Extended Expectation Maximization for Inferring Score Distributions

Keshi Dai[1], Virgil Pavlu[1], Evangelos Kanoulas[2], and Javed A. Aslam[1] *

[1]College of Computer and Information Science
Northeastern University, Boston, USA
{daikeshi,vip,jaa}@ccs.neu.edu
[2]Information School, University of Sheffield, Sheffield, UK
e.kanoulas@sheffield.ac.uk

Abstract. Inferring the distributions of relevant and nonrelevant documents over a ranked list of scored documents returned by a retrieval system has a broad range of applications including information filtering, recall-oriented retrieval, metasearch, and distributed IR. Typically, the distribution of documents over scores is modeled by a mixture of two distributions, one for the relevant and one for the nonrelevant documents, and expectation maximization (EM) is run to estimate the mixture parameters. A large volume of work has focused on selecting the appropriate form of the two distributions in the mixture. In this work we consider the form of the distributions as a given and we focus on the inference algorithm. We extend the EM algorithm (a) by simultaneously considering the ranked lists of documents returned by multiple retrieval systems, and (b) by encoding in the algorithm the constraint that the same document retrieved by multiple systems should have the same, global, probability of relevance. We test the new inference algorithm using TREC data and we demonstrate that it outperforms the regular EM algorithm. It is better calibrated in inferring the probability of document's relevance, and it is more effective when applied on the task of metasearch.

1 Introduction

Given a user's request, an information retrieval system assigns a score to each document in an underlying collection according to some model of relevance; then it returns the documents to the user in a decreasing order of the assigned scores. In reality, this ranked list of documents is a mixture of both relevant and nonrelevant documents. For a wide range of retrieval applications, including information filtering, topic detection, metasearch, distributed IR, *modeling* and *inferring* the distribution of relevant and nonrelevant documents over scores with a reasonable precision can be highly beneficial. For instance, in information filtering, topic detection, and recall-oriented retrieval, inferring the distributions of relevant and nonrelevant documents can be utilized to find the appropriate threshold over the

* We gratefully acknowledge the support provided by the NSF grant IIS-1017903 and by the European Commission grant FP7-PEOPLE- 2009-IIF-254562.

ranked list, below which the chance of encountering a nonrelevant document is larger than the chance of encountering a relevant one [2, 17]. In distributed IR and metasearch, score distributions can be used to normalize document scores, through Bayes' Law, and to combine different collections and/or the outputs of several search engines [1, 13].

Under the assumption of binary relevance, numerous combinations of statistical distributions have been proposed in the literature to model the score distributions of relevant and nonrelevant documents, including two Gaussians of equal variance [15], two Gaussians of unequal variance [16], two Poissons [9], and two Gamma distributions [7]. The most popular model has been a mixture of a Gaussian distribution for relevant documents and an Exponential distribution for nonrelevant documents [13], and this model has been widely used in a number of applications [13, 17, 1, 2]. More recently proposed models include a truncated version of the Gaussian-exponential mixture [2] and a Gamma distribution for nonrelevant documents combined with a mixture of Gaussians for relevant documents [11, 12, 10].

Having selected the form of the distributions, the *expectation maximization* (EM) algorithm [8] is typically run to infer the parameters of the mixture in the absence of any relevance judgments. This is a hard task but relative success has been reported in the past [4, 2, 1, 13]. However, it has been noted that EM suffers from treating all data equally, being very sensitive to initialization, and converging to a local optimum instead of the global one [1–3].

In this paper we consider the form of the distributions as a given and focus on the inference process. Our work is based on the observation that in the literature the EM algorithm has been applied in an inefficient way ignoring some of the constraints that often can be naturally imposed on the inference process. In particular, typically, EM estimates the model parameters on a per-query and per-system basis; iteratively it infers the probability of relevance of a document given its score using the current set of parameters, and then uses this inferred probability to update these parameters. However, often, multiple systems are available. This deployment of EM fails to account for the fact that for a given query, an individual document may be retrieved by multiple systems, in which case it should have a single, global, probability of relevance, consistent across systems. Aslam and Yilmaz [5] have shown that inferring document relevance from multiple retrieved lists can greatly benefit from this constraint.

Thus, we propose a novel framework for inferring score distributions and the document's probability of relevance, by (a) simultaneously considering ranked lists returned by multiple systems, and (b) encoding the aforementioned constraint in the EM algorithm. To measure the performance of this extended EM method, and to gauge its improvement over current approaches, we test it on TREC data. We show that the model parameters derived from the extended EM are more accurate than those derived from the regular EM: they can be used to better estimate precision-recall curves and average precision values, while they can lead to better results in metasearch.

2 Extended Expectation Maximization

The focus of this work is the inference process itself rather than the selection of probability density function (PDF) to be used in modeling the distribution of documents over scores. As a show case we select the widely used Gaussian-Exponential mixture to model score distributions and present a novel way to estimate the score distribution mixture parameters by inferring the probability of relevance for each document from multiple retrieval systems. The following table summarises the notation used.

Notation	**Description**
x	The score for a retrieved document computed by a system
d	The document retrieved by a system
θ	The parameters for the score distribution mixture model
$r = \{rel, nrel\}$	The hidden variable indicating whether a document is relevant
ϕ	PDF for the distribution of relevant scores
ψ	PDF for the distribution of nonrelevant scores

Probability of Relevance. The extension of the EM algorithm proposed in this work is based on the assumption that the probability of relevance inferred by the score distributions of relevant and nonrelevant documents produced by a system should precisely estimate the actual probability of document relevance, which is independent of the system, and is an intrinsic quality of the document itself. In what follows we give a more formal description of this assumption.

Let us assume that a system conflates different documents over approximately the same score x, and let *relevance conflation rate* be the proportion of relevant documents conflated over this score. This expresses the probability that a document with a certain score is relevant. For a retrieved document d with score x,

$$P(r = rel|x) = P(d \text{ is relevant}|x \approx x') = \frac{|\{d|x \approx x'; d \text{ is relevant}\}|}{|\{d|x \approx x'\}|}$$

The mean probability of relevance across all documents is equal to the generality G of the collection, i.e. the proportion of relevant documents in the collection, which is indeed independent of the scoring function of any IR system:

$$G = \frac{\#\text{relevant docs}}{\#\text{docs}}$$

For a given query, one can re-write G by iterating over all scores that a retrieval system would assign to all documents in the collection, or all documents in the collection as follows:

$$\begin{aligned} G &= \frac{1}{\#\text{ docs}} \sum_d \frac{|\{d|x \approx x'; d \text{ is relevant}\}|}{|\{d|x \approx x'\}|} \\ &= \frac{1}{\#\text{ docs}} \sum_d |\{d|d \text{ is relevant}\}| = \frac{\#\text{relevant docs}}{\#\text{docs}} \end{aligned}$$

We call an IR system *consistent* if the *relevance conflation rate* reflects the true probability of relevance as perceived by a user. This notion of consistency implies that all relevant documents conflated around the same score have on average the same value to a users, or in other words that $P(r|x)$ estimates an intrinsic quality of the relevance for the document d, $P(r|d)$, regardless of the IR system.

Score Distribution. A score distribution mixture model can be written as the linear combination of two score distributions: $\pi\phi(x) + (1-\pi)\psi(x)$, where π is the mixing coefficient. For a given search engine s, the proportion of relevant documents with score x is equal to,

$$P_s(r = rel|x) = \frac{P_s(r = rel)P_s(x|r = rel)}{P_s(r = rel)P_s(x|r = rel) + P_s(r = nrel)P_s(x|r = nrel)} \tag{1}$$

$$= \frac{\pi\phi(x)}{\pi\phi(x) + (1-\pi)\psi(x)} \tag{2}$$

If the model fits the score output of a search engine, the above equation accurately estimates the true, global, probability of document relevance $P(r|d)$. This is essentially the consequence of applying Bayesian law to score distributions of relevant and nonrelevant documents [13].

Treating $P_s()$ as an estimator, we can compute its bias:

$$E_x[P_s(r|x)] - \text{mean}[P(r|d)] = \int P_s(r|x)P_s(x)dx - G = \pi - G$$

This bias is very much dependent on the initial choice of the mixture coefficient π for each IR system modeled with score distributions. It is also determined by the ability of the fitting algorithm (EM in our case) to recover correct π from bad initial values, in subsequent iterations. We briefly discuss later how our extended EM algorithm helps on this.

EM and the multi-system extended EM

Estimating the parameters of the mixture in the absence of relevance judgments is an extremely hard task. The difficulty arises from the lack of knowledge about the hidden variable r, which determines the relevance of a document. The probability of a document d being relevant, $P(r|d)$, can be estimated through score distributions using Equation 2. Regular expectation maximization algorithm infers this probability through the posterior $P(r|x, \theta)$, and uses it to maximize the conditional expectation $\mathbb{E}_{r|x,\theta}\{\log P(x, r|\theta)\}$ [8]. θ is the model parameter, and x is the score for document d. The whole process can be summarized as follows:

1. Initialize the parameters for the mixture model of score distributions
2. **E step:** estimate $P(r|x, \theta)$ given the current model parameters
3. **M step:** update parameters to maximize $\mathbb{E}_{r|x,\theta}\{\log P(x, r|\theta)\}$; in our case they are the mixing coefficient π, the Gaussian parameters μ and σ, and the Exponential parameter λ.

4. Repeat steps 2 and 3 until the conditional expectation of the log-likelihood converges

Observing the above optimization iterations, the conflation rate for a document is estimated purely based on parameters for a single query-system run. Inspired by the work in [5], a better estimate can be obtained by combining information from other ranked lists, in which that document also appears. The central idea to our new approach is that a specific document has a global probability of relevance $P(r|d)$ independent of the retrieval system and the score it has been assigned. For instance, in Figure 1, document d is retrieved by both systems A and B with scores $x_{d,A}$ and $x_{d,B}$ respectively. The probability of relevance of d inferred from the score distribution model for system A or B via Equation 2 should be same and equal to $P(r|d)$ if document scores computed by scoring functions can represent the probability of relevance correctly.

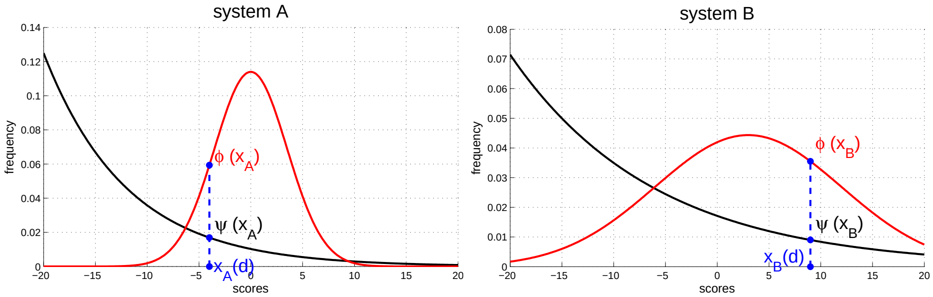

Fig. 1. A document d with scores from two different systems and the conflation rate in each system. ϕ and ψ shown are the (Gaussian, Negative Exponential) model.

In the EM algorithm's E step, for each system s, $P_s(r|x, \theta)$ is expected to infer the true relevance probability $P(r|d)$. Since $P(r|d)$ is de facto the intrinsic probability of relevance and should be independent of the system s, and the qualities of systems may vary, a better estimate can be obtained from many systems by taking the average of probabilities of relevance calculated from the model for each system s:

$$\hat{P}(r|d) = \frac{1}{\#\text{ systems}} \sum_s P_s(r|x_s, \theta_s) \tag{3}$$

The averaged estimator reduces both the bias and the variance and in this way it helps the EM algorithm.

- **Bias**: the bias $\pi_s - G$ for system s depends on the current EM-iteration parameter π_s. Averaging, the estimator's bias becomes $\frac{1}{\#systems} \sum_s \pi_s - G$, which on average is a smaller absolute bias, unless all π_s are unusually high or low.
- **Variance**: averaging random variables always decreases the variance, up to a linear factor in the independent case.

- **Convergence**: if systems are not consistent (in practice they are not), averaging helps by making EM converging faster: averaged probabilities of relevance need fewer iterations to become stable
- **Parameter estimates** of EM output: if the score distribution model chosen does not always fit the data (and in practice it does not [10]), averaged probabilities help identify relevant documents better than the per-system probabilities do.

After estimating the probability of relevance for each document, we update each parameter by setting the derivative of the conditional expectation of the log-likelihood $\mathbb{E}_{r|x,\theta}\{\log P(x, r|\theta)\}$ with respect to each parameter to zero. For every document d with score x_d retrieved by multiple systems, the updating equations become,

$$\mu = \frac{\sum\limits_{d} \hat{P}(r|d) x_d}{\sum\limits_{d} \hat{P}(r|d)}; \; \sigma^2 = \frac{\sum\limits_{d} \hat{P}(r|d)(x_d - \mu)^2}{\sum\limits_{d} \hat{P}(r|d)}; \; \lambda = \frac{\sum\limits_{d} (1 - \hat{P}(r|d))}{\sum\limits_{d} (1 - \hat{P}(r|d)) x_d} \tag{4}$$

One can observe that if relevance judgements were available, $\hat{P}(r|d)$ equals 1 when document d is relevant, and 0 otherwise. Then μ and σ are the mean and standard deviation of scores of relevant documents, and λ is the inverse of the mean of nonrelevant documents' scores. The extended expectation maximization algorithm can be seen below.

Algorithm 1 Extended Expectation Maximization Algorithm

```
Require: a list of retrieved document scores $\mathbf{x}_s$ for all $s \in S$
Initialize the parameters of the mixture $\theta_s$ for each system $s$
while algorithm has not converged do
    for all system $s$ in $S$ do
        Compute the posterior $P(r_d|\mathbf{x}_s, \theta_s)$ for each document $d$ with score $x_s$
    end for
    for all system $s$ in $S$ do
        for all document $d$ in $\mathbf{x}_s$ do
            Estimate the probability of relevance $\hat{P}(r|d)$ according to Equation 3
        end for
        Update model parameters for system $s$ through Equation 4
    end for
end while
```

3 Inferring Score Distributions using Extended EM

We first describe the experiment setup and three inference methods used to estimate the parameters of the score distributions and the documents' probability of relevance. Then, we describe a series of experiments and show that the proposed extension on EM significantly outperforms the regular EM in terms of the precision that one can infer system precision-recall curves and average precision. Finally, we use the inferred probabilities of relevance in the task of metasearch and demonstrate that the extended EM can achieve good performance.

3.1 Experiments

As mentioned earlier, in this work we use a mixture of Gaussian and Exponential density functions to model the score distributions of documents, regardless of some noticeable theoretical and practical problems this model has [14, 11, 3], since our primary goal is the parameter estimation method. Besides, our proposed algorithm can be easily adopted to other score distribution models. To evaluate our methodology we use TREC data and infer the score distributions for automatic search engines run over different queries. For each system-query run the scores of the retrieved documents are first normalized into a 0 to 1 range. The model parameters of the score distributions are then estimated through the following three approaches:

- *judSD*: TREC judgments are used to estimate the model parameters separately for relevant and non-relevant documents.
- *regEM*: the regular EM algorithm is used to estimate model parameters in absence of relevance judgments.
- *extEM*: the proposed extended EM algorithm is used to estimate model parameters in absence of relevance judgments.

Under the assumption that the score distribution can fit the data well, one cannot hope for a more accurate estimation of the model parameters than the one obtained when using relevance judgments. Hence, in our experiments, *judSD* is considered the gold standard. Results obtained by the other two approaches are compared with the ones obtained by *judSD*. Given that relevance judgments are available we could compare all three approaches with the actual gold standard but this way any results would conflate the effects not only of the inference process but also of the model's inherent goodness of fit. Using *judSD* as a gold standard eliminates effect of the imperfect score distribution model.

3.2 Precision-recall curve

Precision recall (PR) curves can be easily inferred through score distributions [14]. Let $\varPhi(x) = \int_x^1 \phi(x)dx$ and $\varPsi(x) = \int_x^1 \psi(x)dx$ be the cumulative density functions *from the right* for the relevant and nonrelevant documents respectively. We use integrals up to 1 because our scores are normalized into a range from 0 to 1. For each recall level r we can estimate the score at which the retrieval system achieves recall equal to r by the inverse of relevant document cumulative density function: $\text{score}(r) = \varPhi^{-1}(r)$. Then, $n(r) = \varPsi(\text{score}(r))$ is the percentage of non-relevant documents found up to recall r in the ranked list. Hence, the precision at recall r can be computed similarly as in [14],

$$\text{prec}(r) = \frac{r}{r + n(r) * N}$$

where N is the number of nonrelevant documents retrieved. Computing precision at all recall levels from the score distribution models ϕ and ψ gives an estimated PR curve.

In the first experiment, we infer the precision recall curve based on the three inference approaches: regEM, extEM, and judSD for all 116 automatic systems submitted to TREC 8 ad-hoc track. We report the mean and standard deviation of the root mean square errors (RMSError) and absolute errors (ABSError) between the predicted precisions at all recall levels using judSD and the one using regEM or extEM.

The results are summarized in the table below. The table shows that extEM produces significantly better PR estimates than regEM, both in terms of mean and standard deviation of the errors. Furthermore, among all 5800 runs (116 systems and 50 queries), there are 5164 (89.0%) runs for which extEM is better than regEM in terms of RMSError with an average absolute (relative) improvement of 0.27 (65.2%), and 5139 (88.6%) runs with an average absolute (relative) improvement of 0.25 (66.9%) in terms of ABSError.

	RMSError		ABSError	
	mean	stdev	mean	stdev
regEM	0.374	0.237	0.325	0.234
extEM	**0.142**	**0.131**	**0.112**	**0.106**

Since extEM can utilize multiple systems to more accurately infer parameters for score distributions, we also report the RMSErrors and ABSErrors over different number of systems. Figure 2 shows the average RMSErrors and ABSErrors of predicted PR curve by regEM and extEM comparing to judSD over different numbers of TREC 8 systems. Systems are ordered by their mean average precision reported by TREC. The n systems on the plot represent best n systems based on TREC evaluation. In all cases extEM outperforms regEM.

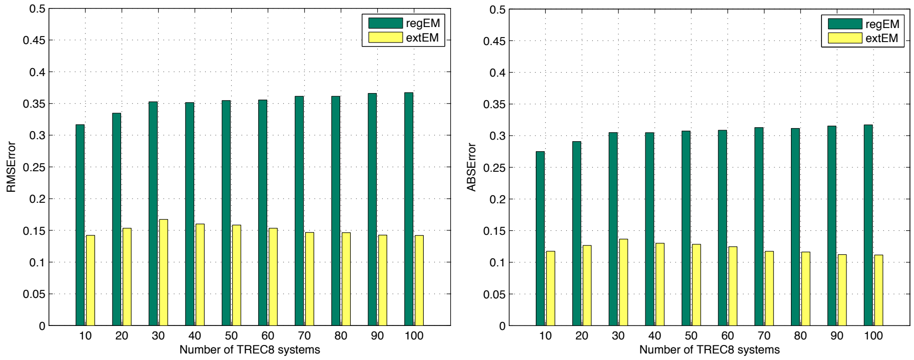

Fig. 2. The average RMSError and ABSError of inferred PR curve through regEM and extEM for different numbers of TREC8 systems.

3.3 Expected average precision

Expected average precision (EAP) is a probabilistic version of average precision, and can be computed as [6]:

$$\mathbb{E}\{AP\} = \frac{1}{R}\sum_{i=1}^{N}(\frac{p_i}{i}(1+\sum_{j=1}^{i-1}p_j)) \tag{5}$$

where N is the number of retrieved documents, and p_i is the probability of the i^{th} document in the rank list being relevant. This probability can be directly inferred through the estimated mixture of score distributions by Equation 2, where

$$p_i = \frac{\pi\phi(x_i)}{\pi\phi(x_i) + (1-\pi)\psi(x_i)}$$

x_i is the score for the document at rank i. When relevance information is available, this probability is either 0 or 1, and EAP reduces to AP. R is the number of relevant documents. We compute R as $\sum_{i=1}^{N} p_i$. Here R may be underestimated comparing with the R reported by TREC, since we only estimate the number of relevant documents from a single retrieved the list. Hence, EAP computed by our approach is often overestimated comparing to AP evaluated by TREC.

We use different number of systems submitted to TREC to infer the model parameters. This time we extend our experiment data to include automatic systems submitted to TREC 6, 7 and 8 ad-hoc tracks, TREC 9 and 10 Web tracks (ad-hoc tasks) and TREC 12 Robust track. The topics used are the TREC topics 301-550 and 601-650. The Robust track topic set in TREC 12 consists of two subsets of topics, the topics 601-650 and 50 old topics selected based on topic hardness from past collections.

We first compute EAP for those systems using judSD, regEM, and extEM, then average EAPs over different queries to get the probabilistic version of mean average precision, mean EAP. Figure 3 shows the mean EAP estimated from regEM or extEM for all automatic systems submitted to different TRECs and its correlation with the one estimated from judSD. A blue square dot indicates mean EAP estimated by regEM averaged over different queries, and a red star dot indicates mean EAP estimated by extEM. As we can see, red star dots are mostly clustered along the diagonal line, showing that numbers predicted by extEM are clearly more correlated with ones predicted by judSD.

3.4 Application to metasearch

Score distributions are often used for IR tasks such as information fusion and metasearch [1, 13]. With score distributions, the probability of a document relevance given the score can be estimated by Equation 2. Documents retrieved by different systems can be simply merged by their estimated probabilities of document relevance. If a document appears in multiple systems, this probability can be computed as the average of different estimations from different systems as shown in 3.

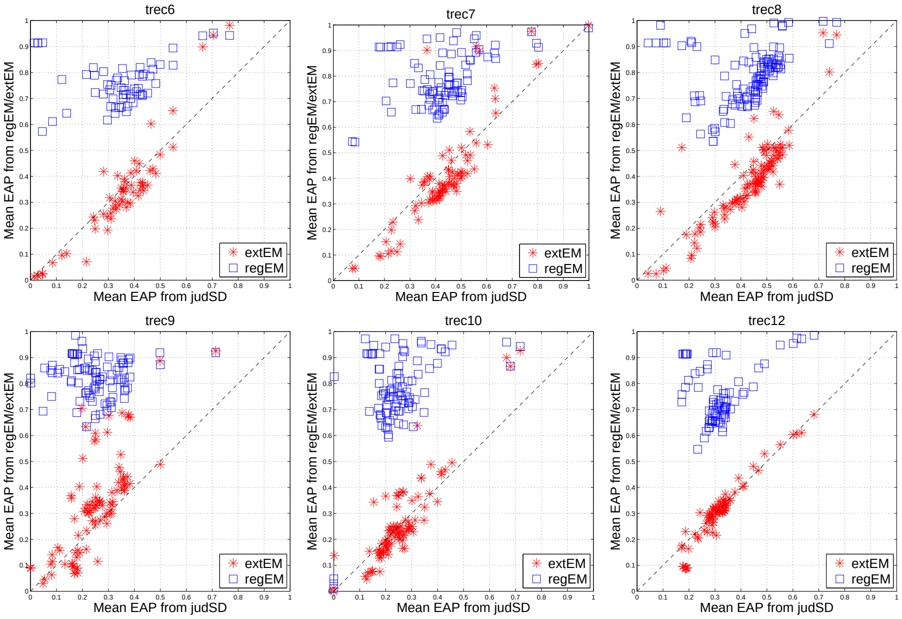

Fig. 3. The scatter plot of mean EAP estimated by regEM(blue square dots)/extEM(red star dots) comparing with the one estimated by judSD for all automatic systems submitted to TREC6, 7, 8, 9, 10, 12.

To test the practical utility of our methodology, we also examine how well it can improve the performance of using inferred score distributions for the task of metasearch. The testbed we use, again, is still all automatic search engines submitted to TREC 6, 7, 8, 9, 10, 12. We randomly select 10, 20, 30, 40, and 50 systems for our experiments, and merge results from those systems using probabilities of relevance for each document estimated by judSD, regEM, and extEM. The whole process is repeated for 20 times for different system numbers, and we report the mean average precision averaged over these 20 repetitions. The same query sets are used for different TRECs as in the previous experiments. The results are also compared with the most popular metasearch algorithm, combMNZ, which does not rely on score distributions.

Table 1 shows that judSD outperforms all other methods in metasearch. This is not surprising since judSD uses the relevance information. Score distributions estimated by extEM performs almost consistently better than ones estimated by regEM regardless of the number of systems. For a few TREC7 and TREC10 systems, extEM performs worse than regEM, which may be caused by some bad-quality systems that undermine the inference process. However, metasearch based on extEM does not do better than combMNZ. This is possible due to the imperfect choice of score distribution model. We only see extEM beats combMNZ on TREC6 collections. With a better model, the metasearch quality based on extEM is expected to be improved. We leave this as the future work.

# of rand systems		10	20	30	40	50
TREC6	judSD	0.3569	0.3859	0.3936	0.4049	0.4071
	extEM	0.2709	0.2998	0.2988	0.3084	0.3102
	regEM	0.2139	0.2623	0.2695	0.2819	0.2852
	combMNZ	0.2474	0.2808	0.2812	0.2886	0.2882
TREC7	judSD	0.3107	0.3310	0.3376	0.3394	0.3452
	extEM	0.2462	0.2652	0.2679	0.2669	0.2773
	regEM	0.2139	0.2623	0.2695	0.2819	0.2852
	combMNZ	0.2584	0.2742	0.2732	0.2738	0.2809
TREC8	judSD	0.3496	0.3614	0.3700	0.3731	0.3730
	extEM	0.2914	0.3061	0.3111	0.3184	0.3162
	regEM	0.2641	0.2873	0.2965	0.3045	0.3036
	combMNZ	0.3055	0.3145	0.3176	0.3224	0.3192
TREC9	judSD	0.2913	0.3114	0.3234	0.3328	0.3347
	extEM	0.2042	0.2238	0.2369	0.2430	0.2466
	regEM	0.1951	0.2157	0.2298	0.2340	0.2415
	combMNZ	0.2389	0.2525	0.2614	0.2675	0.2701
TREC10	judSD	0.3072	0.3332	0.3445	0.3435	0.3527
	extEM	0.2050	0.2123	0.2155	0.2150	0.2214
	regEM	0.1894	0.2102	0.2179	0.2226	0.2288
	combMNZ	0.2458	0.2608	0.2627	0.2608	0.2683
TREC12	judSD	0.3072	0.3332	0.3445	0.3435	0.3527
	extEM	0.2544	0.2756	0.2826	0.2884	0.2892
	regEM	0.2139	0.2623	0.2695	0.2819	0.2852
	combMNZ	0.2852	0.2945	0.2977	0.3027	0.3019

Table 1. Mean average precision achieved by meta-search using judSD, extEM, regEM, combMNZ based on randomly selected 10, 20, 30, 40, 50 systems submitted to TREC 6, 7, 8, 9, 10, 12 averaged over 20 times

4 Conclusions and Future Work

In this paper we propose a novel approach to infer the probability of document relevance through multiple sets of score distributions for different systems. We extend EM algorithm by imposing the constraint that the document appearing in multiple ranked lists returned by different systems should have the same probability of the relevance. In the experiment, the score distributions estimated by new proposed extend EM clearly outperforms the one estimated by regular EM in terms of inferring precision-recall curves and estimating expected average precisions. We also demonstrate the use of these improved probabilities on the task of metasearch.

In future, Equation 3 can be extended to use a weighed average to better estimate the probability of document relevance. Weights can be determined by the quality of the system or the rank of that document in a system. Instead of the simple average, a more sophisticated way to combine multiple estimations from different systems should be investigated. Furthermore, extended EM can also be applied to estimate parameters for those recently proposed state-of-art score distribution models [2, 11].

References

1. Avi Arampatzis and Jaap Kamps. A signal-to-noise approach to score normalization. In *Proceeding of the 18th CIKM*, CIKM '09, pages 797–806, New York, NY, USA, 2009. ACM.
2. Avi Arampatzis, Jaap Kamps, and Stephen Robertson. Where to stop reading a ranked list?: threshold optimization using truncated score distributions. In *Proceedings 32nd ACM SIGIR*, pages 524–531, New York, NY, USA, 2009. ACM.
3. Avi Arampatzis and Stephen Robertson. Modeling score distributions in information retrieval. *Information Retrieval*, 14:26–46, 2011.
4. Avi Arampatzis and André van Hameren. The score-distributional threshold optimization for adaptive binary classification tasks. In *Proceedings of the 24th ACM SIGIR*, pages 285–293, New York, NY, USA, 2001. ACM.
5. Javed A. Aslam and Emine Yilmaz. Inferring document relevance from incomplete information. In *Proceedings of the 6th CIKM*, pages 633–642. ACM Press, November 2007.
6. Javed A. Aslam, Emine Yilmaz, and Virgiliu Pavlu. The maximum entropy method for analyzing retrieval measures. In *Proceedings of the 28th ACM SIGIR*, pages 27–34. ACM Press, August 2005.
7. Christoph Baumgarten. A probabilistic solution to the selection and fusion problem in distributed information retrieval. In *Proceedings of the 22nd ACM*, pages 246–253, New York, NY, USA, 1999. ACM.
8. Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006.
9. Abraham Bookstein. When the most "pertinent" document should not be retrieved—an analysis of the swets model. *Information Processing & Management*, 13(6):377–383, 1977.
10. Keshi Dai, Evangelos Kanoulas, Virgil Pavlu, and Javed A. Aslam. Variational bayes for modeling score distributions. *Information Retrieval*, 14(1):47–67, 2011.
11. E. Kanoulas, V. Pavlu, K. Dai, and J.A. Aslam. Modeling the Score Distributions of Relevant and Non-relevant Documents. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, page 163. Springer, 2009.
12. Evangelos Kanoulas, Keshi Dai, Virgil Pavlu, and Javed A. Aslam. Score distribution models: assumptions, intuition, and robustness to score manipulation. In *SIGIR '10: Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 242–249, New York, NY, USA, 2010. ACM. (http://portal.acm.org/citation.cfm?doid=1835449.1835491).
13. R. Manmatha, T. Rath, and F. Feng. Modeling score distributions for combining the outputs of search engines. In *Proceedings of the 24th SIGIR*, pages 267–275, New York, NY, USA, 2001. ACM.
14. Stephen Robertson. On score distributions and relevance. In *Advances in Information Retrieval, ECIR 2007*, volume 4425/2007, pages 40–51. Springer, June 2007.
15. John A. Swets. Information retrieval systems. *Science*, 141(3577):245–250, July 1963.
16. John A. Swets. Effectiveness of information retrieval methods. *American Documentation*, 20:72–89, 1969.
17. Yi Zhang and Jamie Callan. Maximum likelihood estimation for filtering thresholds. In *Proceedings of the 24th ACM SIGIR*, pages 294–302, New York, NY, USA, 2001. ACM.