Length Normalization in the Era of Neural Rankers

Charles L. A. Clarke
Computer Science, University of Waterloo, Canada

ABSTRACT

Document length was rarely a factor in traditional retrieval evaluation metrics. As a result, traditional rankers could take advantage of this property by favoring longer documents, which were more likely to be relevant. Returning a long, non-relevant document was in no way penalized by these traditional metrics, even if in practice the searcher would spend time fruitlessly scanning the document for relevant material. As we show by way of an illustrative experiment, a policy of ignoring document length may have had unfortunate impacts even in traditional contexts. But this policy becomes increasingly inappropriate in an era of neural rankers, where query-document similarity may be computed directly from text. Freed from ranking based on document- and corpus-level statistics, it should be possible to return precisely the required information and no more. Future evaluation efforts and metrics should reflect this goal.

1. TRADITIONAL EVALUATION

Traditional retrieval evaluation metrics, including AP, MRR, NDCG, RBP, ERR, and others, treat all documents equally, regardless of length. In traditional test collections, a relevance judgment applies to a document as a whole, and if a document includes any relevant material at all, the entire document is judged relevant for the purpose of these metrics. Even when graded relevance judgments are available, these relevance grades do not explicitly consider document length. Although it might the case that a long document containing a small amount of relevant material will receive a lower grade than a similarly sized document dedicated to the topic of the query, these grades reflect only the value of relevant material to the searcher, and fail to reflect the time the searcher might waste viewing non-relevant material.

It has long been known that in many traditional test collections longer documents are more likely to be relevant [25]. This observation, together with the length blindness of traditional evaluation metrics, encourages traditional rankers to effectively assign a larger prior to longer documents, even though retrieving a long but non-relevant document may waste a searcher's time.

For example, the BM25 [22] ranking formula still serves as a frequent baseline in current research and as a standard feature in learned rankers. This formula explicitly incorporates a length normalization parameter (b) which works in concert with a term saturation parameter (k_1) to adjust document scores to account for document length. When tuned on traditional test collections, the value of these parameters will reflect a bias for longer documents.

By assigning equal value to all documents, regardless of length, traditional evaluation metrics both fail to appropriately penalize rankers for returning long non-relevant documents, and fail to appropriately reward them for returning long relevant documents. While a longer relevant document may indeed may be more useful to a searcher than a shorter relevant document, since it may treat the associated topic at greater depth, neither this potential value, nor the associated risks, are recognized by traditional evaluation metrics.

2. FROM THERE TO HERE

All of these observations and concerns mattered little during the 18-year hiatus between 1998 and 2016, when relatively modest progress was made on pure content-based ranking [3, 29], i.e., rankers that use only the human language in the document itself as the source of ranking features. During this era some progress did come through exploiting internal document features, such as fields [20] and proximity[16], as well as through new theoretical insights [14, 32], and improvements to pseudo-relevance feedback [1]. But most progress came through the user of externally computed document properties, such as link analysis [5] and user behavior [2], as well as through the development of learned rankers designed to extract benefits from large numbers of carefully engineered features [7]. During this era, research attention was largely focused on Web search, where searcher intent is often navigational or transactional in nature, and document length is less of a consideration [6].

Over this period, some consideration was given to how document length relates to relevance from a theoretical perspective. In particular, Fang et al. [11] define several length-related constraints for retrieval models. One constraint, called LNC2, suggests that "the score of a document should decrease if we add an extra occurrence of a 'non-relevant word' (i.e., a word not in the query)".

3. NEURAL RANKERS

With recent leaps in query-document similarity enabled by neural models [9, 17, 19], research attention has again returned to pure content-based ranking. However, unlike traditional rankers such as BM25, which operate on document-level and collection-level statistics, most neural rankers, in essence, directly compute similarity between relatively short

EVIA 2019, June 10, 2019, Tokyo, Japan.

segments of text. For example, for the Conv-KNRM model, Dai et al. [9] report truncating the body of documents to 50 terms. Similarly, pre-trained BERT operates on a maximum of 500 tokens for both query and document text segments [19]. Yang et al. [30] apply BERT to an end-to-end retrieval task, working at the sentence and paragraph level. In the case of the DUET Model, Mitra et al. [17] report truncating the body of documents to a (relatively long) $1,000$ terms before matching. Rosset et al. [23] extend this work by applying LNC2, along with other constraints, to regularize neural rankers.

Perhaps responding to the characteristics of most neural rankers, some modern test collections focus on re-ranking passages, rather than full documents. For example, instead of full documents, the MS MARCO [18] dataset uses passages extracted from full documents, with relevance assessment determining if the passage contains an answer to the question posed in the query. The TREC CAR task [10] similarly focuses on passages, rather than full documents. These test collections continue to use evaluation metrics that are blind to document length, including MRR and AP.

One notable exception to the above is the DRMM model of Guo et. al [12]. They take the full document into account explicitly for the reasons outlined in the introduction, i.e., that document length may be a factor in relevance and that a short relevance passage renders the entire document relevant. But to satisfy this requirement they work with histograms, that essentially incorporate collection- and document-level statistics into the model.

Yang et al. [29] examine the performance of neural rankers as applied to the TREC Robust04 test collection, which is based on longer documents. Only the DRMM model significantly outperforms a properly tuned BM25 baseline with pseudo-relevance feedback, similar to the one used below. Again, they use AP and precision@20 for the evaluation. It may be that different evaluation metrics and methdologies might alow the novel abilities of neural rankers to become more apparent.

As neural rankers become the primary method for content-based ranking, we have the opportunity to better incorporate document length considerations into our evaluation metrics. Ideally, we would reward rankers that return only relevant information, perhaps by extracting this information from longer documents as needed. One goal of the current paper is to start a conversation on developing and validating these metrics, while recognizing that substantial work has already been done.

4. METRICS AND METHODS

In this short paper, we do not attempt to create a new evaluation metric from scratch. Instead, we compare how tuning impacts two existing metrics, one which treats all documents equally, regardless of length, and one which explicitly attempts to model the impact of document length on the value of a ranking to a searcher. We hypothesize that, as we better reflect document length in our evaluation metric, giving more credit to longer relevant documents and appropriately punishing longer non-relevant documents, the optimal parameter values in a simple ranking formula (BM25) will adjust to reflect this change. We choose BM25 both because it explicitly accommodates document length and because its performance is known to be highly sensitive to parameter tuning [27].

parameter	distribution	range
b	uniform	$(0.0, 1.0)$
k_1	log uniform	$(0.01, 1000.0)$
depth (m)	log uniform	$(8, 32)$
expansions (n)	log uniform	$(4, 32)$
mixing (γ)	uniform	$(0.0, 1.0)$

Table 1: Distributions used for generating random parameter sets.

4.1 Retrieval

The BM25 ranking formula is well established and well known within the search community, but we include the formula for convenience, especially since we will be extensively discussing the tuning of its parameters.

$$\sum \frac{f_{t,d}(k_1+1)}{k_1\left((1-b)+b\left(l_d/l_{avg}\right)\right)+f_{t,d}} \cdot w_t \tag{1}$$

In this formula, $f_{t,d}$ represents the within-document term frequency, w_t is a global term weight, l_d represents the document length, and l_{avg} is the global average document length. The summation is over query terms. The tunable parameter b, where $0 \leq b \leq 1$, explicitly normalizes for document length, with lower values favoring longer documents. The tunable parameter k_1 controls term saturation, i.e. how much each additional term contributes to the overall score.

When applied to traditional test collections, rankers often obtain their best performance through the application of *pseudo-relevance feedback*, where a final document ranking is obtained through a three-stage process: 1) an *initial retrieval* stage, where an initial ranking of the top-m documents is obtained from the original query; 2) a *feedback* stage, where these m documents are analyzed to extract n query expansion terms; and 3) a *final retrieval* stage, where these expansions terms are combined with the original query, weighted according to a mixing parameter γ, to obtain a final ranking. In this paper, we use classic Robertson term selection [21], while noting that Lin [15] reports superior performance using RM3 feedback.

4.2 Parameter tuning

We tune the BM25 parameters b and k_1 separately for the initial and final retrieval stages. The pseudo-relevance parameters m, n, and γ are also all tuned. Reassured by Bergstra et al. [4] we tune these parameters by selecting parameter sets randomly according to the distributions in Table 1. This approach allows us to explore the full range of parameters, both good and terrible.

After selecting an evaluation metric to tune against, we tune one retrieval stage at a time. We first tune the initial retrieval stage, we then use the best resulting retrieval parameters to tune the feedback stage, and finally we use the best resulting feedback parameters to tune the final retrieval stage. At each tuning stage we generate $1,000$ random parameter sets, retaining the best set for the next tuning stage, according to the selected evaluation metric.

4.3 Average Precision (AP)

Average precision (AP) served as the primary evaluation metric for the classic TREC retrieval experiments [28], but has also been applied in non-neural learning-to-rank con-

TREC 7 Topics (parameter tuning)			Tuned on AP best	Tuned on AP worst	Tuned on TBG best	Tuned on TBG worst
1. initial retrieval	parameters	b	0.267	0.969	0.754	0.008
		k_1	0.761	991.411	1.773	353.051
	metrics	AP	0.205	0.064	0.184	.068
		TBG	2.972	1.613	3.399	0.572
2. pseudo-relevance feedback	parameters	depth (m)	19	29	8	30
		expansions (n)	25	31	21	26
		mixing (γ)	0.198	0.932	0.270	0.747
	metrics	AP	0.257	0.203	0.237	0.212
		TBG	2.844	2.384	3.767	3.129
3. final retrieval	parameters	b	0.356	0.003	0.978	0.001
		k_1	0.623	597.365	1.902	173.461
	metrics	AP	0.259	0.068	0.221	0.076
		TBG	2.920	0.192	3.816	0.257
TREC 8 Topics	metrics	AP	0.281		0.214	
		TBG	3.420		3.869	

Table 2: Results of tuning BM25 with pseudo-relevance feedback on the TREC 7 adhoc collection against average precision (AP) vs. time biased gain (TBG). For each step of the tuning process we show both the best and worst parameter settings discovered, although the only best parameters from one step are used for tuning in the next step, i.e., we don't attempt to find the worst overall parameter settings. The last two lines report performance on the TREC 8 adhoc collection using the best parameters from TREC 7 tuning. The AP results are comparable to the best equivalent runs those years.

texts [31], as well as in very recent neural ranking research [19, 15]. Like most classic evaluation metrics, relevance applies at the document level. In the case of AP, relevance is binary — a document is either relevant or not — and a short, focused document is treated as equal to a longer document containing a small relevant passage, along with much non-relevant material. Other traditional evaluation metrics, such as NDCG [13], support multiple relevance grades, but these grades do not directly reflect document length or the relative density of relevant material. A shorter highly relevant document and a much longer highly relevant document are equals, even though the second may provide considerably more information.

Given that neural rankers work directly with raw text, with the ability to directly recognize relevant material, metrics that treat documents containing differing amounts of relevant material equally become even less ideal than they may been twenty years ago. In addition, if neural rankers are applied to extract relevant information, or redact non-relevant information, from longer documents containing a mixture of material, then evaluation metrics should accommodate this benefit. Ideally, a evaluation metric should reflect actual user experience, appropriately rewarding improved outcomes.

4.4 Time Biased Gain (TBG)

Several text- or passage-oriented metrics have been defined over the years, essentially addressing the concerns raised at the end of the previous subsection. For example, the *U-measure* defined by Sakai and Dou [24] provides a unified approach to evaluation, directly accommodating document length, and going beyond ranked lists to support extractive summarization and similar methods for improved result presentation. Similarly, the *Time Biased Gain* metric of Smucker and Clarke [26] effectively simulates a user-traversing a ranked list, thereby imposing greater penalties for returning longer non-relevant documents. This work was extended in Clarke and Smucker [8], where the value extracted from a ranked list is determined by estimating the time spent reading relevant material.

In this short paper, we use TBG as an illustrative example of the type of metric that may better suit the era of neural rankers, although we assume that additional progress will be required, and should be possible, along these lines. While the original paper [26] should be consulted for details, we note that the central idea behind TBG is to model the time taken by the user to traverse a result list and read relevant information. Parameters of the model are determined from user data, collected from laboratory experiments and log data. For parameter tuning purposes, we re-implemented TBG from the details in the paper, which was checked against the original code[1], although we do not identify or penalize duplicate documents, which are less of a problem in the collections we will use for our experiments.

5. EXPERIMENTAL RESULTS

We use some older TREC collections for our experimental work, specifically the "title-only" queries from TREC 7 and TREC-8 [28]. Results are shown in Table 2. Even using this classic ranking formula, the different assumptions underlying the evaluation metrics lead to different optimal parameters. Most notably, the final value for b when tuned against TBG is close to 1, which completely normalizes for document length, so that no special preference is given to longer documents.

The worst columns illustrate the sensitivity of BM25 to parameter tuning, with a value of AP as low as 0.068 possible on the final retrieval step, even when the best discovered values are used in the previous two steps. Quite apart from anything else, this sensitivity should warn against the use of untuned BM25 as an experimental baseline, which is not uncommon in current research [15].

[1] `plg.uwaterloo.ca/~claclark/eval/tbg.pl`

Consider the performance differences on TREC 8 topics, when parameters are tuned using the TREC 7 test collection. When BM25 is tuned on AP vs. TBG, the improvement in AP is over 16%; when BM25 is tuned on TBG vs. AP the improvement in TBG is over 13%.

6. CONCLUSIONS

The primary goal of this paper is to start a conversation at EVIA on retrieval evaluation in the age of neural rankers. Our experiment illustrates that even in the classic era, document length may not have appropriately reflected in evaluation metrics. Given that modern neural rankers are not dependent on document- and collection- level statistics, let us use this opportunity to improve how we view evaluation, encouraging the creation of test collections and methodologies where numeric improvements truly reflect improved user experience.

7. REFERENCES

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D. Smucker, and C. Wade. UMass at TREC 2004: Novelty and HARD. In *13th TREC*, 2004.

[2] E. Agichtein, E. Brill, and S. Dumais. Improving Web search ranking by incorporating user behavior information. In *29th ACM SIGIR*, pages 16–26, 2006.

[3] T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. Improvements that don't add up: Ad-hoc retrieval results since 1998. In *18th ACM CIKM*, pages 601–610, 2009.

[4] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In *24th NIPS*, pages 2546–2554, 2011.

[5] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. In *7th WWW Conference*, pages 107–117, 1998.

[6] A. Broder. A taxonomy of Web search. *SIGIR Forum*, 36(2):3–10, 2002.

[7] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *22nd ICML*, pages 89–96, 2005.

[8] C. L. A. Clarke and M. D. Smucker. Time well spent. In *5th IIiX*, pages 205–214, 2014.

[9] Z. Dai, C. Xiong, J. Callan, and Z. Liu. Convolutional neural networks for soft-matching N-grams in ad-hoc search. In *11th ACM WSDM*, pages 126–134, 2018.

[10] L. Dietz, M. Verma, F. Radlinski, and N. Craswell. TREC complex answer retrieval overview. In *26th TREC*, 2017.

[11] H. Fang, T. Tao, and C. Zhai. Diagnostic evaluation of information retrieval models. *ACM Transactions on Information Systems*, 29(2):7:1–7:42, 2011.

[12] J. Guo, Y. Fan, Q. Ai, and W. B. Croft. A deep relevance matching model for ad-hoc retrieval. In *25th ACM CIKM*, pages 55–64, 2016.

[13] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[14] V. Lavrenko and W. B. Croft. Relevance based language models. In *24th ACM SIGIR*, pages 120–127, 2001.

[15] J. Lin. The neural hype and comparisons against weak baselines. *SIGIR Forum*, 52(2):40–51, 2018.

[16] Y. Lv and C. Zhai. Positional relevance model for pseudo-relevance feedback. In *33rd ACM SIGIR*, pages 579–586, 2010.

[17] B. Mitra, F. Diaz, and N. Craswell. Learning to match using local and distributed representations of text for Web search. In *26th WWW Conference*, pages 1291–1299, 2017.

[18] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016.

[19] R. Nogueira and K. Cho. Passage re-ranking with BERT. *CoRR*, abs/1901.04085, 2019.

[20] S. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In *13th ACM CIKM*, pages 42–49, 2004.

[21] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46(4):359–364, December 1990.

[22] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *17th ACM SIGIR*, pages 232–241, 1994.

[23] C. Rosset, B. Mitra, C. Xiong, N. Craswell, X. Song, and S. Tiwary. An axiomatic approach to regularizing neural ranking models. In *42st ACM SIGIR*, 2019.

[24] T. Sakai and Z. Dou. Summaries, ranked retrieval and sessions: A unified framework for information access evaluation. In *36th ACM SIGIR*, pages 473–482, 2013.

[25] A. Singhal, G. Salton, M. Mitra, and C. Buckley. Document length normalization. *Information Processing & Management*, 32(5):619–633, 1996.

[26] M. D. Smucker and C. L. A. Clarke. Time-based calibration of effectiveness measures. In *35th ACM SIGIR*, pages 95–104, 2012.

[27] A. Trotman, A. Puurula, and B. Burgess. Improvements to BM25 and language models examined. In *2014 ADCS*, pages 58–65, 2014.

[28] E. M. Voorhees and D. K. Harman. The Text REtrieval Conference. In E. M. Voorhees and D. K. Harman, editors, *TREC — Experiment and Evaluation in Information Retrieval*, chapter 1, pages 3–20. MIT Press, Cambridge, Massachusetts, 2005.

[29] W. Yang, K. Lu, P. Yang, and J. Lin. Critically examining the "neural hype": Weak baselines and the additivity of effectiveness gains from neural ranking models. In *42nd ACM SIGIR*, 2019.

[30] W. Yang, Y. Xie, A. Lin, X. Li, L. Tan, K. Xiong, M. Li, and J. Lin. End-to-end open-domain question answering with BERTserini. *CoRR*, abs/1902.01718, 2019.

[31] E. Yilmaz and S. Robertson. On the choice of effectiveness measures for learning to rank. *Information Retrieval*, 13(3):271–290, 2010.

[32] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.