[14] suggested another pooling method: judge more documents for *topics* that have had many relevant documents found so far and fewer documents for topics with fewer relevant documents found so far. Voorhees[12] claims that adding extra documents later in the pools using his method may affect the assessors' relevance assessments.

In contrast to the sort-by-document-ID approach of TREC, our relevance assessments and pseudo-qrels both rely on sorted pools. Our assumptions are[10]:

(1) Popular documents, i.e., those retrieved at high ranks by many systems, are more likely to be relevant than others;

(2) If there are more relevant documents near the top of the list to be judged than near the bottom, then this makes it easier for the assessors to make judgments more efficiently and consistently than when relevant documents are randomly spread across the list.

An analysis by Sakai and Kando[8] shows that Assumption (1) is valid. Whether Assumption (2) is valid or not should be investigated in our future work.

More recently, Carterette *et al.*[2] used two methods for selecting documents to be judged for efficient contruction of test collections: the first method, the Minimal Test Collections method, is a greedy online algorithm that induces system *rankings* by identifying differences between them; the second method, Statistical Average Precision, samples documents to produce unbiased, minimum-variance estimates of true AP. The experiments were conducted for the TREC Million Query Track, and these methods were designed specifically for evaluation with AP. While these techniques are attractive, we are seeking pooling and pseudo-qrels strategies that do not depend on a particular evaluation metric.

Finally, note that, in contrast to the various "online" document selection algorithms mentioned above, i.e., those in which a particular relevance assessment can change the set of remaining documents to be judged, our approach is strictly offline just like the traditional sort-by-document-ID pools. This simple approach offers several advantages: since we know exactly which documents will be judged and in what order, prior cost estimation and later analysis is easy; since we have static pool files, it is easy for distributed organisers to collaborate.

5. Conclusions

This paper has reported on revised ACLIRA IR4QA results based on qrels version 2 which covers the depth-100 pool for every topic. While the version 1 and version 2 results are generally in agreement, some differences in system rankings and significance test results suggest that the additional effort was worthwhile. This paper also reported on a set of additional experiments with new pseudo-qrels, which mimics the qrels without relying on any manual relevance assessments. Our pseudo-qrels experiments are surprisingly successful: the Pearson correlation coefficients between performances based on our size-100 pseudo-qrels and those based on qrels version 2 are over 0.9, and even the Kendall rank correlations are 0.58-0.86. Hence, for the next round of IR4QA at NTCIR-8, we may be able to predict system rankings with reasonable accuracy using size-100 pseudo-qrels, right after the run submission deadline.

All experiments reported in this paper, however, are based on the assumption that qrels version 2 is the ground truth. Note that both qrels version 2 and pseudo-qrels were created based on the same sorted pools. An interesting question would be: What if the real qrels files were created using the sort-by-document-ID methodology, or any other method that does not rely on the sorted pools? Would our pseudo-qrels method, which relies on the sorted pools, still mimic the real qrels accurately? This question can be addressed by conducting pseudo-qrels experiments with some existing TREC or NTCIR collections and submitted runs.

References

1) Aslam, J.A. and Savell, R.: On the Effectiveness of Evaluating Retrieval Systems in the Absence of Relevance Judgments, *ACM SIGIR 2003 Proceedings*, pp.361–362 (2003).

2) Carterette, B., Kanoulas, E., Aslam, J.A. and Allan, J.: Evaluation over Thousands of Queries, *ACM SIGIR 2008 Proceedings*, pp.651–658 (2008).

3) Cormack, G.V., Palmer, C.R. and Clarke, C.L.A.: Efficient Construction of Large Test Collections, *ACM SIGIR '98 Proceedings*, pp.282–289 (1998).

4) Kando, N.: Overview of the Seventh NTCIR Workshop, *NTCIR-7 Proceeings*, pp. 1–9 (2008).

5) Mitamura, T., Nyberg, E., Shima, H., Kato, T., Mori, T., Lin, C.-Y., Song, R., Lin, C.-J., Sakai, T., Ji, D. and Kando, N.: Overview of the NTCIR-7 ACLIA Tasks: Advanced Cross-Lingual Information Access, *NTCIR-7 Proceedings*, pp.11–25 (2008).

6) Sakai, T.: Evaluating Information Retrieval Metrics based on Bootstrap Hypothesis Tests, *IPSJ Transactions on Databases*, Vol.48, No.SIG 9 (TOD35), pp.11–28 (2007).

7) Sakai, T.: Information Retrieval Test Collections and Evaluation Metrics: A Tutorial (in Japanese), *IPSJ SIG Technical Report 2008-FI-89 / 2008-NL-183*, pp.1–8 (2008).

8) Sakai, T. and Kando, N.: Are Popular Documents More Likely To Be Relevant? A Dive into the ACLIA IR4QA Pools, *EVIA 2008 Proceedings*, pp.8–9 (2008).

9) Sakai, T., Kando, N., Lin, C.-J. Song, R. Shima, H. and Mitamura, T.: NTCIR-7 ACLIA IR4QA Results based on Qrels Version 2 (available at `http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings7/toc_ntcir.html#IR4QA`) (2009).

10) Sakai, T., Kando, N., Lin, C.-J., Mitamura, T., Shima, H., Ji, D., Chen, K.-H. and Nyberg, E.: Overview of the NTCIR-7 ACLIA IR4QA Task (also available at the above URL), *NTCIR-7 Proceedings*, pp.77–114 (2008).

11) Soboroff, I., Nicholas, C. and Cahan, P.: Ranking Retrieval Systems without Relevance Judgments, *ACM SIGIR 2001 Proceedings*, pp.66–73 (2001).

12) Voorhees, E.M.: The Philosophy of Information Retrieval Evaluation, *Proceedings of the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems (LNCS 2406)*, pp.355–370 (2001).

13) Yilmaz, E., Aslam, J. and Robertson, S.: A New Rank Correlation Coefficient for Information Retrieval, *ACM SIGIR 2008 Proceedings*, pp.587–594 (2008).

14) Zobel, J.: How Reliable Are the Results of Large-Scale Information Retrieval Experiments?, *ACM SIGIR '98 Proceedings*, pp.307–314 (1998).

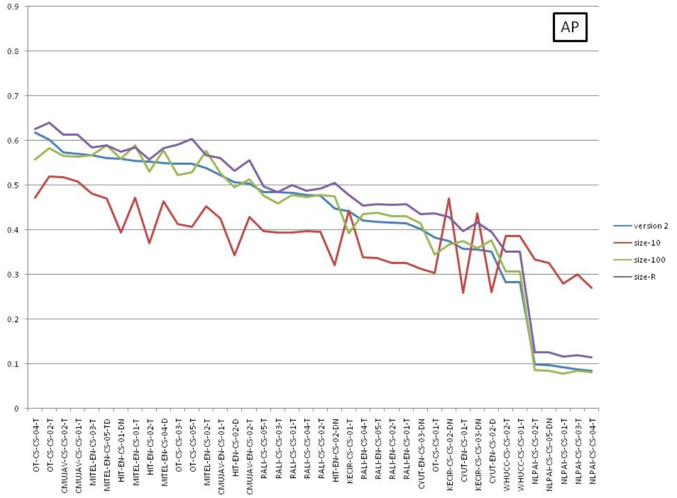

Fig. 4 Qrels version 2 vs. pseudo-qrels: Ranking CS runs by Mean AP.

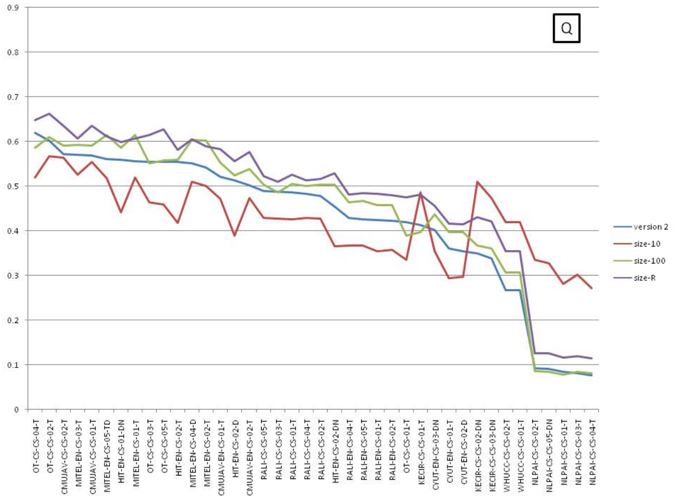

Fig. 5 Qrels version 2 vs. pseudo-qrels: Ranking CS runs by Mean Q.

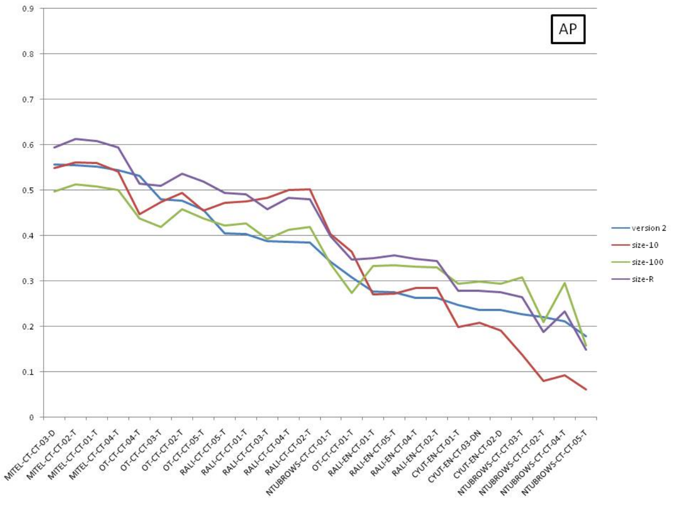

Fig. 6 Qrels version 2 vs. pseudo-qrels: Ranking CT runs by Mean AP.

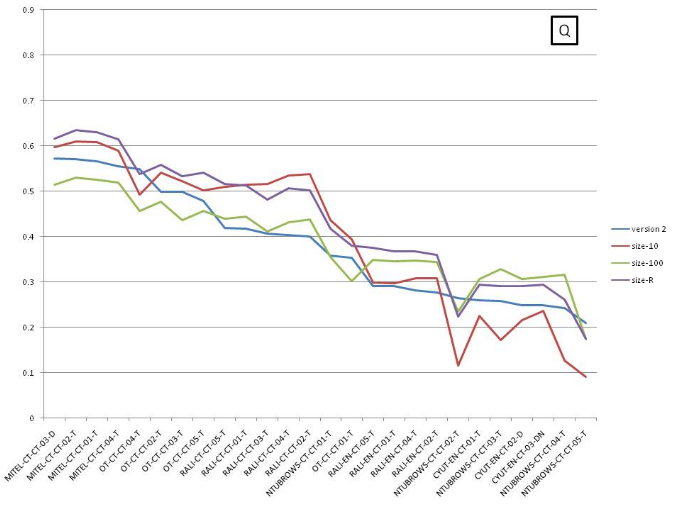

Fig. 7 Qrels version 2 vs. pseudo-qrels: Ranking CT runs by Mean Q.

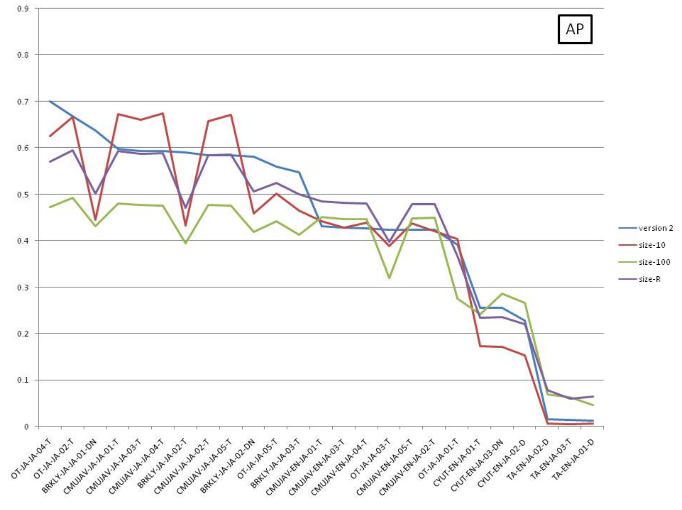

Fig. 8 Qrels version 2 vs. pseudo-qrels: Ranking JA runs by Mean AP.

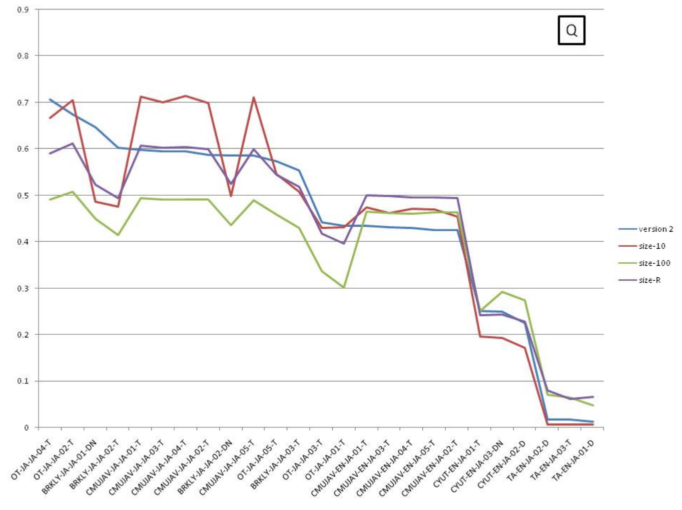

Fig. 9 Qrels version 2 vs. pseudo-qrels: Ranking JA runs by Mean Q.