Ranking and Feedback-based Stopping for Recall-centric Document Retrieval

Noah Hollmann and Carsten Eickhoff

ETH Zurich, Zurich 8092, Switzerland

Abstract. Systematic reviews require researchers to identify the entire body of relevant literature. Algorithms that filter the list for manual scanning with nearly perfect recall can significantly decrease the workload. This paper presents a novel stopping criterion that estimates the score-distribution of relevant articles from relevance feedback of random articles (S-D Minimal Sampling). Using 20 training and 30 test topics, we achieve a mean recall of 93.3%, filtering out 59.1% of the articles. This approach achieves higher F2-Scores at significantly reduced manual reviewing work loads. The method is especially suited for scenarios with sufficiently many relevant articles (>5) that can be sampled and employed for relevance feedback.

Keywords: Cutoff problem, Stopping criteria, Total Recall, Medical Information Retrieval, Relevance-Feedback

1 Introduction

Systematic reviews give a comprehensive overview of all published evidence on a given topic. It has been estimated that every year, more than 4000 systematic reviews are conducted and published with each review requiring at least 6-12 months of preparation time [4]. In order to write a systematic review in a first step all related articles have to be collected. Often a huge initial number of articles is retrieved and subsequently filtered by manually scanning each document's abstract. This practice creates a considerable workload for researchers. With medical libraries expanding rapidly it is crucial to find methods that can algorithmically reduce the number of articles that need to be reviewed by domain experts, while not missing any relevant ones. This task is known as the Total Recall Problem in the Information Retrieval community.

Systematic reviews of diagnostic test accuracy (DTA) compare the effectiveness of index tests for a target condition. Filtering relevant studies for DTA reviews has been identified to be exceptionally challenging due to an increased class-imbalance, a broader-than usual target class definition, and a lack of metadata quality *e.g.*, missing abstracts [12]. However, significant advances in this domain are expected to be applicable to other areas as well. Due to unreliable performance, the Cochrane Organization, a leading authority in systematic DTA reviews does not, currently, recommend to use any search filters in the review process [13].

This paper describes ETH Zurich's participation in "Task 2: Technologically Assisted Reviews in Empirical Medicine" at the CLEF eHealth Evaluation lab 2017 [7]. The aim of this task is to find reliable filtering methods for Diagnostic Test Accuracy (DTA) reviews. In a first step a human expert collects a list of PubMed-articles by Boolean search for each topic. The aim of the task is to filter this initial list of articles with total recall. The filtered list can then be reviewed by experts at a lower expenditure of time and resources.

We propose a learning to rank pipeline that extracts information for each article from PubMed, creates numerical features from this information, ranks each article and finally determines a cut-off point on the ranked list based on a novel score distribution approach.

The remainder of this document is structured as follows. Section 3 describes our learning-to-rank system alongside a statistical stopping criterion for manual result list inspection. Section 4 empirically compares the proposed method with a wide range of state-of-the-art baselines. Section 5 discusses a number of qualitative observations and Section 6 concludes with an outlook on future research directions.

2 Related Work

Creating search filters for systemic DTA reviews has been studied, but all approaches failed to retain high recall while also significantly reducing the number of documents that have to be assessed for inclusion.
Ritchie and colleagues (2007) assessed the performance of 23 published and unpublished methodological search filters (Haynes 1994; Devill 2000; Bachmann 2002; Devill 2002b; Shipley 2002; Vincent 2003; Falck-Ytter 2004; Haynes 2004; Critical Appraisal Skills Programme 2006; InterTASC Information Specialists Subgroup 2006) in finding 160 studies in MEDLINE included in one diagnostic review of urinary tract infections. The original review had used sensitive searches across multiple databases to locate studies and had not applied a diagnostic search filter. The sensitivity of the strategies ranged from 20.6% to 86.9%, precision ranged from 1% to 9.4%. The strategy designed by Vincent and colleagues (2003) performed best in terms of the best compromise of sensitivity (86.9%) and precision (3.3%). No strategies had adequate sensitivity for systematic review searching. They identified terms, that were found frequently in relevant articles (such as "sensitivity and specificity", "mass screening") and created a list of search filters from these.
.. identified problems related to creating search filters based on metadata such as abstract, title and MeSH-Headings. They fond that abstracts and MeSH terms are frequently missing in older articles and thus metadata quality is generally low. Also the search class for systematic reviews is very broad with no restrictions on publication types, language etc. Thus an effective algorithm has to tackle the problem of missing data and be flexible in its application.
Lease *et al.* emphasize the similarity of e-discovery, another instance of the total-recall problem in the context of law, to filtering of systematic review articles.

Prior research has found that using automated techniques alone has led to poor performance when using automated methods alone (Grossman and Cormack 2011; Oard et al. 2010).

3 Methodology

In order to judge the relevance of an article for a given query, we propose a learning-to-rank pipeline that extracts information for each candidate article and represents them as dense numerical feature vectors. These vectors can be used to train ranking systems that create ordered lists of articles. In a final step the algorithm decides where to cut off the ranked list for manual inspection.

For each article, we extract the title, abstract, MeSH headings, a list of publication types and the publication language via the NCBI EUtilities[1]. MeSH headings are a list of tags from a comprehensive controlled vocabulary for indexing journal articles in the life sciences. While a title is available for all papers, the abstract is missing in many cases (12.2% on the training set).

3.1 Feature Extraction

From this data we extract a number of features and group them into two categories: dynamic and static features. While static features only depend on the article, dynamic features depend on the article in relation to each query. Static features will capture the aptness of an article to be included in any systematic review, while dynamic features express its relevance for the query at hand. Static features include a similarity score between each article's 128-dimensional Doc2Vec embedding [11] and an average embedding of relevant documents, a number of statistical features denoting how likely publication type and language are relevant, the publication year and the number of words in abstract, title and MeSH headings. Dynamic features include a tf-idf similarity measure of query text and query title to document title, abstract and MeSH headings. The tf-idf score is calculated using Lucene and is optimized using a stop word filter. Also included in the dynamic features is the cosine similarity of the document embeddings between the query and various document fields. For this purpose also the query and query title are mapped to a vector using Doc2Vec. The method includes a total of 51 features the respective effectiveness of which will be discussed in Section 4.

Static relevancy using document embedding In previous search filters the occurrence of words that frequently appear in relevant documents for arbitrary DTA reviews was used to filter the articles [14]. We propose a deep learning approach that models the similarity to frequent words by creating a document

[1] https://www.ncbi.nlm.nih.gov/books/NBK25500/

embedding of a common relevant document.
We create the vectors

$$Doc2Vec_q^+ = \sum_{a \in A_q^+} \frac{d2v(a)}{|A_q^+|}$$

$$Doc2Vec_q^- = \sum_{a \in A_q^-} \frac{d2v(a)}{|A_q^-|}$$

where $d2v$ denotes the Doc2Vec representation of an article. Now we create a vector that averages document representations of relevant articles from training topics Q

$$Doc2Vec^+ = \sum_{q \in Q} \frac{Doc2Vec_q^+}{|Q|}$$

This vector contains the word embedding of the average relevant article in our training set. We expect this vector to produce words that frequently appear in relevant articles and that are similar to the ones proposed by Vincent *et al.* [14]. The 20 words most likely contained in our document embedding $Doc2Vec^+$ closely resemble the list proposed by Vincent *et al.*:

> chronic, abnormal, clinically, diagnosis, complication, diagnose, patients, treatment, diagnosing, non-invasive, patient, dysfunction, tissue, symptomatic, abnormalities, minimally, treatments, diagnostic, complications, insufficiency

Similarly to modelling this centroid of relevancy, we can subtract the average irrelevant article across training queries, giving:

$$Doc2Vec^+ = \sum_{q \in Q} \frac{Doc2Vec_q^+ - Doc2Vec_q^+}{|Q|}$$

If we generate related words we get:

> diagnosis, non-invasive, diagnose, diagnosing, diagnostic, imaging, clinically, patients, patient, chronic, abnormal, minimally, treatment, complication, assess, noninvasive, evaluating, helpful, assessing, scans

The similarity of title, abstract and MeSH headings to vectors created by both methods become ranking features. We call the first approach the *classic* method, the second one the *difference* method. This score is calculated for title, abstract and MeSH headings, respectively.

MetaMap MetaMap[2] is a tool that maps biomedical text to the UMLS Metathesaurus using symbolic, natural-language processing (NLP) and computational-linguistic techniques. It is a state-of-the-art library that has shown to be highly

[2] metamap.nlm.nih.gov

effective [3]. Mapping text to UMLS Context Unique Identifiers (CUIs) will reduce ambiguity of medical documents and also tag the CUIs with a semantic group. We measure the tf-idf score of the CUIs identified in query title and query to the CUIs identified in the document content fields using a BM25 model.

Statistical features The publication type probability $P(T)$ is the probability that a given publication type is included in the list of publication types of a relevant article. The publication type score $S(T)$ is calculated such that $S(T) = P(T) * N(T)$, where $N(T)$ is the number of times that publication type was relevant.

Treating missing values After creating features for each article and its topic combination, we process the feature file to account for missing data. For missing abstracts, we set all abstract related features to the average on the training set and do the same for MeSH headings. We tried setting these features to an unused value such as -100, which produced worse results.

3.2 Ranking Models

On the basis of the previously described features, we use Ranklib[3] to train a number of rankers and validate their performance. After evaluating a broad range of models including coordinate ascent, MART [8], AdaRank [16], RandomForests [5] and LambdaMART [15], we decided for a straightforward coordinate ascent model that yielded the most reliable results on the given data. Unless stated otherwise, all further experiments in this paper are based on this ranking model.

3.3 Stopping Criteria

While generating a model to score the relevancy of articles is essential to finding a set of relevant articles, it is equally important to find the right point to stop retrieving more documents. In order to threshold the ranking, we propose a naive baseline technique that cuts off at a fixed rank and an extended baseline that will decide based solely on previous distributions of relevant articles according to their retrieval model score.

In order to find a suitable cutoff method we need to define a metric to optimize for. In a systematic review we emphasize recall over precision and reliability of optimality. Additionally, we would like to use as little manual relevance feedback as possible for our ranking. The optimal cost C_{opt} for some optimal rank r trading off between recall, precision and relevance feedback can be found according to:

$$C_{opt} = \min_{\forall r \in (1,|A|)} f(r, |A^+_{<r}|, |A^+_{>r}|, |R|)$$

[3] https://sourceforge.net/p/lemur/wiki/RankLib/

for any weighting formula f, a list of articles A, a list of positive articles that are retrieved $A^+_{<r}$, a list of positive articles that are not retrieved $A^+_{<r}$ and a set of documents the method got relevance feedback for R. A weighting formula that weights precision with α and recall with β looks the following:

$$f(r, |A^+_{<r}|, |A^+_{>r}|, |R|) = \alpha * (1 - \frac{|A^+_{<r}|}{r}) + \beta * (1 - \frac{|A^+_{<r}|}{|A^+_{>r}|})$$

Static cutoff models We use two very simple baseline models that cut off the list at a fixed rank $r*$ or a fixed score $s*$ for every topic. The parameters $r*$ and $s*$ can be fit on the training topics. We use an improved static method that normalizes the scores for each ranking linearly, which yields better results than the original version.

BMI Method We take as another baseline method the default TREC rule for stopping to read a ranked list using relevance feedback for each document. We stop when the number of documents reviewed exceeds $2R^+1000$, where R^+ is the number of relevant documents retrieved so far.

Knee Method This method locates a so-called "knee" or negative inflection point in the gain curve of relevant documents. The gain curve indicates for each rank x how many relevant documents were found up to that rank. The method stops when the slope following the knee is less than $\frac{1}{\alpha}$ of the slope before the knee and the index is higher than some β.

Classic S-D Method Score distributions (S-Ds) of relevant and irrelevant documents have been studied since the early days of IR. By modelling the score-distribution of relevant documents P_+ for a topic we can estimate the number of relevant documents $|A^+_{<r}|$ that are retrieved until rank r using

$$|A^+_{<r}| = \int_1^r P_+(score(x))\,\mathrm{d}x$$

where $score(x)$ denotes the score at rank x. We can approximate the best cutoff according to some metric that uses $|A^+_{<r^*}|$ for all r^* and r. Using the distribution we can estimate a cost function for all possible cutoff points and select the best position. We follow Arampatzis *et al.* [2] in modelling the distribution of relevant documents by a Gaussian. In this "classic" S-D Method, we learn the distribution $P^{learn}_+ \sim \mathcal{N}(\mu, \sigma)$ by fitting a Gaussian to the score-distribution of relevant articles in the training set and make the simplifying assumption that the same distribution will also hold for previously unseen test topics.

Feedback-Based S-D Method In practice, however, the true distribution may vary strongly for each topic. Instead of using a distribution P^{learn}_+ that was

trained beforehand, we can sample some documents $X \subseteq A$ randomly with a sampling distribution $P_{sample}(a)$ for $a \in A$ that assigns a probability of being sampled for each article. We approximate the Gaussian distribution $P_+^{feedback}$ by fitting it on the score distribution of relevant sampled articles $X_+ = \{rf(x) = 1 | x =\in X\}$ where $rf(x) = 1$ if x is relevant and 0 otherwise.

In the simplest case we give each document the same probability $\frac{1}{|A|}$ to be sampled and fit a Gaussian distribution $P_+^{feedback}$ on X_+. However, notice that we do not use the irrelevant articles at all. Sampling irrelevant documents will create additional work and may, in some cost functions, be penalized for the use of relevance feedback as well (*e.g.* $CostUniRF$). It is therefore desirable to ask for feedback on as few irrelevant articles as possible. However, since the classes are very imbalanced, if we sample each document with uniform probability we will get much more irrelevant documents than relevant ones. We correct for this observation by boosting the sampling-probability of articles with a high score which will increase the probability to get feedback on a relevant article. We use a sampling distribution $P_{sample}(s)$ that assigns a sampling probability according to the score $s = score(x)$ of an article x now. During sampling a random score s with probability $P_{sample}(s)$ is drawn and the document with the closest score is sampled.

If we try to fit a Gaussian distribution to the data X_+ after the optimized sampling step we notice that the mean of the resulting distribution is biased towards higher scores. A document with score s will be in X with the sampling probability $P_{sample}(s)$ instead of the true probability of a score in the entire data set $P_{overall}(s)$. Thus, a document with score s will be in the sampled data X with $P_{sample}(s)/P_{overall}(s)$ times the true probability. A distribution that is fit on the data X will therefore show approximately the same bias. In order to remove this bias from the sampled data X we discretize the scores into N chunks $Chunk = [1, .., N]$ associating a score $score_c(n) = s_0 + \frac{n}{ns_n}$ from the lowest score s_0 to the highest score s_n to the pieces. For each chunk we calculate the sampling bias

$$B(n) = \frac{P_{overall}(score_c(n))}{P_{sample}(score_c(n)}, \ n \in Chunk$$

, where $P_{overall}$ is obtained for each score by fitting a Gaussian to the score distribution of all articles A. The bias $B(n)$ indicates how much more often a data point in the chunk n was sampled relative to how often it would be sampled if every document had the same probability. Now we obtain an unbiased $\hat{P}_+ \sim \mathcal{N}(\hat{mu}, \hat{\sigma})$ by fitting a Gaussian to the n data points $\frac{P_{sample}(chunk(n))}{B(n)} \ \forall n \in Chunk$.

The standard deviation $\hat{\sigma}$ was found by using the data points of the n chunks instead of the raw sampled data X. We find a better standard deviation σ^* by iteratively testing the fit of $P_+^* \sim \mathcal{N}(\hat{mu}, \sigma^*)$ on the sampled data X and selecting the best σ^*. We test the method with two sampling distributions: the uniform distribution and a triangular distribution with $P_{triangular}(s) = \gamma * (s + \delta)$ with γ such that $P_{triangular}(s)$ is a probability distribution and the offset δ such that $P(s) > 0$ for all scores.

Some topics will have very few relevant documents. In the most extreme case just a single one, making it impossible to make a robust estimate. A very low number of relevant documents from sampling will result in a high variance in the score distribution of the relevant samples which will decreases the similarity of our predicted score distribution to the true one. For a very low number of relevant sampled articles $|A^+_{sampled}| < \alpha$ the method will perform worse than some other method M' for some parameter α (*e.g.* 4). Fitting a Gaussian on the relevant articles requires at least two relevant articles in the sampling phase. Thus, in both cases the method should use the alternative technique M^* to handle this run. The method M^* can be any of the methods described before. We introduce a parameter β that defines the sampling rate $\beta = \frac{|A_{sampled}|}{|A|}$ of sampled articles and use the fixed score method if $|A_{sampled}| < \alpha^*$ for some optimal α^* on the training set.

4 Experiments

4.1 Data set

The experiments are evaluated on the data set provided by the CLEF 2017 Task 2 eHealth Challenge [10, 7]. The models are trained on 20 topics with a total of 125k articles (∼2.5% relevant) and then evaluated on 30 topics with overall about 120k articles. Each topic consists of a query and a title, while each article contains a PubMed-ID and its relevancy. A relevant document corresponds to a document that was selected by a human exert to be possibly included in the systematic review based only on the abstract and title. In a next step these documents are filtered again according to the content of their full documents filtering out about two thirds again. We do not predict the final inclusion in a systematic review but rather the relevancy of a document according to its abstract.

4.2 Evaluation Metrics

Average Precision & Mean Average Precision (MAP) MAP measures the quality of a ranking including the last documents in the ranking. It is apt to evaluate recall-centric systems in which the ordering of the last documents still matters. Notice that mean average precision is greatly influenced by the number of relevant documents contained in the ranking. A random ranking with 50% relevant documents will achieve an MAP-score of 0.5. A perfectly ordered ranking with 5% relevant documents will only achieve an MAP-score of ~ 0.2. Average Precision will be most useful to compare different models for the same data and hardly give an absolute performance measure.

Work Saved over Sampling (WSS) WSS is an intuitive measure that indicates how much less work $w(r)$ has to be done to achieve a recall r on the set of articles A if an optimal cutoff is applied to the ranking, compared to an

unordered list. Since in an unordered list we need to examine a fraction of r documents to achieve recall r we get:

$$WSS(r) = r - \frac{w(r)}{|A|}$$

CLEF 2017 Task 2 Uniform Cost CLEF eHealth Challenge Task 2 2017 provides α, β, and γ-parametrized cost based metric that measures the performance of a ranking of documents $|A|$ with relevant documents A^+ that is cut off at rank r. The cost depends on the amount of relevance feedback used $|R|$ weighted by γ, the effort $r - |R|$ to review the documents weighted by α and the share of relevant documents missed $\frac{|A^+_{>r}|}{|A^+|}$ weighted by $\beta(|A| - r)$.

$$C = \alpha(r - |R|) + \beta(|A| - r)(\frac{|A^+_{>r}|}{|A^+|}) + \gamma|R|$$

The weights proposed by CLEF are $\alpha = 1$, $\beta = 2$, $\gamma = 2$, we denote the measure as $CostUniRF$. In a more realistic setting we would not get punished for using relevance feedback ($\gamma = \alpha$), which we will simply call $CostUni$. Note that the official evaluation script assigns $\gamma = 3$, likely because a document using relevance feedback has cost α for being shown plus cost γ for the feedback. We use $\gamma = 2$ as indicated on the official evaluation measure description.

4.3 Results

Table 1 reports WSS@95, precision, recall and last relevant document scores obtained for each of the 30 test topics. Unless stated otherwise we use the minimal feedback based S-D method as a stopping criterion. The mean WSS@95 score is 0.544 at precision 0.091 and recall 0.933. Precision and recall compare favorably to the search filters that are reviewed in the Cochrane handbook [13], albeit on different datasets. Howard *et al.* [9] achieve a WSS@95 of 0.488 on 15 topics using PubMed articles and Cohen [6] reports WSS@95 score 0.408. These comparisons suggest a reliable performance, but do not replace a true side-by-side comparison. We achieve a recall of 100% on 11 topics, while the lowest recall is at 65.2% followed 70.0% and 86.0%. Here recall can be traded off for precision by raising the penalty of missing a relevant document in the cost function that is optimized by the cutoff method.

Features We assess the descriptive power of features from three different content fields: Abstracts, Title and MeSH headings (see Table 2) by training our model only using features that are derived from one content field at a time, excluding all other sources of evidence. We find MeSH headings to only insignificantly improve the performance of our ranking. Abstracts are the best source for

Table 1. Overview of per topic performance. N refers to the number of documents in the topic and Included refers to the number of articles judged relevant after abstract screening.

Topic	N	Included	WSS@100	WSS@95	Last Rel.	Cutoff	Recall	Precision
CD010633	1575	4 (0.25%)	91.4%	96.4%	55	565	100.0%	0.7%
CD012019	10319	3 (0.029%)	71.9%	76.9%	2379	3250	100.0%	0.1%
CD010339	12809	114 (0.89%)	49.3%	27.1%	9331	2844	86.0%	3.4%
CD009786	2067	10 (0.48%)	42.7%	47.7%	1080	180	70.0%	3.9%
CD009185	1617	92 (5.7%)	55.4%	47.9%	841	702	98.9%	13.0%
CD010276	5497	54 (0.98%)	63.6%	60.3%	2182	1410	94.4%	3.6%
CD011145	10874	202 (1.9%)	59.9%	20.6%	8633	2598	87.6%	6.8%
CD010772	318	47 (15%)	18.4%	15.5%	266	96	85.1%	41.7%
CD010653	8004	45 (0.56%)	60.6%	17.8%	6574	3291	95.6%	1.3%
CD010775	243	11 (4.5%)	69.3%	74.3%	61	105	100.0%	10.5%
CD010896	171	6 (3.5%)	50.6%	55.6%	74	99	100.0%	6.1%
CD008803	5222	99 (1.9%)	62.3%	48.4%	2691	1948	98.0%	5.0%
CD009519	5973	104 (1.7%)	75.8%	61.6%	2291	1348	98.1%	7.6%
CD007431	2076	24 (1.2%)	30.1%	21.0%	1638	1594	95.8%	1.4%
CD009579	6457	138 (2.1%)	56.7%	28.9%	4590	2215	93.5%	5.8%
CD009135	793	77 (9.7%)	20.9%	12.8%	689	252	87.0%	26.6%
CD010705	116	23 (20%)	12.5%	0.9%	112	49	65.2%	30.6%
CD008782	10509	45 (0.43%)	79.3%	83.6%	1724	3205	100.0%	1.4%
CD008760	66	12 (18%)	43.4%	48.4%	32	43	100.0%	27.9%
CD009551	1913	46 (2.4%)	79.0%	81.1%	361	221	84.8%	17.6%
CD009372	2250	25 (1.1%)	72.5%	73.8%	589	425	88.0%	5.2%
CD010023	983	52 (5.3%)	43.6%	43.8%	550	586	100.0%	8.9%
CD010386	627	2 (0.32%)	77.6%	82.6%	108	329	100.0%	0.6%
CD010783	10907	30 (0.28%)	75.3%	76.5%	2557	3430	100.0%	0.9%
CD010860	96	7 (7.3%)	50.3%	55.3%	41	42	100.0%	16.7%
CD010542	350	20 (5.7%)	4.8%	5.7%	327	280	90.0%	6.4%
CD008081	972	26 (2.7%)	41.6%	34.1%	638	532	96.2%	4.7%
CD010173	5497	23 (0.42%)	76.1%	78.9%	1156	1654	100.0%	1.4%
CD009925	6533	460 (7%)	52.0%	13.6%	5642	2027	86.7%	19.7%
CD009647	2787	56 (2%)	45.8%	23.6%	2128	1674	98.2%	3.3%
Average			47.2%	54.4%			93.3%	9.4 %

Table 2. Feature Effectiveness measured as MAP using only these features

Feature name	Description	MAP
Abstract Features	tf-idf, Gensim Similarity, Gensim Relevancy, Number of words	0.2612
Title Features		0.2053
MeSH Heading Features		0.1386
Dynamic Features	tf-idf, Gensim Similarity	0.2319
Static Features	Gensim Relevancy, Number of words, Language, Publication type	0.148
Overall		0.2866
Random		0.0479

ranking the documents even though they might not be present in some cases. We also compare dynamic features, that show the similarity of query and document with static features, that reflect the general aptness of a document for a systematic DTA review in Table 2. Using only dynamic features yields a MAP score of 0.239, while using only static features yields a MAP score of 0.148. Adding both feature groups together results in a MAP of 0.2866.

We further asses the effectiveness of six different feature groups: Gensim Similarities, tf-idf, Metamap, Gensim Relevancies, language and publication types in Table 3. For these features we estimate the importance of a feature group by observing the effect of systematically removing one group at a time and comparing the results to the original results obtained using all features. We find features that are not text related such as language and publication type will only weakly affect the ranking. Also tf-idf achieves the highest gain among the three methods to estimate similarity of a query to a document, even though it does not natively consider synonyms and word combinations.

Stopping Criteria Table 4 examines the effectiveness of eight cutoff methods according to the evaluation measures above. We find the sampling S-D methods to be more effective than all other methods on F2-Score, $CostUniRf$ and $CostUni$, while $CostUniRf$ was optimized by it. Minimal S-D sampling proves 11.2% more effective on the average F2-Score than the best non-S-D sampling method on average. The average cost $CostUni$ is 17.1% lower than on the best non S-D sampling method. Using $CostUniRf$ and thus penalizing relevance feedback the method is 9.2% stronger than the second best contestant. The cost of Minimal S-D sampling is 4.3% lower than using uniform S-D sampling.

The parameters of our stopping criteria were optimized on the 20 available

Table 3. Feature Effectiveness measured as MAP using only these features

Feature name	MAP	MAP Gain
Gensim Similarity	0.1337	+0.0056
TD/IDF	0.2254	+0.0402
Metamap	0.2035	+0.0044
Gensim Relevancy Method 1	0.1439	+0.033
Gensim Relevancy Method 2	0.0724	-0.055
Gensim Relevancy Both	0.1444	+0.026
Language	0.0497	+0.0015
Publication Type	0.0986	+0.003

Table 4. Comparison of Stopping Criteria

Model	Recall	Precision	F2-Score	$\varnothing CostUniRf$	$\varnothing CostUni$
Fixed Rank	93.5%	6.2%	0.11067	2041	2041
Fixed Score	94.5%	7.7%	0.13041	1903	1903
Knee Method	89.0%	7.8%	0.13581	3179	2386
BMI Method	91.3%	6.8%	0.12064	2896	2106
S-D Classic	98.9%	5.7%	0.10011	2722	2722
S-D List Sampling	93.7%	8.5%	0.14137	1847	2095
S-D Uniform Sampling	94.8%	7.9%	0.13708	1830	1579
S-D Minimal Sampling	94.2%	9.2%	0.14978	1754	1529
Optimal	94.1%	11.0%	0.18045	1263	1263

training topics. This yielded an optimal average normalized score of 1.45 and an optimal cutoff rank of 1410. For the knee method we found a slope after the knee parameter of $\alpha = 9$ and the minimum cutoff rank $\beta = 500$ to be optimal. For the relevance-based S-D methods we found the sampling $\alpha = 5\%$ and the minimum number of feedback $\beta = 4$.

Significance of results The small sample size to test our methods demands a closer inspection of the significance of our results. Especially the Fixed-score method performs well and it should be tested if the S-D Minimal Sampling method improves on it with statistical significance. Both methods are optimized towards minimizing $CostUni$, which will be the attribute to compare for both methods. Figure 1 shows the $CostUni$ distribution test set topics. Using a paired one-sided t-test on the 2% significance level we confirm that $CostUni$ is significantly lower for S-D based sampling than the Fixed score method. However we can not confirm such a result on the metric $CostUniRF$ where the method is punished for using relevance feedback.

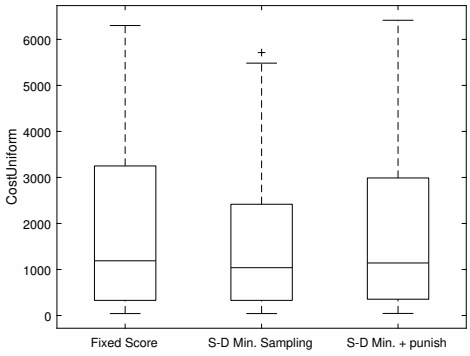

Fig. 1. Distribution of Uniform Cost for Fixed Score and S-D Minimal Sampling. The right most box shows the Uniform Cost when relevance feedback is punished on top ($CostUniRF$). The line in each box depicts the median and the boxes indicate the 25th and 75th percentiles, respectively.

Fit to normal distribution The S-D based stopping methods assume the score distribution of relevant documents P_+ to be Gaussian distributed. Figure 2 visualizes the fit of a standard normal distribution to the summed up and normalized score distributions of each topic. Aside from the intuitively appealing fit, we conduct a Kolmogorov-Smirnov test of normality [1]. We find that normality is retained at $p \leq 5\%$ significance level for all topics.

Correction of distributions in S-D Sampling Stopping Criterion The relevance-based S-D methods rely on removing the sampling bias from the data.

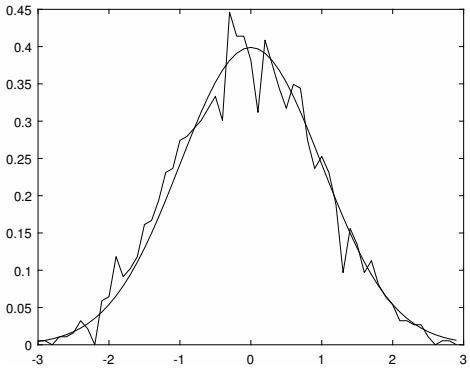

Fig. 2. Shape of positive score distribution compared to normal gaussian

Figure 3 compares the distribution before and after being corrected on some examples. We observe the goodness-of-fit to be much higher and very close to the actual distribution after the correction. By increasing the amount of relevance feedback we can increase the goodness-of-fit of our distribution to that of the best normal distribution.

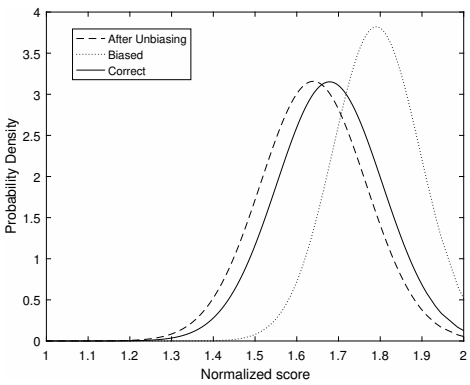

Fig. 3. Example distribution correction CD009185. The biased distribution is skewed towards higher scores compared to the correct distribution.

Correlation of Feedback Size and Cutoff Quality in S-D Sampling Stopping Criterion Figure 4 shows the correlation of $CostUniform$ of our algorithm against the number of relevant articles sampled in each run. We evaluate the same ten topics multiple times, each time sampling until we reach N relevant documents. As expected, when the number of sampled relevant documents increases the cost of the run decreases. Increasing the sample size will

decrease the variance and thus improve the predictive quality of the method. However, if we continue increasing this number the cost of sampling will grow as it will be harder to find more relevant documents.

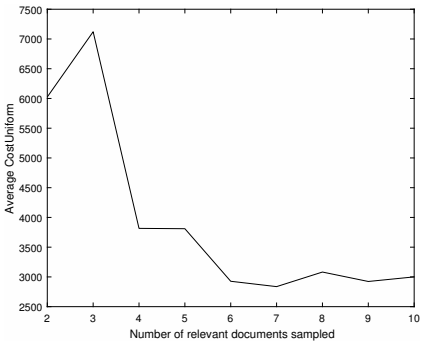

Fig. 4. Mean cost on the test topics when we sample until we found N relevant topics, while N is on the x-axis.

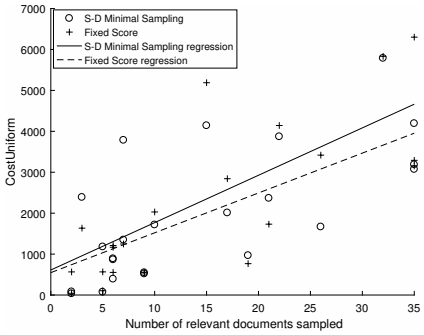

Fig. 5. Correlation of CostUniform and number of relevant documents sampled by the minimal sampling method on the 30 test topics

5 Discussion

5.1 Variance between topics and the need for relevance feedback

Static features that do not depend on the topic query but only on the article can be observed to perform much worse in predicting the relevance of an article. Dynamic features however rely heavily on the initial query which is created by human experts using the system and will thus be written in a very different way for each topic. Also, researchers have a focus on different properties of a document. Figure 6 shows relevant articles from three topics. They are arranged on the plane such that we can see the sum of dynamic features on one axis and the sum of static features on the other. As predicted by the above hypothesis, we see that the preference for inclusion for these two feature groups is very different. In one topic A three articles are included that have low static features, while in topic B there are only articles with high static features. Also probably due to a query that was created very differently, we observe low dynamic features in topic B, but much higher ones in topic A.

An effective system will need to make use of relevance feedback for the articles in order to be able to capture the great differences between each topic and rank the documents more accurately. Human experts will also judge differently which articles need a full content scan and which can be refused immediately. This is reflected in the wide score range of relevant documents that we observe on

the data in absolute numbers (2 to 460) as well as relative numbers (0.029% to 20%). In addition, the quality of a ranking will vary depending on medical focus and the initial query. Figure 7 shows the scores of relevant documents in the 30 test topics. As expected each topic shows a unique score distribution with very different means. We conclude that the cutoff point will be hard to predict using the scores or ranking position without using any relevance feedback. Systems using relevance feedback rely on sampling some relevant documents which is hard due to the high class imbalance in the data of systematic DTA reviews [12]. When sampling randomly one might need to include up to 3000 irrelevant documents to sample a single relevant document. Therefore, a system using relevance feedback needs a high initial ranking quality and should use feedback mostly at the top of this ranking.

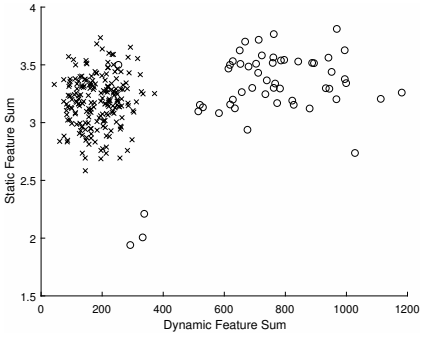

Fig. 6. The distribution of relevant articles on dynamic and static features for two selected topics. The dots correspond to CD010276, the crosses to CD011145

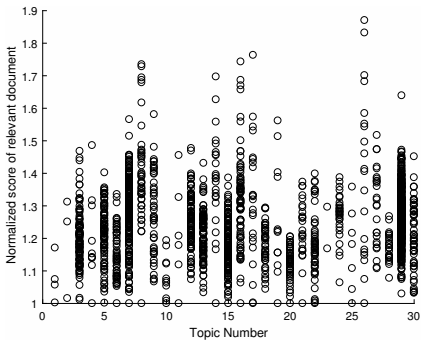

Fig. 7. The distribution of relevant articles on the scores for the topics in the test set

5.2 Missing Metadata

Petersen *et al.* [12] hypothesize that missing metadata is one of the main reasons that systematic DTA reviews are difficult to support with an IR system. In our experiments abstracts have shown to be by far the most effective field in predicting the relevance of a document, however they are missing in 12.3% of all documents in the test data. We suppose that abstracts that are missing in our records were available to the researchers judging their relevance. Thus the probability that a document is relevant should not depend on the presence or absence of abstracts. Figure 8 shows that our model fulfills this property. Documents that lie on the x-axis do not have an abstract text in our data, but their frequency at the end of the ranking is similar to that of documents with an abstract text. Also, the length of the abstract text does not seem to influence the relevancy in our model.

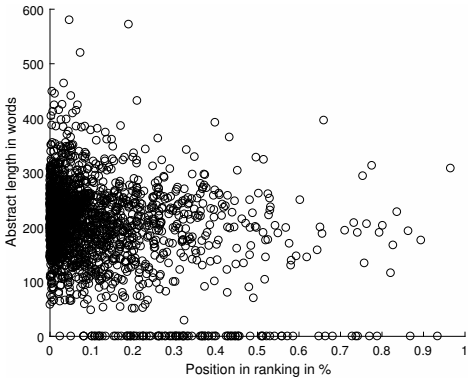

Fig. 8. Each point represents a relevant document. The x-axis shows the position of the document in the ranking and the y-axis the number of words in the abstract. The documents that lie on the x-axis are missing the abstract text in our data.

6 Conclusion

In this paper, we present a recall-centric learning-to-rank scheme accompanied by a statistical cutoff criterion that identifies the optimal point for stopping human inspection of results by estimating the score-distribution of relevant documents in a biased sampling process. Our experiments show that this approach is able to reduce the size of the ranked list by more than half while retaining a recall close to 95% without using relevance feedback in the ranking step. This level of performance significantly exceeds the results obtained by traditional cutoff methods.

In the future, we will move beyond the currently employed simple Gaussian score distributions in favor of more accurate approximations of the true distribution of relevance. Additionally, we plan to further evaluate this method in other recall-driven domains such as e-discovery.

References

1. Ahrens, H.: Pearson, e. s., and h. o. hartley (edit.): Biometrika tables for statisticians vol. i, 3. edition. university press, cambridge 1966. Biometrische Zeitschrift 10(3), 226–226 (1968), https://doi.org/10.1002/bimj.19680100309
2. Arampatzis, A., Kamps, J., Robertson, S.: Where to stop reading a ranked list?: Threshold optimization using truncated score distributions. In: Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 524–531. SIGIR '09, ACM, New York, NY, USA (2009)
3. Aronson, A.R.: Effective mapping of biomedical text to the umls metathesaurus: the metamap program. Proc AMIA Symp pp. 17–21 (2001)
4. Bastian, H., Glasziou, P., Chalmers, I.: Seventy-five trials and eleven systematic reviews a day: How will we ever keep up? PLoS Medicine 7(9), e1000326 (sep 2010), https://doi.org/10.1371/journal.pmed.1000326

5. Breiman, L.: Machine Learning 45(1), 5–32 (2001), https://doi.org/10.1023/a:1010933404324
6. Cohen, A.M.: Performance of support-vector-machine-based classification on 15 systematic review topics evaluated with the wss@95 measure: Table 1. Journal of the American Medical Informatics Association 18(1), 104 (jan 2011), https://doi.org/10.1136/jamia.2010.008177
7. E. Kanoulas, D. Li, L.A., Spijker, R.: Clef 2017 technologically assisted reviews in empirical medicine overview. In: Working Notes of CLEF 2017 - Conference and Labs of the Evaluation forum. CEUR Workshop Proceedings (2017)
8. Friedman, J.H.: Greedy function approximation: A gradient boosting machine. Annals of Statistics 29, 1189–1232 (2000)
9. Howard, B.E., Phillips, J., Miller, K., Tandon, A., Mav, D., Shah, M.R., Holmgren, S., Pelch, K.E., Walker, V., Rooney, A.A., Macleod, M., Shah, R.R., Thayer, K.: Swift-review: a text-mining workbench for systematic review. Systematic Reviews 5(1) (may 2016), https://doi.org/10.1186/s13643-016-0263-z
10. Lorraine Goeuriot, Liadh Kelly, H.S.A.N.A.R.E.K.R.S.J.P., Zuccon, G.: Clef 2017 ehealth evaluation lab overview. In: CLEF 2017 - 8th Conference and Labs of the Evaluation Forum, Lecture Notes in Computer Science (LNCS). Springer (2017)
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Neural and Information Processing System (NIPS) (2013), https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf
12. Petersen, H., Poon, J., Poon, S.K., Loy, C.: Increased workload for systematic review literature searches of diagnostic tests compared with treatments: Challenges and opportunities. JMIR Medical Informatics 2(1), e11 (may 2014), https://doi.org/10.2196/medinform.3037
13. de Vet HCW, Eisinga A, R.I.A.B.P.D.: Chapter 7: Searching for studies. in: Cochrane handbook for systematic reviews of diagnostic test accuracy (2008)
14. Vincent, S., Greenley, S., Beaven, O.: Clinical evidence diagnosis: developing a sensitive search strategy to retrieve diagnostic studies on deep vein thrombosis: a pragmatic approach. Health Information & Libraries Journal 20(3), 150–159 (aug 2003), https://doi.org/10.1046/j.1365-2532.2003.00427.x
15. Wu, Q., Burges, C.J.C., Svore, K.M., Gao, J.: Adapting boosting for information retrieval measures. Information Retrieval 13(3), 254–270 (sep 2009), https://doi.org/10.1007/s10791-009-9112-1
16. Xu, J., Li, H.: Adarank: A boosting algorithm for information retrieval. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 391–398. SIGIR '07, ACM, New York, NY, USA (2007), http://doi.acm.org/10.1145/1277741.1277809