Sampling Query Variations for Learning to Rank to Improve Automatic Boolean Query Generation in Systematic Reviews

Harrisen Scells
University of Queensland
Brisbane, Australia
h.scells@uq.net.au

Guido Zuccon
University of Queensland
Brisbane, Australia
g.zuccon@uq.edu.au

Mohamed A. Sharaf
United Arab Emirates
University, Al Ain, UAE
msharaf@uaeu.ac.ae

Bevan Koopman
CSIRO
Brisbane, Australia
bevan.koopman@csiro.au

ABSTRACT

Searching medical literature for synthesis in a systematic review is a complex and labour intensive task. In this context, expert searchers construct lengthy Boolean queries. The universe of possible query variations can be massive: a single query can be composed of hundreds of field-restricted search terms/phrases or ontological concepts, each grouped by a logical operator nested to depths of sometimes five or more levels deep. With the many choices about how to construct a query, it is difficult to both formulate and recognise effective queries. To address this challenge, automatic methods have recently been explored for generating and selecting effective Boolean query variations for systematic reviews. The limiting factor of these methods is that it is computationally infeasible to process all query variations for training the methods. To overcome this, we propose novel query variation sampling methods for training Learning to Rank models to rank queries. Our results show that query sampling methods do directly impact the ability of a Learning to Rank model to effectively identify good query variations. Thus, selecting appropriate query sampling methods is a key problem for the automatic reformulation of effective Boolean queries for systematic review literature search. We find that the best sampling strategies are those which balance the diversity of queries with the quantity of queries.

ACM Reference Format:
Harrisen Scells, Guido Zuccon, Mohamed A. Sharaf, and Bevan Koopman. 2020. Sampling Query Variations for Learning to Rank to Improve Automatic Boolean Query Generation in Systematic Reviews. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3366423.3380075

1 INTRODUCTION

Systematic reviews form the cornerstone of evidence based medicine. A systematic review (SR) synthesises all relevant literature for a highly focused research question. The majority of medical literature is contained in online databases, e.g., PubMed. A key process in creating SRs is the formulation of Boolean queries which retrieve studies that are then screened (assessed) for inclusion in the review. Query formulation ultimately influences both the costs and quality of the review. These queries are unlike those used in traditional web search tasks. Queries used for SR literature search are often large and complex, with comprehensiveness in mind. The complexity of these Boolean queries arise from nested Boolean clauses, database field restrictions, time, date, study, and language restrictions, ontological hierarchies, among other operators. Query formulation in this context typically requires extensive effort from information specialists: trained librarians with deep knowledge of the search system and the study matter. The role of an information specialists is to balance the cost of screening every study with the requirement of retrieving all relevant studies. A narrow query results in many false negatives; i.e., studies relevant to the review but not retrieved, introducing bias and possibly leading the review to an incorrect conclusion [20]. A broad query results in many false positives, increasing the time and costs required to complete a review. The trade-off between narrow and broad queries has real monetary consequences. Currently, on average, a SR takes up to two years and costs in excess of a quarter of a million dollars [19], while having low levels of precision [14]. The research underlying this paper aims to reduce the time and costs associated with SR creation by focusing on the query formulation phase [30]. The query impacts all downstream activities of the review (e.g., screening, synthesis), so improving this process is key to reducing the overall time and cost of SRs. The outcomes of this paper are methods to *support* automatic Boolean query formulation by investigating how different sampling strategies affect Learning to Rank (LTR) models. These models rank query variations to be used in place of queries initially built manually, in an attempt to greatly improve effectiveness [28, 30]. Empirical evaluation of queries authored by information specialists shows that they do not always produce effective queries [28]. Recent work also shows that queries produced by information specialists can be improved by automatically generating query variations using the query transformation chain (QTC) framework [30]. The QTC framework produces alternative queries by modifying the syntax and semantics of a starting Boolean query using predefined transformations. This results in the creation of a 'universe' of possible alternative queries through the iterative application of transformations. Candidate selection is then responsible for the automatic identification of one (or more) effective query variations to continue the process. Scells et al. [28] have proposed a number of candidate selection techniques, and empirically found LTR to be the most effective. One key limitation with QTC, is *the approach used for sampling training queries*. The process of generating query variations for training produces a computationally infeasible amount of queries. The computational infeasibility is worsened when (i) the original query has many clauses, (ii) when many transformations are applicable to each clause, and (iii) when many iterations of the process are considered: all typical conditions of queries in the context of this paper. The computational complexity required for generating the training queries is exponential

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380075

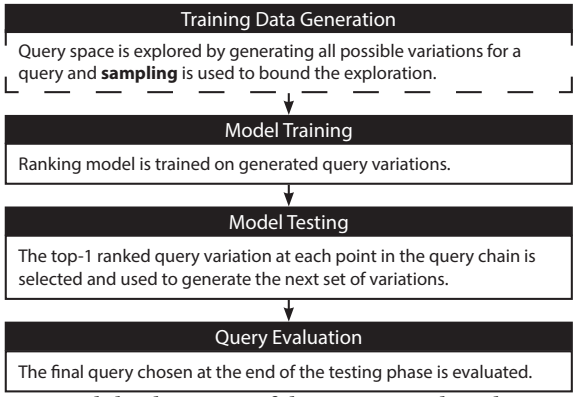


Figure 1: High-level overview of the experimental pipeline. Emphasis is placed on generation (dotted box), where training queries are created by exploring the query space. Exploration is computationally expensive; thus bounded by the sampling methods. In the testing phase, sampling is not required: only the selected query in each step of the chain is used to generate the next set of variations.

in nature. If n is the total number of variations at one step in the chain, and t is the number of transformations, then the complexity of generating training queries is $O(n^t)$. One solution to the problem was to randomly or greedily sample a predetermined number of query variations [28, 30]. The way query variations are sampled, however, can influence the effectiveness of LTR models trained on these queries. We provide evidence for this with a case study in Section 2 which establishes the motivations for the remainder of this paper. Following this, we investigate the impact sampling has on the effectiveness of trained models and establish novel sampling strategies for this problem. Furthermore, previous work has considered exploring the space of candidate queries using a breadth-first exploration method. In this paper, we contribute a depth-first exploration method and adapt applicable sampling strategies. Figure 1 presents a high-level overview of the experimental pipeline for QTC, where the sampling experiments fit, the impact sampling has on the rest of the pipeline. The aim of this paper is to show the impact different sampling strategies have on LTR models, balancing computational time with the number of sampling strategies tested.

2 MOTIVATING EXAMPLE

We motivate this work with an example where training query variations are sampled according to three naïve strategies: (1) sample queries which improve over the original query (*positive*) (2) sample queries which do not improve over the original query (*negative*), and (3) sample queries *randomly*. Improvement over the original query is measured using precision, recall, and F_1 measure. Evaluation is achieved by using studies that were marked for possible inclusion in the original review, as retrieved by the original query. Distributions of sampled queries with respect to F_1 are presented in Figure 2. The random distribution is the same for both plots and is included in both for comparison. Both the precision- and recall-based distributions using positive strategies have queries with a higher average F_1 compared to negative. The randomly sampled distribution of queries has the lowest average F_1 in both cases.

Next, three LTR models are trained using queries from each sampling strategy, optimising for either precision or recall. The use of evaluation measures for optimisation is adopted from previous work [28] and is done to cater for standard SRs or alternative types

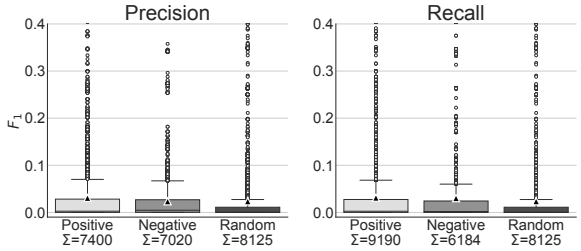


Figure 2: Distributions of *sampled* queries using strategies that sample positively, negatively, and randomly. Average is signified by ▲. Total samples in each distribution is signified by Σ. Queries with F_1 >0.4 are omitted for clarity.

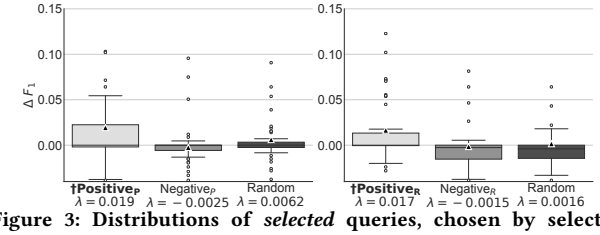


Figure 3: Distributions of *selected* queries, chosen by selectors trained on only positive queries (precision: Positive$_P$, recall: Positive$_R$), only negative queries (precision: Negative$_P$, recall: Negative$_R$), and randomly selected queries. Averages (λ) are signified by ▲. Queries with F_1 >0.15 are omitted for clarity. Statistical significance with Bonferroni correction ($p < 0.05$) between original queries and selected queries is signified by †.

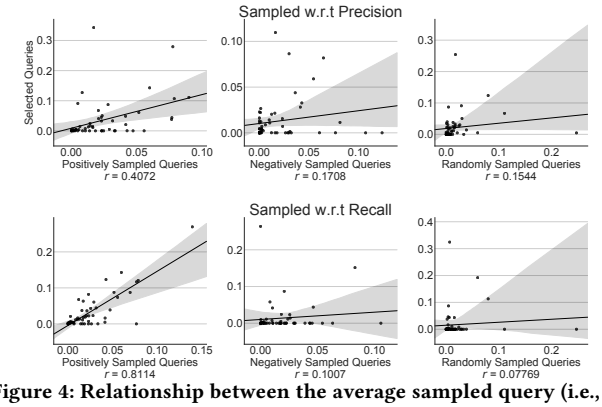


Figure 4: Relationship between the average sampled query (i.e., average value of queries from each sampled distribution; those in Figure 2) and selected queries (i.e., those in Figure 3). Horizontal axis: average F_1 of sampled queries. Vertical axis: F_1 of selected queries. Linear relationship is signified by the solid black lines. Pearson's r between sampled and selected queries is recorded beneath.

of reviews such as rapid reviews (i.e., where total recall requirements are traded off for high levels of precision). Note that the optimisation of the LTR model is different to how queries are sampled: optimisation refers to the objective function of the LTR model. Only the LTR model trained on queries sampled using precision optimises for precision. Likewise, only the LTR model trained on queries sampled using recall optimises for recall. Both the model optimising for precision and the model optimising for recall are trained on the same randomly sampled queries. Six LTR models in total are created.[1] A summary of the results is presented in Figure 3. The order of each box-plot is the same as in Figure 2. Selected queries are evaluated with respect to gains and losses in F_1 over

[1] (3 sampling strategies×optimise precision)+(3 sampling strategies×optimise recall)

the original queries. The results for the LTR model trained on positively sampled precision queries, optimised for precision (Positive$_P$) showcase best the impact of sampling. The Positive$_P$ model selects query variations that are on average statistically significantly better than the original queries ($p < 0.05$). Likewise, the counterpart positive recall optimised model (Positive$_R$) selects query variations that are also on average statistically significantly better than the original queries, but not as effective as the precision-based model. Meanwhile, both models trained on the negatively sampled queries (Negative$_P$ and Negative$_R$) select queries which are less effective than the original queries. The models trained on queries sampled randomly (Random$_P$ and Random$_R$) select queries with effectiveness somewhere between the two other models. The relationship between sampled queries for training a LTR model and queries selected by the model trained on those sampled queries is presented in Figure 4. Queries selected by models trained on positive samples are strongly positively correlated with each other. Meanwhile, negatively and randomly samples are only weakly positively correlated.

These results highlight the importance of sampling in training LTR models in this context. The consequence of training a LTR model on only queries that are more effective than the original query (positive) is that selected variations are significantly more effective than the original query. On the other hand, LTR models trained on negative or random samples result in the selection of variations that are less effective than the original query. Indeed, the relationship between selected queries and queries used for training is correlated with effectiveness. This motivates the work and methods presented in this paper.

3 AUTOMATIC QUERY TRANSFORMATIONS

The QTC framework proposed by Scells and Zuccon [28] sampled queries using a greedy strategy. Another study [30] which used QTC instead sampled queries randomly. Both approaches to sampling used the same *exploration method* (breadth-first), but a different *sampling strategy* (greedy vs. random). In this paper, we add to and build upon these previous sampling strategies, and propose a new exploration method based on depth-first search. Next, we provide a working description of QTC for generating and selecting variations of a Boolean query. Given an input Boolean query, the QTC first identifies the set of transformations $\mathcal{T}'$ that are applicable, defined as: $\mathcal{T}' = \{\forall \tau \in \mathcal{T} | a(\tau, c) = 1\}$, where $\mathcal{T}$ is the set of transformations, and a the applicability function, defined as:

$$a(\tau, c) = \begin{cases} 1, & \text{if } \tau \text{ is applicable to clause } c \\ 0, & \text{otherwise} \end{cases}$$

A clause c is any component of a Boolean query: an individual keyword or a grouping of keywords and other clauses. The applicability of a transformation to a clause is dependent on the intrinsic aspects of a clause.Once transformations have been applied to a input query q, a candidate query q^* is selected from the set $\hat{\mathbb{Q}}_q$ which includes the generated candidates and the original query. At each point in the chain, a candidate query q^* is selected by maximising a *candidate selection function*: $q^* = \text{argmax}_{\hat{q} \in \hat{\mathbb{Q}}_q} f(\hat{q})$. This function can be instantiated according to a classification or a ranking task. Empirical results in previous work have shown ranking to be superior [28]. In previous studies using QTC, the objective of the learning method is to maximise an evaluation measure [28, 30]. Thus, in order to train such a model, examples of queries and evaluation scores (i.e., labels) must be provided. One trivial method to generate large amounts of training queries is query variations. However, as it is computationally infeasible to explore all possible variations of a query, *sampling* must be applied.

4 SAMPLING

We propose two **exploration methods** for sampling the query space to obtain training queries: *breadth-first* and *depth-first*. Each of these methods can be instantiated according to a number of **sampling strategies**, which each lead to sampling different queries.

4.1 Breadth-First Exploration

The breadth-first exploration method uses pooling to sample queries through reduction. The amount of reduction is determined by two parameters which controls the ratio of queries to sample: n, controls the *minimum* number of queries to sample; δ, controls the percentage of queries to sample. If $|\hat{Q}| < n + (\delta \times |\hat{Q}|)$, all queries are included in the sample. Queries are sampled into each strata randomly. The sampling strategies using breadth-first exploration we consider are:

4.1.1 Greedy Sampling Strategies. These sampling strategies are adapted from the greedy candidate selector of Scells and Zuccon [28]. With these strategies, queries are sampled by choosing those which minimise the number of retrieved documents while maximising an evaluation measure. In addition to the n and δ parameters, an evaluation measure has to be specified.

- **Greedy Naive**: Minimise the number of citations retrieved by the candidate query (N, where $N > 0$), while maximising the number of relevant citations retrieved.
- **Greedy Diversity**: Maximise the evaluation score while diversifying the queries. We use Maximal Marginal Relevance (MMR) [7] as our diversity function. To apply MMR to query re-ranking, we replace the similarity between a document and query of the original MMR formulation with the evaluation measure of the query ($Ev(Q)$). Thus, $\texttt{score} = \lambda \cdot Ev(Q) + (1 - \lambda) \cdot Sim(q_i, Q)$. We use cosine similarity for $Sim()$.

4.1.2 Evaluation Sampling Strategies. These strategies use explicit relevance judgements for sampling queries – this is possible as sampling is only performed on training data (i.e. no need for relevance judgements when reformulating a query for a new SR). Only the scores for a given evaluation measure are considered, thus each strategy below must be paired with an evaluation measure.

- **Evaluation Stratified**: Two strata are obtained by sampling proportionately across pooled candidate queries.
- **Evaluation Balanced**: Sample uniformly according to the score of a given evaluation measure across pooled candidate queries.
- **Evaluation Positively Biased**:Candidates are sampled only when they are more effective than the seed query.
- **Evaluation Negatively Biased**: Candidates are sampled only when they are less effective than the seed query.
- **Evaluation Diversity**: Re-rank pooled candidate queries by applying MMR to the score of a given evaluation measure for queries, then sample uniformly across the new distribution.

4.1.3 Transformation Sampling Strategies. The types of transformations applied to generate each variation are considered when sampling. Transformation samplers aims to sample across the types of transformations applied to queries.

- **Transformation Stratified**: Stratified (proportional) sampling across the pooled queries. The number of strata is equal to the total number of types of transformations in the pool.
- **Transformation Balanced**: Balanced (uniform) sampling across the pooled candidate queries. Candidates are balanced into the number of types of transformations in the candidate pool.

4.1.4 Clustering Sampling Strategy. k-means++ [4] with $k = 5$ clusters queries using features from [28]. Queries are sampled from each cluster in a round-robin fashion, up to the cut-off n.

4.1.5 Random Sampling Strategy. Included as a naïve approach to sampling to determine if other techniques provide significant benefits over random sampling. This approach has been used in previous work [30]. Candidates are sampled uniformly.

4.2 Depth-First Exploration

The depth-first exploration method uses depth-first search to traverse query chains. Candidate queries are sampled in the same fashion as in Section 4.1, however rather than sampling from a pool of queries, the chain of previous queries is used to determine inclusion in the sample set. To adapt breadth-first strategies to depth-first, a cost-based approach is used. Each strategy has a budget of how many queries may be sampled. The benefit of using the depth-first exploration method over the breadth-first method is that queries can be sampled with respect to a sequence of transformations, rather than pooling at depth. The following strategies are be adapted to depth-first:

- **Evaluation Positively Biased**: Sample candidates where, given an evaluation measure, the chain of queries including the most recent query are more effective than the original query.
- **Evaluation Negatively Biased**: Choose candidates where, given an evaluation measure, the chain of queries including the most recent query are less effective than the original query.
- **Transformation Balanced**: Sample candidates where the chain of transformations applied to previous candidates is balanced in terms of which transformations have been applied.
- **Random**: Sample candidates according to a uniform likelihood.

In addition to adapted strategies, the following strategy is unique to the depth-first exploration method:

- **Transformation Biased**: Choose candidates where the chain of transformations applied to previous candidates are all the same as the most recent transformation applied.

5 EXPERIMENTAL SETUP

Experiments are performed using 125 SR queries from Scells et al. [31]. Evaluation considers relevant studies as those retrieved by the original query and marked as eligible to be included in the final review (abstract-level relevance). When sampling queries using breadth-first exploration, n is set to 10 and δ is set to 0. Using depth-first exploration, a budget of 65 queries was found through empirical experimentation to produce a similar number of sampled queries to breadth-first exploration strategies. Each sampled query is evaluated using Precision, Recall, and F_1. The evaluation measure is used as the label for training LTR models.For diversity-based sampling strategies (*Greedy Diversity* and *Evaluation Diversity*), λ is set to 0.3, trading similar scoring queries for diverse queries. For random sampling strategies, the likelihood of sampling is set to 65%. The PubMed entrez API [27] is used as the retrieval system. For each sampling strategy, a LTR model for each evaluation measure is trained and evaluated. All queries are used to train LTR models using five-fold cross validation, split into 80% training and 20% validation. In total, a model is trained using each of 25 sampling strategies × 6 target evaluation measures × 5 folds: 750 LTR models. The LTR model used is the QuickRank [6] implementation of LambdaMART [40]. Each LTR model is set to optimise DCG@1, placing the most importance on the top-1 query (i.e., the query selected to continue the chain). Queries are validated by evaluating them on each of the aforementioned evaluation measures. We use the average F_1 of selected queries to evaluate the performance of the LTR models, as is typically done in this context. The aim of this paper is to address any limitations and downsides to the strategies for sampling queries in previous work [28, 30], and to identify better sampling strategies. The limitations of previous work present three research questions to be addressed:

RQ1: How does sampling affect the distribution of *sampled query effectiveness* within the sampled set?
RQ2: How does sampling affect the distribution of *selected query effectiveness* in the Query Transformation Chain framework?
RQ3: Are there relationships between the set of *sampled queries* and the effectiveness of the *selected queries*?

In **RQ1** and **RQ2** we study the effectiveness of queries with respect to two dimensions of sampling: the **exploration method** and the **sampling strategy**. **RQ1** investigates if sampling affects the distribution of queries: e.g., if there are significant differences when sampled query variations are sampled positively or negatively. **RQ2** investigates if sampled queries have a significant impact on the effectiveness of selected query variations: e.g., if a LTR model trained on positively biased queries selects query variations that are significantly more effective than a LTR model trained on negatively biased queries. **RQ3** investigates any nuanced relationships between sampled and selected queries.

6 RESULTS & ANALYSIS

Next, we present the analysis and results of the sampling experiments; each of the following sections corresponds to a research question. The baseline is the original queries. Gains are therefore measured with respect to these (unless otherwise stated).

RQ1: Distributions of Sampled Queries Figure 5 presents the distributions of F_1 and size of sampled queries for each sampling strategy in both exploration methods. In both breadth- and depth-first exploration, the queries sampled using Positive$_P$ obtain the highest average F_1. The queries sampled using the Greedy Naïve$_P$ and Greedy Naïve$_R$ obtain the second highest F_1 using breadth-first exploration. Comparing the two Greedy sampling strategies with each other, the F_1 of queries obtained by the Greedy Naïve sampling strategy is, on average, higher than queries obtained using the Greedy Diversified strategy. The queries in all other breadth-first evaluation-based sampling strategies (Balanced, Diversified, Stratified, Negative) obtain similar average F_1 to each other. Furthermore, the queries from the Transformation Stratified, Transformation Balanced, Cluster, and Random sampling strategies all obtain similar average F_1 to the above evaluation-based strategies. For depth-based exploration, all sampled distributions of queries except Positive$_P$ obtain similar average F_1. When observing the number of sampled queries in each distribution and the their average F_1, no correlation was found. This indicates that the scores of queries from each sampling strategy are not influenced by the

number of queries sampled by that strategy. In summary, the choice in sampling strategies does affect the sampled queries – the effect of which is studied in the following research questions.

RQ2: Distribution of Selected Queries Figure 6 presents the selected queries using the same LTR model trained with query variations obtained with different sampling strategies. For the breadth-first sampling strategies, Greedy Diversified $_P$ and Greedy Diversified$_R$ select queries that are significantly more effective than the original. Meanwhile, for depth-first sampling strategies, only the Positive$_P$ model was able to select queries that were significantly more effective than the original. The combination of breadth-first exploration and Greedy Diversity$_P$ sampling lead to a LTR model which selects the most effective queries. Next, the differences between sampling strategies which are applicable in both the breadth- and depth-first exploration are compared. Figure 7 presents the differences between selected queries using the same sampling strategies, but different exploration methods. This figure illustrates that the depth-first sampling strategies which rely on evaluation for sampling produce better training queries for the LTR models than the same breadth-first strategies. However, there is little difference for the Balanced Transformation and Random sampling strategies. This indicates that the exploration method does play a role in the effectiveness of the LTR model, not just the sampling strategy. This comparison highlights the significant impact exploration has on the sampling strategy. Depth-first exploration can significantly improve the effectiveness of selected queries and is a key contribution of this paper. While the number of training queries in each distribution is relatively low (previous work considered approximately 10,000-700,000 training queries *per topic* [28, 30]), impacting the resulting LTR models, the effects of sampling are still highly visible. Generating query variations for the training phase becomes increasingly computationally expensive as more sampling strategies are considered. The number of sampling strategies considered for comparison limits the number of queries that may be generated by each strategy. Also, the original queries have already passed the rigorous scrutiny of *peer review*, and already considered to be highly effective. Any gains by a selected query over the original has considerable time and monetary impacts [19]. These results suggest that sampling only by query effectiveness is not the best strategy. The combination of effectiveness and diversity that the breadth-first Greedy Diversified sampling strategies permit results in the most effective LTR models.

RQ3: Sampled and Selected Relationship The first the relationship we investigate is between sampled and selected queries is investigated. Figure 8 presents the relationship for the best and worst queries using the breadth- and depth-first exploration methods (identified in RQ2). Selected queries are weakly positively correlated with the sampled queries (Figure 8, right). Queries selected by these best performing models are moderately correlated with sampled queries (Figure 8, left). This suggest that particular sampled queries do impact the effectiveness of LTR models. The second relationship we investigate is between the number of sampled queries for each model and the effectiveness of queries selected using that model (Figure 9). A moderate positive correlation is observed, suggesting increasing the amount of training queries leads to more effective LTR models. In summary, there are relationships between the *sampled* queries and the *selected* queries. These relationships

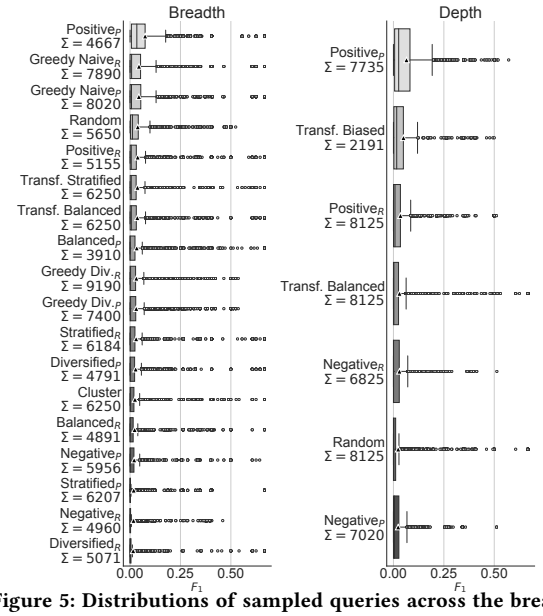


Figure 5: Distributions of sampled queries across the breadth-first (left) and depth-first (right) sampling strategies. Averages are signified by ▲. Total samples in each distribution is signified by Σ.

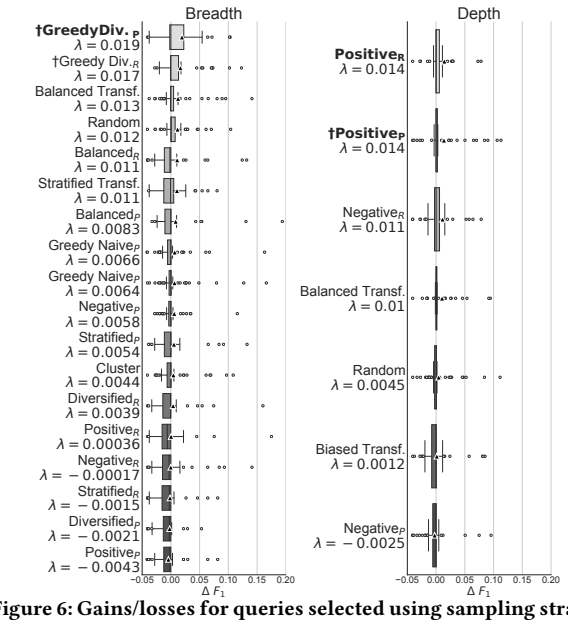


Figure 6: Gains/losses for queries selected using sampling strategies with breadth-first (left) and depth-first exploration methods (right). Averages (λ) are signified by ▲. Statistical significance with Bonferroni correction ($p < 0.05$) is signified by †. The most effective model(s) for each exploration method is highlighted in bold.

empirically suggest that (a) a well-distributed sampled set of queries (in terms of effectiveness), and (b) the quantity of queries sampled, both lead to more effective models.

7 RELATED WORK

We consider the problem of reducing the cost and time associated with the construction of SRs. SRs require considerable effort to construct and often become out of date by the time of publication [35]. The average SR takes upwards of 2 years and USD $230K to create [19]. Currently only 36% of Cochrane SRs are deemed up-to-date. Previous attempts to reduce costs attempt to automate

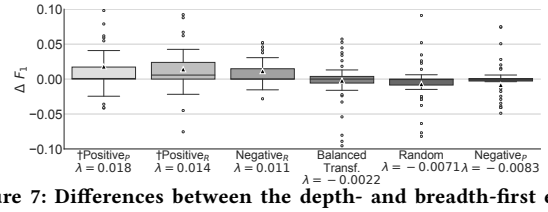


Figure 7: Differences between the depth- and breadth-first exploration. Averages (λ) signified by ▲. Statistical significance with Bonferroni correction ($p < 0.05$) is indicated by †.

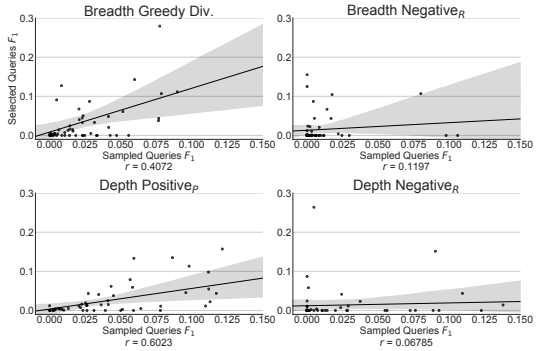
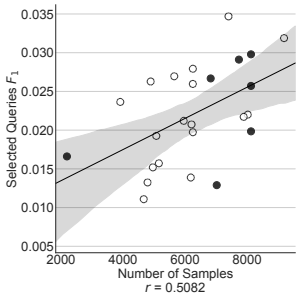


Figure 8: Relationship between average of queries in Figure 5, and selected queries from Figure 6 for best and worst models for breadth- and depth-first exploration. Horizontal axis is average F_1 of sampled queries. Vertical axis is F_1 of selected queries. Linear relationship between these variables is signified by black line. Pearson's r between sampled and selected queries reported beneath.

downstream processes, mostly by text mining methods to support the appraisal [10, 25], analysis [37] and synthesis [26, 33, 34] of studies post search. An emerging area of research in this context that exploits advances in information retrieval is screening prioritisation: ranking retrieved studies [1–3, 9, 15, 16, 23, 29, 39, 42]. This however does not reduce screening costs: all studies are still required to be screened. Instead, it possibly shortens the total time to compile the review: when a relevant study is identified when screening, it can be forwarded downstream. We tackled the problem by addressing query formulation. Directly addressing the problem at this stage represents a novel direction of research in comparison with previous work, and directly impacts downstream tasks of the SR creation pipeline. Syntactic and semantic modifications of queries (e.g., expansion/reduction) have been shown to significantly improve effectiveness of queries in general and specific domains outside of that considered here [8, 12, 17, 22, 41]. The QTC framework aims to automatically modify Boolean queries for SR literature search [28, 30]. One issue raised in these works is the computational and storage challenge of generating training data. This paper investigated novel sampling strategies to address these issues and how they impact resulting LTR models.Researchers have noticed that of sampling may have a higher impact on effectiveness than specific aspects of any learning algorithm. Wu et al. [38] identified that sampling in the context of Deep Embedding Learning has a much higher impact on model effectiveness than the loss function. In the context of LTR, Donmez et al. [11] and Aslam et al. [5] found that sampling methods have a significant impact on the effectiveness of ranking models. Donmez et al. used an active sampling approach to maximise the estimated loss differential over unlabelled data. While Aslam et al. investigated the effect sampling has on LTR in a typical ad-hoc document retrieval setting. Their work uses sampling methods that do and do not use prior relevance knowledge to sample,

Figure 9: Relationship between all queries selected using a model trained on all sampling methods (breadth-first methods: white, depth-first methods: black) and the number of samples used to train each model. The linear relationship between these variables is signified by a solid black line.



and evaluates sampling methods by how well a LTR model trained on different sampling methods can rank documents given a query. This setting is similar to the one we study in our work, although we rank queries not documents. Similar studies have also investigated the impact of sampling in a traditional LTR context from different perspectives. For example, Kanoulas et al. [13] considered the distribution of positive and negative training examples on a large scale, and Lucchese et al. [18] investigated negative sampling to improve LTR models. More related to this work is the study by Mehrotra et al. [21] who developed query sampling methods to reduce labelling costs for training an Active LTR model. Key to the effectiveness of the selection of queries in that work is the informativeness and representativeness characteristics of queries but it is unclear how to apply these techniques in the context of this work.

8 CONCLUSIONS & FUTURE WORK

We present novel query sampling strategies within the context of SR literature search. We also devise and formalise two exploration methods for traversing query variation space that embed sampling strategies. Sampling queries in this domain is necessary as queries used to search literature for SRs are verbose and complex: the number of variations produced are computationally infeasible to handle. While sampling cuts down on computation costs it leads to another problem: different subsets of training data result in different effectiveness of LTR models. We empirically show these differences by evaluating each sampling strategy. Our results suggest that sampling strategies that rely on transformations or features of queries are the least effective for training LTR models. The strategies that diversify the effectiveness of queries provide higher gains than those which rely on biasing effectiveness alone. While the gains of selected queries over original queries may seem marginal, small changes to precision while maintaining recall lead to significant reductions in the total time and cost of the SR process [32]. There are many domains that also use Boolean queries such as patent or legal search. Typically, such requirements fall under 'professional search' or 'eDiscovery' [24, 36]. The methods laid out in this work can be easily adopted in these domains. Moreover, our framework can be extended to add additional sampling strategies, e.g., a word embeddings-based strategy. Determining the most effective sampling strategy for training LTR models can lead to better methods for assisting information specialists formulate queries (e.g., automatic query recommendation and refinement). This can lead to significant time and cost savings for the researchers conducting SRs (i.e., less citations to screen as the query provided to them is more effective than one formulated without automatic assistance). Even minor improvements to queries can have a significant impact on SR creation.

Acknowledgements. Harrisen Scells is the recipient of a CSIRO PhD Top Up Scholarship. Dr Guido Zuccon is the recipient of an Australian Research Council DECRA Research Fellowship (DE180101579) and a Google Faculty Award. This research is also supported by the National Health and Medical Research Council Centre of Research Excellence in Informatics and E-Health (1032664).

REFERENCES

[1] Mustafa Abualsaud, Nimesh Ghelani, Haotian Zhang, Mark D Smucker, Gordon V Cormack, and Maura R Grossman. 2018. A system for efficient high-recall retrieval. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1317–1320.

[2] Amal Alharbi, William Briggs, and Mark Stevenson. 2018. Retrieving and ranking studies for systematic reviews: University of Sheffield's approach to CLEF eHealth 2018 Task 2. In *CEUR Workshop Proceedings*, Vol. 2125. CEUR Workshop Proceedings.

[3] Amal Alharbi and Mark Stevenson. 2017. Ranking Abstracts to Identify Relevant Evidence for Systematic Reviews: The University of Sheffield's Approach to CLEF eHealth 2017 Task 2.. In *CLEF (Working Notes)*.

[4] David Arthur and Sergei Vassilvitskii. 2007. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 1027–1035.

[5] Javed A Aslam, Evangelos Kanoulas, Virgil Pavlu, Stefan Savev, and Emine Yilmaz. 2009. Document selection methodologies for efficient and effective learning-to-rank. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 468–475.

[6] Gabriele Capannini, Claudio Lucchese, Franco Maria Nardini, Salvatore Orlando, Raffaele Perego, and Nicola Tonellotto. 2016. Quality versus efficiency in document scoring with learning-to-rank models. *Information Processing & Management* 52, 6 (2016), 1161–1177.

[7] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 335–336.

[8] Claudio Carpineto and Giovanni Romano. 2012. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)* 44, 1 (2012), 1.

[9] Jiayi Chen, Su Chen, Yang Song, Hongyu Liu, Yueyao Wang, Qinmin Hu, Liang He, and Yan Yang. 2017. ECNU at 2017 eHealth Task 2: Technologically Assisted Reviews in Empirical Medicine.. In *CLEF (Working Notes)*.

[10] Gordon V Cormack and Maura R Grossman. 2015. Autonomy and reliability of continuous active learning for technology-assisted review. *arXiv preprint arXiv:1504.06868* (2015).

[11] Pinar Donmez and Jaime G Carbonell. 2008. Optimizing estimated loss reduction for active sampling in rank learning. In *Proceedings of the 25th international conference on Machine learning*. ACM, 248–255.

[12] Kalervo Järvelin and Jaana Kekäläinen. 2000. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 41–48.

[13] Evangelos Kanoulas, Stefan Savev, Pavel Metrikov, Virgil Pavlu, and Javed Aslam. 2011. A large-scale study of the effect of training set characteristics over learning-to-rank algorithms. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM, 1243–1244.

[14] S. Karimi, S. Pohl, F. Scholer, L. Cavedon, and J. Zobel. 2010. Boolean versus ranked querying for biomedical systematic reviews. *BMC MIDM* 10, 1 (2010), 1.

[15] Athanasios Lagopoulos, Antonios Anagnostou, Adamantios Minas, and Grigorios Tsoumakas. 2018. Learning-to-Rank and Relevance Feedback for Literature Appraisal in Empirical Medicine. In *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, 52–63.

[16] Grace E. Lee and Aixin Sun. 2018. Seed-driven Document Ranking for Systematic Reviews in Evidence-Based Medicine. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18)*. ACM, New York, NY, USA, 455–464. https://doi.org/10.1145/3209978.3209994

[17] Daniel Locke, Guido Zuccon, and Harrisen Scells. 2017. Automatic Query Generation from Legal Texts for Case Law Retrieval. In *Asia Information Retrieval Symposium*. Springer, 181–193.

[18] Claudio Lucchese, Franco Maria Nardini, Raffaele Perego, and Salvatore Trani. 2017. The Impact of Negative Samples on Learning to Rank. In *LEARNER*.

[19] Jessie McGowan and Margaret Sampson. 2005. Systematic reviews need systematic searchers (IRP). *Journal of the Medical Library Association* 93, 1 (2005), 74.

[20] Faith McLellan. 2001. 1966 and all that—when is a literature search done? *The Lancet* 358, 9282 (2001), 646.

[21] Rishabh Mehrotra and Emine Yilmaz. 2015. Representative & informative query selection for learning to rank using submodular functions. In *Proceedings of the 38th international ACM sigir conference on research and development in information retrieval*. ACM, 545–554.

[22] Mandar Mitra, Amit Singhal, and Chris Buckley. 1998. Improving automatic query expansion. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 206–214.

[23] M. Miwa, J. Thomas, A. O'Mara-Eves, and S. Ananiadou. 2014. Reducing systematic review workload through certainty-based screening. *JBI* 51 (2014), 242–253.

[24] Douglas W Oard, Jason R Baron, Bruce Hedin, David D Lewis, and Stephen Tomlinson. 2010. Evaluation of information retrieval for E-discovery. *Artificial Intelligence and Law* 18, 4 (2010), 347–386.

[25] Alison O'Mara-Eves, James Thomas, John McNaught, Makoto Miwa, and Sophia Ananiadou. 2015. Using text mining for study identification in systematic reviews: a systematic review of current approaches. *Systematic reviews* 4, 1 (2015), 5.

[26] John Rathbone. 2017. *Automating systematic reviews*. Ph.D. Dissertation. Bond University.

[27] E Sayers and V Miller. 2010. *Entrez programming utilities help [internet]*. National Center for Biotechnology Information (US).

[28] Harrisen Scells and Guido Zuccon. 2018. Generating Better Queries for Systematic Reviews. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR '18)*. ACM, New York, NY, USA, 475–484.

[29] Harrisen Scells, Guido Zuccon, Anthony Deacon, and Bevan Koopman. 2017. QUT ielab at CLEF eHealth 2017 technology assisted reviews track: Initial experiments with learning to rank. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, Vol. 1866. CEUR Workshop Proceedings, Paper–98.

[30] Harrisen Scells, Guido Zuccon, and Bevan Koopman. 2019. Automatic Boolean Query Refinement for Systematic Review Literature Search. In *Proceedings of the 2019 World Wide Web Conference*.

[31] Harrisen Scells, Guido Zuccon, Bevan Koopman, Anthony Deacon, Shlomo Geva, and Leif Azzopardi. 2017. A Test Collection for Evaluating Retrieval of Studies for Inclusion in Systematic Reviews. In *SIGIR'2017*.

[32] Ian Shemilt, Nada Khan, Sophie Park, and James Thomas. 2016. Use of cost-effectiveness analysis to compare the efficiency of study identification methods in systematic reviews. *Systematic reviews* 5, 1 (2016), 140.

[33] James Thomas and Angela Harden. 2008. Methods for the thematic synthesis of qualitative research in systematic reviews. *BMC medical research methodology* 8, 1 (2008), 45.

[34] Mercedes Torres Torres and Clive E Adams. 2017. RevManHAL: towards automatic text generation in systematic reviews. *Systematic Reviews* 6, 1 (2017), 27.

[35] G. Tsafnat, P. Glasziou, M.K. Choong, A. Dunn, F. Galgani, and E. Coiera. 2014. Systematic review automation technologies. *SR* 3, 1 (2014), 74.

[36] Suzan Verberne, Jiyin He, Udo Kruschwitz, Birger Larsen, Tony Russell-Rose, and Arjen P De Vries. 2018. First International Workshop on Professional Search (ProfS2018).. In *SIGIR*. 1431–1434.

[37] Byron C Wallace, Joël Kuiper, Aakash Sharma, Mingxi Brian Zhu, and Iain J Marshall. 2016. Extracting PICO sentences from clinical trial reports using supervised distant supervision. *Journal of Machine Learning Research* 17, 132 (2016), 1–25.

[38] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. 2017. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*. 2840–2848.

[39] Huaying Wu, Tingting Wang, Jiayi Chen, Su Chen, Qinmin Hu, and Liang He. 2018. Ecnu at 2018 ehealth task 2: Technologically assisted reviews in empirical medicine. *Methods* 4, 5 (2018), 7.

[40] Qiang Wu, Christopher JC Burges, Krysta M Svore, and Jianfeng Gao. 2010. Adapting boosting for information retrieval measures. *Information Retrieval* 13, 3 (2010), 254–270.

[41] Jinxi Xu and W Bruce Croft. 1996. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 4–11.

[42] Jie Zou, Dan Li, and Evangelos Kanoulas. 2018. Technology Assisted Reviews: Finding the Last Few Relevant Documents by Asking Yes/No Questions to Reviewers. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 949–952.