A Deep Learning Approach for Selective Relevance Feedback

Suchana Datta[1][0000−0001−9220−6652], Debasis Ganguly[2][0000−0003−0050−7138], Sean MacAvaney[2][0000−0002−8914−2659], and Derek Greene[1][0000−0001−8065−5418]

[1] University College Dublin, Ireland
[2] University of Glasgow, United Kingdom
suchana.datta@ucdconnect.ie, debasis.ganguly@glasgow.ac.uk, sean.macavaney@glasgow.ac.uk, derek.greene@ucd.ie

Abstract. Pseudo-relevance feedback (PRF) can enhance average retrieval effectiveness over a sufficiently large number of queries. However, PRF often introduces a drift into the original information need, thus hurting the retrieval effectiveness of several queries. While a selective application of PRF can potentially alleviate this issue, previous approaches have largely relied on unsupervised or feature-based learning to determine whether a query should be expanded. In contrast, we revisit the problem of selective PRF from a deep learning perspective, presenting a model that is entirely data-driven and trained in an end-to-end manner. The proposed model leverages a transformer-based bi-encoder architecture. Additionally, to further improve retrieval effectiveness with this selective PRF approach, we make use of the model's confidence estimates to combine the information from the original and expanded queries. In our experiments, we apply this selective feedback on a number of different combinations of ranking and feedback models, and show that our proposed approach consistently improves retrieval effectiveness for both sparse and dense ranking models, with the feedback models being either sparse, dense or generative.

1 Introduction

The keywords that a user enters as query to a search engine are often insufficient to express the user's information need, resulting in a *lexical gap* between the text in the query and the relevant documents [2]. Standard pseudo-relevance feedback (PRF) methods, such as the relevance model [19] and its variants [14,36,35,26], can overcome this problem and ultimately yield improvements in retrieval effectiveness. Generally speaking, PRF methods are designed to enrich a user's initial query with distinctive terms from the top-ranked documents [34,27,43]. Despite the demonstrated success of PRF in improving retrieval effectiveness, a number of studies have identified certain limitations of this strategy [3,25,9,13]. For the most part, these limitations share a common theme: there is no consistent PRF setting that works well across a wide range of queries; to put in simple words, *one size does not fit all*. Figure 1 illustrates such a situation, where nearly 38.9% of

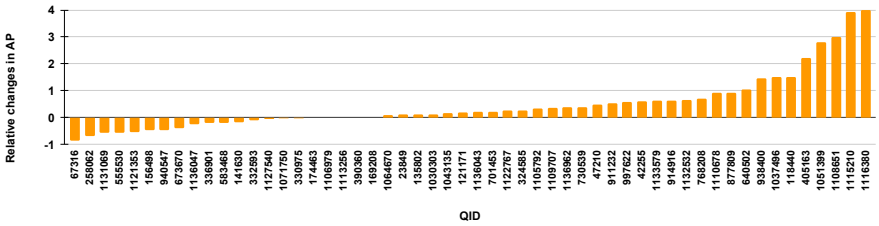

Fig. 1: Relative changes in AP, i.e., (AP(post-fdbk) - AP(pre-fdbk))/AP(pre-fdbk), for TREC DL'20 queries. We observe that many queries are negatively impacted by PRF (bars below the x-axis).

queries from TREC DL'20 topic set are penalized as a result of PRF. It has been shown that not all documents contribute equally well to PRF, as certain documents may impair retrieval effectiveness when used to expand a query [20,1]. This can even be true when relevant documents are used to enrich a query's representation [39]. It has also been observed that some queries are amenable to more aggressive query expansion, while others work better with more conservative settings [32]. Moreover, not all terms might contribute equally well in terms of enriching the representation of a query [4,16], which suggests that a selective approach to PRF can potentially improve the overall IR effectiveness.

Rather than following the previous approaches on adapting the number of feedback terms [32] or attempting to choose a robust subset of documents for PRF [20,1], we rather focus on solving the more fundamental decision question of "*whether or not to apply PRF for a given query*" [9,25] with the help of a supervised data-driven approach. We hypothesise that selectively applying feedback to only those queries that are amenable to PRF can improve the overall retrieval effectiveness by avoiding query drift in cases where feedback would not be beneficial. Our idea is depicted in Figure 2.

The main novelty of our proposed selective pseudo relevance feedback (SRF) approach is that in contrast to existing work on selective PRF, we propose a data-driven supervised neural model for predicting which queries are conducive to PRF. More specifically, during the training phase we make use of the relevance assessments to learn a decision function that, given the query and the top-retrieved set of documents both with and without feedback, predicts whether it is useful to apply PRF. During the inference phase, we make use of only a part of the shared parameter network which, given a query and its top-retrieved document set, predicts whether PRF is to be applied (schematically illustrated in Figure 2). This way of inference reduces computational costs for queries where PRF should eventually be ignored.

A key advantage of our SRF approach is that it can be applied to the output ranked list obtained by **any retrieval model**, ranging from sparse models (e.g., BM25, LM-Dir etc.) to dense ones (e.g., MonoBERT [31]). Moreover, in the SRF workflow it also is possible to use **any PRF model** to enrich a query's

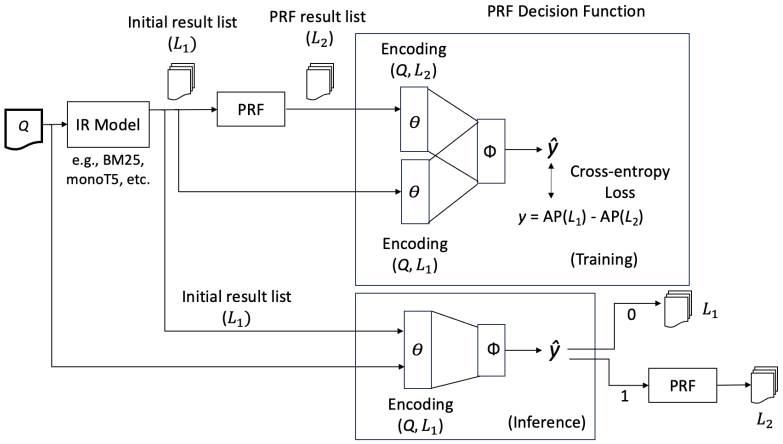

Fig. 2: A schematic diagram of selective feedback. The main contribution of this paper is a supervised data-driven approach towards realising the decision function.

representation, ranging from sparse models (e.g., RLM [19]) to dense ones (e.g., ColBERT-PRF [41]) to even generative ones (e.g., GRF [26]).

2 Related Work

The evolution of relevance feedback in IR spans from traditional query expansion models [32,4] to cluster-based feedback document selection [20,16]. While prior research has considered both unsupervised selective feedback [9] and feature-driven methods [25], we introduce a data-driven neural strategy for selective relevance feedback. Several existing methods, both supervised and unsupervised, hinge on *decision-based relevance feedback*. One unsupervised approach uses Query Performance Prediction (QPP) scores [38,47,10,37,15], which we include as a baseline. The higher the QPP score, the greater the chance of identifying relevant documents at the top rank positions with the initial query. However, high variances in retrieval status values, as seen in neural re-rankers like MonoBERT [31], can make QPP scores deceptive. To avoid such heuristics, our method focuses solely on query terms and the documents retrieved by that query in order to learn the selection function.

PRF on and for Dense Retrieval. Recently, the community has seen a significant interest in feedback for dense retrieval to boost performance. Precursors to dense feedback models made use of word embeddings for PRF, e.g., KDERLM [35] which proposed a generalised RLM with word embeddings, and PRF-NMF [45], which leveraged matrix factorisation to bridge the semantic gap between terms from a query and its top-retrieved documents.

The study by [44] explored relevance feedback principles within dense retrieval models. Li et al. [22] analyzed feedback signal quality, comparing tradi-

tional models like Rocchio [34] with dense retrievers like ANCE-based retrievers [42], finding the latter more resilient. Representation models, such as ColBERT [18], can allow us to append additional embedding layers to the query representation, as demonstrated by [40]. This method employed contextualized PRF to cluster and rank feedback document embeddings in order to select suitable expansion embeddings, thus improving document ranking. In other work, [48] leveraged implicit feedback from historical clicks for relevance feedback in dense retrieval. The authors introduced counterfactual-based learning-to-rank, showing that historic clicks can be highly informative in terms of relevance feedback. Lastly, [23] proposed the idea of fusing feedback signals from both sparse and dense retrievers in the context of PRF.

More recently, PRF on dense IR models has garnered significant interest [21,29,46,41]. The concept of 'dense for PRF' was first emphasized in [28], which proposed a reinforcement-based learning algorithm designed to explore and exploit various retrieval metrics, aiming to learn an optimized PRF function. With the recent success of LLMs, [26] proposed a generative feedback method (GRF) that makes use of LLM generated long-form texts instead of first pass retrieved results to build a probabilistic feedback model. In contrast, our work aims to develop a generic PRF strategy that does not apply feedback blindly, but rather learns a selection function in a supervised manner to analyze the suitability of relevance feedback for each query irrespective of sparse or generative PRF.

Selective PRF. Prior work on selective PRF has considered either fully unsupervised approaches [9] or feature-based supervised approaches [25] for selective relevance feedback (SRF). The former makes use of query performance prediction (QPP) based measures to predict if a query should be expanded, where the decision depends on whether the QPP score exceeds a given threshold. On the other hand, existing supervised approaches first represent each query as a bag of characteristic features derived from its top-retrieved set of documents. A classifier is then trained on these features to predict whether or not a query should be expanded [25].

3 Model Description

3.1 A Generic Decision Framework for PRF

Given a set of queries $\mathcal{Q} = \{Q_1, \ldots, Q_n\}$, a standard relevance feedback model M uses the information from the top-retrieved documents of each query to enrich its representation, i.e., $M : Q \mapsto \phi_M(Q)$. Consequently, each query $Q \in \mathcal{Q}$ is transformed to an enriched representation $\phi_M(Q)$, which is then used either for re-ranking the initial list, or to execute a second-step retrieval.

Unlike the standard PRF setting, a decision-based selective PRF framework first applies a *decision function*, $\theta : Q \mapsto \{0, 1\}$, which outputs a Boolean to indicate if the retrieval results for Q is likely to be improved after application of PRF. As per our proposal, the overall PRF process on the set of queries $\mathcal{Q}$ does not simply replace each query Q with its enriched form $\phi_M(Q)$. Rather, it

makes use of the function $\theta(Q)$ for each query Q to decide whether to output the initial ranked list or to make use of the enriched query representation $\phi_M(Q)$, as obtained by a PRF model M (leading to either re-ranking the initial list or re-retrieving a new list via a second stage retrieval). The top-k ranked list of documents, $L_k(Q) = \{D_1^Q, \ldots, D_k^Q\}$, retrieved for a query Q, in addition to being a function of the query Q itself, is thus also a function of i) the feedback model M, ii) the enriched query representation $\phi_M(Q)$, and iii) the decision function θ, i.e.,

$$L_k(Q) = \begin{cases} \sigma(Q), & \text{if } \theta(Q) = 0 \\ \sigma(\phi_M(Q)), & \text{if } \theta(Q) = 1, \end{cases} \tag{1}$$

where $\sigma(Q)$ denotes a retrieval model, e.g., BM25, that outputs an ordered set of k documents sorted by the similarity scores.

Previous approaches have explored the use of both unsupervised and supervised approaches for addressing this decision problem. We now briefly explain both strategies in our own context.

Unsupervised decision function. An unsupervised approach, such as [9], applies a threshold parameter on a QPP estimator function, $\theta_{\text{QPP}} : Q \mapsto [0, 1]$. More concretely, if the predicted QPP score is lower than the threshold parameter, it is likely to indicate that the retrieval performance for the query has scopes for further improvement and subsequently PRF is applied for this query. Formally speaking, the decision function of an unsupervised approach takes the form

$$\theta(Q) \stackrel{\text{def}}{=\!=} \mathbb{I}(\theta_{\text{QPP}} < \tau), \tag{2}$$

where $\tau \in [0, 1]$ is the threshold parameter.

Supervised decision function. In the supervised approach, the decision function also depends on the enriched query representation and its top-retrieved documents. More precisely, a supervised PRF decision is a parameterized function of features of i) the query Q, ii) its top-retrieved documents $L_k(Q)$, iii) the enriched query $\phi_M(Q)$, and iv) its top-retrieved set $L_k(\phi_M(Q))$ [25]. The training process itself makes use of a set of queries $\mathcal{Q}_{\text{train}}$ for which ground-truth indicator labels are computed by evaluating the relative effectiveness obtained with the original query vs. the enriched query with the help of available relevance assessments. Formally,

$$y(Q) = \mathbb{I}(\text{AP}(\phi_M(Q)) > \text{AP}(Q)), \tag{3}$$

where $\text{AP}(Q)$ denotes the average precision of a query $Q \in \mathcal{Q}_{\text{train}}$. The indicator values of $y(Q)$ are used to learn the parameters of a classifier function to yield a supervised version of the decision function θ:

$$\theta(Q) \stackrel{\text{def}}{=\!=} \zeta \cdot \mathbf{z}_{Q,\phi_M(Q)}, \text{ where } \theta(Q) \approx \underset{\zeta}{\text{argmin}} \sum_{Q' \in \mathcal{Q}'_{\text{train}}} (y(Q') - \zeta \cdot \mathbf{z}_{Q',\phi_M(Q')})^2. \tag{4}$$

In Equation 4, ζ represents a set of learnable parameters, with $\mathbf{z}_{Q',\phi_M(Q')}$ denoting a set of features extracted from both the original query Q' and the enriched

query $\phi_M(Q')$ along with the features from their top-retrieved set of documents $L_k(Q')$ and $L_k(\phi_M(Q'))$. The variable $y(Q')$, as per the definition of $y(Q)$, denotes the ground-truth indicating if PRF should be applied for Q'. The optimal parameter vector ζ, as learned from a training set of queries $\mathcal{Q}_{\text{train}}$ (Equation 4) is then used to predict the decision for any new query Q. The features we use are described later in the paper in Section 4.2.

3.2 Transformer-based Encoding for PRF Decision

We now describe a data-driven approach for learning the decision function with deep neural networks. Instead of making use of a specific set of extracted features as used in the QPP model in [11], the learning objective of a transformer-based PRF model makes use of the terms present in the documents and the queries. As with Equation 4, we make use of both the content of the original query Q and its enriched form $\phi_M(Q)$, along with their top-retrieved sets. More formally,

$$\theta(Q) \stackrel{\text{def}}{=\!=} \zeta \cdot (\mathcal{E}(Q, D_1^Q, \ldots, D_k^Q) \oplus \mathcal{E}(\phi_M(Q), D_1^{\phi_M(Q)}, \ldots, D_k^{\phi_M(Q)})), \quad (5)$$

where $\theta(Q)$ is learned by computing

$$\underset{\zeta}{\text{argmin}} \sum_{Q' \in \mathcal{Q'}_{\text{train}}} (y(Q') - \zeta \cdot (\mathcal{E}(Q', L_k(Q')) \oplus \mathcal{E}(\phi_M(Q'), L_k(\phi_M(Q'))))^2. \quad (6)$$

In Equation 6, $\mathcal{E}$ is a parameterized function for encoding the interaction between a query Q and its top-retrieved documents, $L_k(Q)$. This encoding function, generally speaking, maps a query (a sequence of query terms) and a sequence of documents (which are themselves sequences of their constituent terms) into a fixed length vector, i.e., $\mathcal{E} : Q, L_k \mapsto \mathbb{R}^p$ (p an integer, e.g., for BERT embeddings $p = 768$). Here $\oplus$ indicates an interaction operation (e.g., a merge layer in a neural network) between the query-document encodings corresponding to the original query and the enriched one.

The transformer-based encoding uses the BERT architecture which takes as input the contextual embeddings of the terms for each pair comprising a query Q and its top-retrieved document $D_i^Q \in L_k(Q)$. The 768 dimensional '[CLS]' representations of each *query-document* pair is then encoded with LSTMs as a realisation of the encoded representation of a query and its top-retrieved set, i.e., to define $\mathcal{E}(Q, L_k(Q))$ as per the notation of Equation 6. We further obtain a BERT-based encoding of the expanded query $\phi_M(Q)$ and its top-retrieved set and then merge the two representations before passing them through a feed-forward network. More formally,

$$\mathcal{E}(\bar{Q}, L_k(\bar{Q})) = \text{LSTM}(\text{BERT}(\bar{Q}, D_1^{\bar{Q}})_{[\text{CLS}]}, \ldots, \text{BERT}(\bar{Q}, D_k^{\bar{Q}})_{[\text{CLS}]}). \quad (7)$$

The variable $\bar{Q} \in \{Q, \phi_M(Q)\}$, i.e., in one branch of the network it corresponds to the original query, whereas in the other it corresponds to the expanded one. Figure 3 shows the transformer-specific implementation of the encoding function. The set of learnable parameters ζ in this case comprises of the LSTM and the

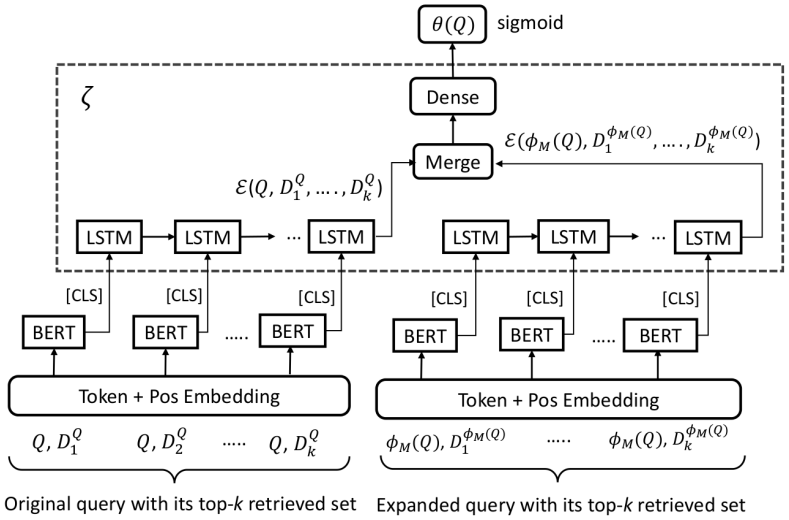

Fig. 3: Training of a transformer based query-document architecture with shared parameters for selective PRF. During inference, only the left part of the network is used to output whether to apply PRF or not for a given query.

fully connected (dense) layer parameters, as shown in Figure 3. We name this particular model **Deep-SRF-BERT** (Deep Selective Relevance Feedback with the use of BERT transformers).

Model Inference. During inference, only the part corresponding to the original query and its top-retrieved set of documents is used to predict the output variable (a sigmoid) which if higher than 0.5 indicates that PRF should be applied.

3.3 Model Confidence-based PRF Calibration

Prior work has applied confidences of prediction models to improve retrieval effectiveness [5]. In our work, we use the uncertainties in the prediction of the decision function to further improve search results. Rather than only reporting either results with or without relevance feedback, we make use of the confidence of the decision function $\theta(Q)$ to combine the results from the two lists – one without feedback and the other with feedback. Specifically, if the supervised model outlined in Section 3.1 is decisive in its choice between $L_k(Q)$ (original query retrieved list) and $L_k(\phi_M(Q))$ (the list retrieved with the expanded query), then one of the rankings is expected to dominate over the other. However, when the model $\theta(Q)$ itself is not confident about the prediction, we can potentially achieve better results if we 'meet somewhere in the middle'.

Formally, we propose a rank-fusion based method, where the fusion weights are obtained from the predictions of the PRF decision model $\theta(Q)$. The predicted value $\theta(Q)$ being a sigmoid represents the probability of classifying the decision

into one of the two outcomes - the closer $\theta(Q)$ is to 0, the higher is the model's confidence in not applying feedback, and similarly the closer $\theta(Q)$ is to 1, the higher is the model's confidence in applying PRF. The predicted value of $\theta(Q) \in [0, 1]$ can thus be used as weights to fuse the two different ranked lists, i.e., the fusion score $\sigma_F(Q, D)$ of a document D for a query Q is given by

$$\sigma_F(Q, D) = \frac{1 - \theta(Q)}{\text{Rank}(D, L_k(Q))} + \frac{\theta(Q)}{\text{Rank}(D, L_k(\phi_M(Q)))}, \tag{8}$$

where the notation $\text{Rank}(D, L)$ denotes the rank of a document D in a list L.

If $D \notin L$, then the rank is set to a large value $\aleph(> k)$, which in our experiments was set to 1000 (higher than all possible values of k we experimented with). For values of $\theta(Q)$ close to 0.5 (the most uncertainty in prediction), the fusion-based approach leads to a more uniform contribution from both the lists. In contrast, a value of $\theta(Q)$ close to 0 ensures that the majority of the score contribution comes from the original query (because $1 - \theta(Q) >> \theta(Q)$), and a similar argument applies for $\theta(Q) \to 1$, in which case the major contribution comes from the second term in the right-hand side of Equation 8.

4 Evaluation

4.1 Research Questions

Since a primary contribution of this paper is the idea of applying a fully data-driven approach, the first research question that we investigate is whether a shift from the existing feature-based approach for selective PRF to a data-driven one does indeed result in improved retrieval effectiveness. Therefore, we formulate our first research question as follows.

- RQ1: Does SRF lead to overall improvements in IR effectiveness over non-selective and other baseline approaches?

Our second research question aims to investigate whether the model prediction uncertainty-based fusion of the two ranked lists – one retrieved with the original query and the other with the expanded one, can potentially improve the retrieval effectiveness further.

- RQ2: Can we use the confidence estimates from our selective PRF model for a *soft selection* of information from both pre-feedback and post-feedback sources to further improve IR effectiveness?

In our third research question, the aim is to investigate how effectively the selection strategy in PRF *transfers* across different feedback approaches, i.e., training the SRF approach only once on a PRF model (e.g., RLM), and then apply the decision model on other PRF models (e.g., ColBERT-PRF).

- RQ3: Does selection effectively transfers the learning across different PRF approaches?

4.2 Methods Investigated

We consider a range of unsupervised and supervised methods, described below. Some baselines refer to existing methods, while others are extensions of alternative approaches to allow us to provide a fair comparison, such as by using a more recent QPP method instead of the originally-proposed clarity score [9].

PRF is a standard non-selective relevance feedback model, namely the relevance model (RLM) [19]. We use the RM3 version of the relevance model as reported in [17], which is a linear combination of the weights of the original query model and new expansion terms. In fact, we use RLM as one of the base PRF model M which means that the standard RLM degenerates to a specific case of the generic selective PRF framework of Equation 1 with $\theta(Q) = 1 \forall Q \in \mathcal{Q}$, i.e., when for each query we use its enriched form $\phi_M(Q)$.

R2F2 refers to an adaptation of the Reciprocal Rank-based Fusion (RRF) [6], a simple yet effective approach for combining the document rankings from multiple IR systems. For our task, instead of combining ranked lists from two different retrieval models, we merge the ranked lists of the original and the expanded queries, i.e., $L_k(Q)$ and $L_k(\phi_M(Q))$ as per our notations. We name the adapted method Reciprocal Rank Fusion-based Feedback (R2F2). Formally, the score for document D after fusion is given by

$$\sigma_F(Q, D) = \frac{1-\alpha}{\text{Rank}(D, L_k(Q))} + \frac{\alpha}{\text{Rank}(D, L_k(\phi_M(Q)))}, \tag{9}$$

where, similar to Equation 8 $\text{Rank}(D, L)$ denotes the rank of a document in a list L (this being a large number $\aleph$ if $D \notin L$), and $\alpha \in [0, 1]$ is a linear combination hyper-parameter that we adjust with grid search on each training fold. A lower value of α puts more emphasis on the initial retrieval list, whereas a higher value ensures that the feedback rank of a document contributes more. Equation 9 is a special case of Equation 8 with a constant value of $\theta(Q) = \alpha$ for each query Q.

QPP-SRF is an adaptation of the method proposed in [9], where the QPP score of a query is used as estimate to decide if PRF should be applied for that query (see $\theta(Q)$ in Section 3.1). The idea here is that a high QPP score is already indicative of an effective retrieval performance, in which case, the method avoids any further risk of potentially degrading the retrieval quality with query expansion. We refer to this method as QPP-based selective relevance feedback (QPP-SRF). The method requires a base QPP estimator to provide θ_{QPP} scores.

To choose the QPP estimator, we conducted a set of initial experiments using several standard unsupervised QPP approaches. The recently introduced supervised QPP method qppBERT-PL [12] demonstrated the best downstream retrieval effectiveness. Therefore, we report results of QPP-SRF combined with qppBERT-PL, where training is conducted using the settings as reported in [12]. A key parameter for QPP-SRF is the threshold value τ ($\tau \in [0, 1]$) which controls the decision around whether PRF is applied or not. In our experiments we tune

Table 1: Summary of the data used in our experiments. The columns '$|\bar{Q}|$' and '$\#\bar{Rel}$' denote average number of query terms and average number of relevant documents.

Collection	#Docs	Topic Set	#Topics	$\|\bar{Q}\|$	$\#\bar{Rel}$
MS MARCO Passage	8,841,823	MS MARCO Train	502,939	5.97	1.06
		TREC DL'19	43	5.40	58.16
		TREC DL'20	54	6.04	30.85

τ on the train folds. To ensure that the threshold can be applied for any QPP estimate, we normalize the QPP estimates in the range $[0, 1]$.

TD2F is an unsupervised selective feedback approach that is conceptually similar to QPP-SRF [9]. Rather than using a QPP method, it computes the difference of the term weight distributions across the sets of documents retrieved with the original and the expanded queries, i.e., the sets $L_k(Q)$ and $L_k(\phi_M(Q))$ as per our notations introduced in Section 3.1. Formally,

$$\theta(Q) = \frac{1}{|V|} \sum_{t \in V} \log P(t|L_k(Q)) - \log P(t|L_k(\phi_M(Q))), \tag{10}$$

where the set V denotes the vocabulary of the two lists, i.e., $V = V_{L_k} \cup V_{L_k(\phi_M(Q))}$. As per [9], we set the feedback decision threshold τ to a value such that over 95% of the queries satisfy the criterion that $\theta(Q) \leq \tau$. We name this method as Term Distribution Divergence based Feedback, or TD2F for short.

LR-SRF is the only existing supervised method that uses the query features, along with their top-retrieved documents, to predict the PRF decision [25]. The ground-truth labels for learning the decision function is obtained for a training set of queries with existing relevance assessments, i.e. $y(Q) = \mathbb{I}(\text{AP}(\phi_M(Q)) > \text{AP}(Q))$. The method then uses Equation 4 to train a feature-based logistic regression classifier. In particular, the experiments reported in [25] used the following features for training the logistic regression model: i) the clarity [10] of top-retrieved documents, ii) the absolute divergence between the query model Q and the relevance model [19], iii) the Jensen-Shannon divergence [24] between the language model of the feedback documents, and iv) the clarity of the query language model. We name this method as Regression-based Selective relevance Feedback (LR-SRF).

Proposed methods In addition to conducting experiments with our proposed model Deep-SRF-BERT (Figure 3), we also incorporate confidence-based calibration (as per objective **RQ2**) with rank fusion (Equation 8 and 9), which we denote by adding the suffix R2F2[3].

[3] Implementation available at: `https://github.com/suchanadatta/AdaptiveRLM.git`

4.3 Experimental Setup

Dataset and train-test splits. Our retrieval experiments are conducted on a standard ad-hoc IR dataset, the MS MARCO passage collection [30]. The relevance of the passages in the MS MARCO collection are more of personalized in nature. A common practice is to use the TREC DL topic sets, which contains depth-pooled relevance assessments on the passages of the MS MARCO collection. For TREC DL, we conduct experiments on a total of 97 queries from the years 2019 and 2020 [7,8]. Table 1 provides an overview of the dataset.

We use a random sample of 5% of queries (constituting a total of about 40K queries) to train the supervised models in our experiments, whereas evaluation is conducted on the TREC DL (both '19 and '20) query sets. We use a small sample from the training set since the training process requires executing a feedback model (e.g., RLM) for all queries. Therefore, the model needs to learn a task-specific encoding for each query-document pair, both for the original and the expanded queries.

To investigate the generalisation of our selective feedback model, we employ RLM as the feedback approach to train the decision function (Figure 3). During inference, we employ three different PRF approaches, namely RLM, ColBERT-PRF [41] and GRF [26] to test the effectiveness of selective feedback.

Parameter settings. A common parameter for all the methods is the number of top-retrieved documents k used for the feedback process and also for training the supervised PRF decision models. For each method we tune the $k \in [5, 40]$ via grid search on the training folds, and use the optimal value on the test fold. We use the same approach to tune the parameter α in Equation 8, which controls the importance of the feedback process for the rank-based fusion methods. For the R2F2-based methods, we conduct a grid search for α in the set $\{0, 0.1, \ldots, 1\}$. The number of terms used for relevance feedback was tuned for the collection and we use the optimal value across all the methods considered.

To obtain the initial retrieval list, we use both sparse and dense models. As a sparse model, we employ BM25 [33] to retrieve the top-1000 results from MS MARCO collection and a supervised neural model, namely, MonoT5 [31] which operates by reranking the top-1000 of BM25. MonoT5 model was trained on the MS MARCO training queries.

4.4 Results and Discussion

Main observations. The key findings of our experiments are reported in Table 2. We observe that the accuracy of the decisions is quite satisfactory, even for the unsupervised threshold-based approaches. The results also indicate that more accurate PRF decisions usually lead to an increase in retrieval effectiveness.

For **RQ1**, we find that supervised selective PRF approaches yield improved results over their unsupervised counterparts. Of particular interest is the fact that a data-driven approach (as per our proposal in this paper) outperforms the feature-based approach LR-SRF [25], which answers RQ1 in the affirmative.

Table 2: Comparison of different SRF approaches on the TREC DL (2019 and 2020) topic sets with BM25 and MonoT5 set as the initial retrieval models. MAP values are computed for top-1000 documents. Paired t-test ($p < 0.05$) shows a significant improvement of Deep-SRF over the best performing baselines (comparing bold-faced results with the underlined ones).

		BM25 (ϕ: RLM)			BM25 (ϕ: GRF)			BM25 (ϕ: ColBERT-PRF)		
	Methods	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.3766	0.5022	N/A	0.3766	0.5022	N/A	0.3766	0.5022
	PRF	N/A	0.4321	0.5134	N/A	0.4883	0.6226	N/A	0.4514	0.6067
	R2F2	N/A	0.4381	0.5140	N/A	0.5094	0.6332	N/A	0.4968	0.6184
	QPP-SRF	0.7835	0.4400	0.5152	0.7844	0.5321	0.6667	0.7742	0.5238	0.6400
	TD2F	0.7611	0.4392	0.5135	0.7580	0.4579	0.5900	0.7642	0.4910	0.6038
	LR-SRF	0.7842	0.4411	0.5154	0.7784	0.5107	0.6512	0.7854	0.5254	0.6414
Ours	Deep-SRF-BERT	**0.8081***	0.4705	0.5374	**0.8093**$*$	0.5654	0.6821	**0.8165***	0.5631	0.6765
	Deep-SRF-BERT-R2F2		**0.4961**	**0.5486**		**0.5730**	**0.6839**		**0.5785**	**0.6873**
	Oracle	1.0000	0.5038	0.5528	1.0000	0.5876	0.6941	1.0000	0.5820	0.6936

		MonoT5 (ϕ: RLM)			MonoT5 (ϕ: GRF)			MonoT5 (ϕ: ColBERT-PRF)		
	Methods	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.5062	0.6451	N/A	0.5062	0.6451	N/A	0.5062	0.6451
	PRF	N/A	0.5081	0.6463	N/A	0.5200	0.6487	N/A	0.5297	0.6491
	R2F2	N/A	0.5112	0.6484	N/A	0.5241	0.6494	N/A	0.5324	0.6502
	QPP-SRF	0.7963	0.5189	0.6559	0.7871	0.5313	0.6604	0.7900	0.5419	0.6673
	TD2F	0.7789	0.5071	0.6453	0.7670	0.4991	0.6403	0.7612	0.5179	0.5986
	LR-SRF	0.7958	0.5180	0.6543	0.7980	0.5422	0.6628	0.7928	0.5500	0.6654
Ours	Deep-SRF-BERT	**0.8152***	0.5306	0.6640	**0.8160**$*$	0.5529	0.6694	**0.8067**$*$	0.5624	0.6733
	Deep-SRF-BERT-R2F2		**0.5317**	**0.6659**		**0.5607**	**0.6719**		**0.5711**	**0.6746**
	Oracle	1.0000	0.5416	0.6786	1.0000	0.5722	0.6803	1.0000	0.5801	0.6821

In relation to **RQ2**, we see that a soft combination of the initial and the feedback lists via a confidence-based calibration (Deep-SRF-BERT-R2F2) improves results further.

An interesting finding is that the SRF decision function trained on RLM on a set of queries generalises well not only for a different set of queries (the test set), but also across different feedback models. This suggests that the queries which improve with RLM also improve with other feedback models, such as GRF or ColBERT-PRF. This can be seen from the GRF and the ColBERT-PRF group of results for both BM25 and MonoT5. This entails that the SRF based decision function does not need to be trained for specific PRF approaches, which makes it more suitable to use in a practical setup, affirming **RQ3**.

We observe that the best results obtained by our method are close to those achieved by an 'oracle'. In the ideal oracle scenario, PRF is applied *only if* the AP of a query is actually improved (i.e., the oracle uses the relevance assessments for the test queries). The fact that the results from Deep-SRF-BERT are close to the oracle suggests that further attempts to increase the accuracy of PRF decisions may have little impact on retrieval effectiveness, likely due to saturation effects.

Per-query analysis. Table 3 shows examples of queries from the TREC DL dataset. Firstly, we see that the average differences in the AP values before and after feedback are mostly higher for the green cells, which indicates that

Table 3: Contingency tables of the Deep-SRF-BERT model with sample queries from TREC DL. Here, $|Q|$ is the count of queries for each of the 4 possible cases of prediction (true/false positives and true/false negatives), and $\overline{\Delta\text{AP}}$ denotes the average ΔAP values of each cell, where $\Delta\text{AP}(Q) = \frac{\text{AP}(\phi_M(Q)) - \text{AP}(Q)}{\text{AP}(Q)}$.

Predicted \ **Actual**	$\Delta\text{AP} > 0$		$\Delta\text{AP} \leq 0$	
$\Delta\text{AP} > 0$	What is active margin? Exon definition Biology	$\|Q\| = 59$ $\overline{\Delta\text{AP}} = 0.1302$	Why is Pete Rose banned from hall of fame? What are best foods to lower cholesterol?	$\|Q\| = 8$ $\overline{\Delta\text{AP}} = 0.0525$
$\Delta\text{AP} \leq 0$	Define BMT medical Who is Robert Gray?	$\|Q\| = 11$ $\overline{\Delta\text{AP}} = 0.0246$	Do Google docs auto save? How many sons Robert Kraft has?	$\|Q\| = 19$ $\overline{\Delta\text{AP}} = 0.0737$

the penalty incurred due to queries for which the model (Deep-SRF-BERT) predicts incorrectly is not too high. This also conforms to the fact that at close to 80% accuracy, Deep-SRF-BERT achieves results close to the oracle. Secondly, a manual inspection of the examples reveals that the queries for which the Deep-SRF-BERT model correctly decides to apply PRF appear to be those with under-specified information needs, i.e., those queries which are likely to be benefited from enrichment, e.g., the query 'what is active margin' as seen from Table 3.

5 Conclusions and Future Work

In this paper, we proposed a selective relevance feedback framework that includes a data-driven supervised neural approach to optimize retrieval effectiveness by applying feedback on queries in a selective fashion. By testing this approach using multiple PRF models over sparse and dense architectures, we observed that it performs favorably compared to alternative strategies, approaching the performance of an oracle system.

This work opens the door for interesting future studies. Although our method is effective, it requires executing PRF to gauge result quality. Exploring techniques to determine the necessity of the PRF step could reduce computational costs for queries where PRF is ultimately ignored. Further work could also examine strategies for predicting the parameters of PRF itself, such as the number of relevant documents.

Acknowledgement The first and the fourth authors were partially supported by Science Foundation Ireland (SFI) grant number SFI/12/RC/2289_P2.

References

1. Bashir, S., Rauber, A.: Improving retrievability of patents with cluster-based pseudo-relevance feedback documents selection. In: Proc. CIKM '09. p. 1863–1866. ACM, New York, NY, USA (2009)

2. Belkin, N.J., Oddy, R.N., Brooks, H.M.: Ask for information retrieval: Part i. background and theory. Journal of documentation (1982)
3. Billerbeck, B., Zobel, J.: Questioning query expansion: An examination of behaviour and parameters. In: Proc. 15th Australasian Database Conference - Volume 27. p. 69–76. ADC '04, Australian Computer Society, Inc., AUS (2004)
4. Cao, G., Nie, J.Y., Gao, J., Robertson, S.: Selecting good expansion terms for pseudo-relevance feedback. In: Proc. SIGIR '08. p. 243–250. ACM, New York, NY, USA (2008)
5. Cohen, D., Mitra, B., Lesota, O., Rekabsaz, N., Eickhoff, C.: Not all relevance scores are equal: Efficient uncertainty and calibration modeling for deep retrieval models. In: Proc. SIGIR'21. pp. 654–664. ACM, New York, NY, USA (2021)
6. Cormack, G.V., Clarke, C.L.A., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proc. SIGIR '09. p. 758–759. ACM, New York, NY, USA (2009)
7. Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the TREC 2020 deep learning track. In: Proc. TREC 2020. NIST Special Publication, vol. 1266 (2020)
8. Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M.: Overview of the trec 2019 deep learning track (2019)
9. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: A framework for selective query expansion. In: Proc. CIKM '04. pp. 236–237. ACM, New York, NY, USA (2004)
10. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: Proc. SIGIR '02. p. 299–306. ACM, New York, NY, USA (2002)
11. Datta, S., Ganguly, D., Greene, D., Mitra, M.: Deep-qpp: A pairwise interaction-based deep learning model for supervised query performance prediction. In: Proc. WSDM '22. pp. 201–209. ACM, New York, NY, USA (2022)
12. Datta, S., MacAvaney, S., Ganguly, D., Greene, D.: A 'pointwise-query, listwise-document' based query performance prediction approach. In: Proc. SIGIR '22. pp. 2148–2153. ACM, New York, NY, USA (2022)
13. Deveaud, R., Mothe, J., Ullah, M.Z., Nie, J.Y.: Learning to adaptively rank document retrieval system configurations. ACM Trans. Inf. Syst. **37**(1) (2018)
14. Ganguly, D., Leveling, J., Jones, G.J.F.: Cross-lingual topical relevance models. In: COLING. pp. 927–942. Indian Institute of Technology Bombay, India (2012)
15. He, B., Ounis, I.: Combining fields for query expansion and adaptive query expansion. Inf. Process. Manage. **43**(5), 1294–1307 (sep 2007)
16. He, B., Ounis, I.: Finding good feedback documents. In: Proc. CIKM '09. p. 2011–2014. ACM, New York, NY, USA (2009)
17. Jaleel, N.A., Allan, J., Croft, W.B., Diaz, F., Larkey, L.S., Li, X., Smucker, M.D., Wade, C.: Umass at TREC 2004: Novelty and HARD. In: TREC 2004 (2004)
18. Khattab, O., Zaharia, M.: ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT, p. 39–48. ACM, New York, NY, USA (2020)
19. Lavrenko, V., Croft, W.B.: Relevance based language models. In: Proc. SIGIR '01. pp. 120–127. ACM, New York, NY, USA (2001)
20. Lee, K.S., Croft, W.B., Allan, J.: A cluster-based resampling method for pseudo-relevance feedback. In: Proc. SIGIR '08. p. 235–242. ACM, New York, NY, USA (2008)
21. Li, C., Sun, Y., He, B., Wang, L., Hui, K., Yates, A., Sun, L., Xu, J.: NPRF: A neural pseudo relevance feedback framework for ad-hoc information retrieval. In: Proc. EMNLP '18. pp. 4482–4491. ACL, Brussels, Belgium (2018)

22. Li, H., Mourad, A., Koopman, B., Zuccon, G.: How does feedback signal quality impact effectiveness of pseudo relevance feedback for passage retrieval. In: Proc. SIGIR '22. p. 2154–2158. ACM, New York, NY, USA (2022)
23. Li, H., Wang, S., Zhuang, S., Mourad, A., Ma, X., Lin, J., Zuccon, G.: To interpolate or not to interpolate: Prf, dense and sparse retrievers. In: Proc. SIGIR '22. p. 2495–2500. ACM, New York, NY, USA (2022)
24. Lin, J.: Divergence measures based on the shannon entropy. IEEE Trans. Inf. Theor. **37**(1), 145–151 (sep 2006)
25. Lv, Y., Zhai, C.: Adaptive relevance feedback in information retrieval. In: Proc. CIKM '09. p. 255–264. ACM, New York, NY, USA (2009)
26. Mackie, I., Chatterjee, S., Dalton, J.: Generative relevance feedback with large language models. In: Proc. SIGIR '23. pp. 2026–2031. ACM, New York, NY, USA (2023)
27. Mitra, M., Singhal, A., Buckley, C.: Improving automatic query expansion. In: Proc. SIGIR '98. p. 206–214. ACM, New York, NY, USA (1998)
28. Montazeralghaem, A., Zamani, H., Allan, J.: A reinforcement learning framework for relevance feedback. In: Proc. SIGIR'20. pp. 59–68. ACM, New York, NY, USA (2020)
29. Naseri, S., Dalton, J., Yates, A., Allan, J.: Ceqe: Contextualized embeddings for query expansion. In: Advances in Information Retrieval. pp. 467–482. Springer International Publishing, Cham (2021)
30. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: MS MARCO: A human generated machine reading comprehension dataset. In: CoCo@NIPS. CEUR Workshop Proceedings, vol. 1773 (2016)
31. Nogueira, R.F., Yang, W., Cho, K., Lin, J.: Multi-stage document ranking with BERT. CoRR **abs/1910.14424** (2019)
32. Ogilvie, P., Voorhees, E., Callan, J.: On the number of terms used in automatic query expansion. Information Retrieval **12**(6), 666–679 (2009)
33. Robertson, S., Walker, S., Beaulieu, M., Gatford, M., Payne, A.: Okapi at TREC-4 (1996)
34. Rocchio, J.J.: Relevance Feedback in Information Retrieval. Prentice Hall, Englewood, Cliffs, New Jersey (1971)
35. Roy, D., Ganguly, D., Mitra, M., Jones, G.J.: Word vector compositionality based relevance feedback using kernel density estimation. In: Proc. CIKM '16. pp. 1281–1290. ACM, New York, NY, USA (2016)
36. Salakhutdinov, R., Mnih, A.: Bayesian probabilistic matrix factorization using markov chain monte carlo. In: Proc. ICML '08. pp. 880–887. ACM, New York, NY, USA (2008)
37. Shtok, A., Kurland, O., Carmel, D.: Using statistical decision theory and relevance models for query-performance prediction. In: In Proc. SIGIR'10. p. 259–266. ACM, New York, NY, USA (2010)
38. Shtok, A., Kurland, O., Carmel, D., Raiber, F., Markovits, G.: Predicting query performance by query-drift estimation. ACM Trans. Inf. Syst. **30**(2) (2012)
39. Terra, E., Warren, R.: Poison pills: Harmful relevant documents in feedback. In: Proc. CIKM '05. p. 319–320. ACM, New York, NY, USA (2005)
40. Wang, X., Macdonald, C., Tonellotto, N., Ounis, I.: Pseudo-relevance feedback for multiple representation dense retrieval. In: ICTIR. pp. 297–306. ACM, New York, NY, USA (2021)
41. Wang, X., MacDonald, C., Tonellotto, N., Ounis, I.: ColBERT-PRF: Semantic pseudo-relevance feedback for dense passage and document retrieval. ACM Transactions on the Web **17**(1), 1–39 (2023)

42. Xiong, L., Xiong, C., Li, Y., Tang, K., Liu, J., Bennett, P.N., Ahmed, J., Overwijk, A.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. In: ICLR (2021)
43. Xu, J., Croft, W.B.: Improving the effectiveness of information retrieval with local context analysis. ACM Transactions on Information Systems **18**(1), 79–112 (2000)
44. Yu, H., Xiong, C., Callan, J.: Improving Query Representations for Dense Retrieval with Pseudo Relevance Feedback, p. 3592–3596. ACM, New York, NY, USA (2021)
45. Zamani, H., Dadashkarimi, J., Shakery, A., Croft, W.B.: Pseudo-relevance feedback based on matrix factorization. In: Proc. CIKM '16. pp. 1483–1492. ACM, New York, NY, USA (2016)
46. Zheng, Z., Hui, K., He, B., Han, X., Sun, L., Yates, A.: BERT-QE: Contextualized Query Expansion for Document Re-ranking. In: Findings of the ACL: EMNLP 2020. pp. 4718–4728. ACL (2020)
47. Zhou, Y., Croft, W.B.: Query performance prediction in web search environments. In: Proc. SIGIR '07. p. 543–550. SIGIR '07, ACM, New York, NY, USA (2007)
48. Zhuang, S., Li, H., Zuccon, G.: Implicit feedback for dense passage retrieval: A counterfactual approach. In: Proc. SIGIR '22. p. 18–28. ACM, New York, NY, USA (2022)