Title page

Title of the article:

Refining Boolean Queries to Identify Relevant Studies for Systematic Review Updates

Full name, postal address, e-mail and telephone number of the corresponding author:

Amal Alharbi

211 Portobello, Regent Court

Computer Science Department, University of Sheffield

S1 4DP

Sheffield, United Kingdom

ahalharbi1@sheffield.ac.uk

+447397194194

Full name, department, institution, city and country of all co-authors:

Mark Stevenson

Computer Science Department, University of Sheffield, Sheffield, United Kingdom

keywords:

Systematic reviews; screening; lexical statistics; query reformulation; Systematic reviews updates

Word count, excluding title page, abstract, references, figures and tables: 3,840

ABSTRACT

Objective: Systematic reviews are important in healthcare but are expensive to produce and maintain. This paper explores the use of automated transformations of Boolean queries to improve the identification of relevant studies for updates to systematic reviews.

Materials and Methods: A set of query transformations, including operator substitution, query expansion and query reduction, were used to iteratively modify the Boolean query used for the original systematic review. The most effective transformation at each stage is identified using information about the studies included/excluded from the original review. A dataset consisting of 22 systematic reviews was used for evaluation. Updated queries were evaluated using the included/excluded studies from the updated version of the review. Recall and precision were used as evaluation measures.

Results: The updated queries were more effective than the ones used for the original review, both in terms of precision and recall. The overall number of documents retrieved was reduced by more than half while the number of relevant documents found increased by 10.3%.

Conclusion: Identification of relevant studies for updates to systematic reviews can be carried out more effectively by using information about the included/excluded studies from the original review to produce improved Boolean queries. These updated queries reduce the overall number of documents retrieved while also increasing the number of relevant documents identified, thereby representing a considerable reduction in effort required by systematic reviewers.

Keywords: Systematic reviews ; screening ; lexical statistics ; query reformulation ; Systematic reviews updates.

INTRODUCTION

Systematic reviews identify, assess and synthesise the evidence available to answer complex research questions. They are essential in healthcare where the volume of evidence in scientific research publications is vast and cannot feasibly be identified or analysed by individual clinicians or decision makers. However, the process of creating a systematic review is time-consuming and expensive, with a single review often requiring up to a year's effort by expert reviewers [1, 2] and costing up to quarter of a million US dollars [3]. The pace of scientific publication in medicine and related fields also means that evidence bases are constantly changing and review conclusions can quickly become out of date [4]. In fact, it has been estimated that 7% of systematic reviews in the medical field are already out of date by the time of publication; and almost a quarter (23%) two years after they have appeared [5]. Reliance on review conclusions based on out of date evidence increases the risk of recommendations for practice that are sub-optimal and potentially harmful to patients. With over eight thousand reviews being produced per year [6], keeping them up to date represents a formidable challenge. There is therefore a need to develop methods to support the update process of systematic reviews to reduce workload required from researchers and ensure the reviews are consistent with current evidence [7].

The problem of identifying relevant evidence is a significant part of the effort required to produce and update reviews [2]. Evidence is typically identified via a two-stage process. First, a search is carried out over databases of scientific publications to identify any studies of potential relevance to the question being addressed in the review. In the second phase, the studies are screened, normally by two reviewers, and relevance assessment judgements are applied to each document within the retrieved set.

The majority of studies included in a systematic review are normally identified by carrying out searches over databases of scientific publications. These searches generally rely on complex

Boolean queries written by domain experts. The need to ensure that relevant evidence is not missed means that the searches have been developed to optimise recall rather than precision (which is typically as low as 1-2%). Queries often return large numbers of studies and require significant effort during the screening stage to identify the studies that should be included in the review.

The Boolean queries used for systematic reviews are usually formatted using OVID or PubMed search syntax and employ advanced operators in addition to the standard logical operators, AND, OR and NOT. Figure 1 shows an example Boolean query for review CD005025 'Reminder packaging for improving adherence to self-administered long-term medications' [8]. This OVID format query consists of 52 lines (clauses). Lines are numbered so they can be referenced. For example, line 15 combines the results of all previous lines.

```
1. Reminder Systems/
2. exp Patient Compliance/
3. Treatment Refusal/
4. Patient Dropouts/
5. exp Attitude to Health/
6. Patient Satisfaction/
7. (complian* or noncomplian* or non-complian*).tw.
8. (adhere* or nonadhere* or non-adhere*).tw.
9. persist*.tw.
10. (refusal or refuse*).tw.
11. (improve* adj5 (followup or follow up)).tw.
12. (dropout* or drop out*).tw.
13. (treatment adj5 (stop* or abandon*)).tw.
14. (patient* adj5 (attitude* or acceptance or satisfaction)).tw.
15. 1 or 2 or 3 or 4 or 5 or 6 or 7 or 8 or 9 or 10 or 11 or 12
    or 13 or 14
 :
 :

51. 50 and 40
52. limit 51 to yr="2005 - 2010"
```

Figure 1. Example portion of Boolean query for review CD005025 [8].

The process of evidence identification is similar for new reviews and review updates, although for updates it is generally only carried out over studies that have appeared since the search for the original review was conducted. This study proposes an approach to improving the identification of relevant studies for a systematic review update by adapting the Boolean query using information produced during the screening stage of the original review. An iterative

algorithm is proposed to generate query variants by applying a set of transformations. These are assessed using information about which studies were included in the original review and the most effective chosen to update the query.

RELATED WORK

There has been significant interest in the development of techniques to support the identification of studies for inclusion in systematic reviews by applying text mining techniques, for example, [9-16]. The vast majority of this work has focused on the identification of studies for new reviews and only a few papers explored the problem of identifying relevant studies for review updates [9, 17, 18]. Most of those studies evaluated their approaches using simulations of the update process, i.e. by choosing a cut-off point (for example three years before the review was published) and assuming that all studies published before this point were included in the original review while those published afterwards were added during an update [18]. An exception is work that used update information for nine drug therapy systematic reviews [19].

Scells and Zuccon [20] introduced an approach to improving the Boolean queries used for study identification in systematic reviews. The query used in the review was iteratively altered by applying a set of transformations such as replacing logical operators and field restrictions. They found that the modified queries generated by this approach improved upon those used in the original review. The best modified queries were identified using classifiers and learning to rank methods. Their approaches produced queries with higher precision and F-measure scores than the original query but not improved recall (which was not their goal). Their method was used to demonstrate that it was possible to improve the Boolean query used for the original review rather than to develop queries for review updates.

Previous work on the refinement and generation of Boolean queries for other types of professional search, such as prior art search, is also relevant to the work described here. Kim et al. [21] proposed a Boolean query suggestion technique in which a decision tree was learned from pseudo-relevant documents then used to generate queries. Graf et al. [22] developed a method for automatically generating queries for prior art search by analysing the distribution of terms among topic-relevant documents. Harris et al. [23] presented an interactive Boolean search system which helps the user to create a Boolean search query. The interactive system suggests semantically similar search terms to the user.

METHOD

The aim of the work proposed in this paper is to improve the performance of Boolean queries used to retrieve relevant studies for updates of systematic reviews. The approach is based on the query transformation technique described by Scells and Zuccon [20] and extended in two important ways. Firstly, it is applied to the problem of generating queries for review updates and makes use of information about which studies were included/excluded from the original review to guide the query modification. Secondly, it extends the set of query transformations introduced by Scells and Zuccon and demonstrates that the new transformation lead to the generation of more effective queries.

The proposed approach starts with the Boolean query used for the original review. A set of transformed queries are generated by applying a range of transformations (e.g. operator substitution, query expansion and query reduction) to the original query. Each transformed query is then evaluated using the relevance judgements produced for the original review and the best transformation selected. The process is then repeated by applying the transformations to the newly selected query and evaluating the transformed queries produced. The process continues until the best transformed query is no better than the query from the previous iteration

(i.e. the query cannot be improved using this process). The approach is outlined in Algorithm 1.

In this approach, relevance judgements from the original review (i.e. information about included/excluded studies) are used to select the transformed query at each iteration. This information is readily available since the studies returned by the Boolean query are manually screened and include/exclude decisions reported. However, since our aim is to develop improved queries which can be used to support review updating, evaluation is carried out using the studies that have been identified for the review update (See "Experiments") and the approach does not have access to this information.

Algorithm 1 Automatic improvement of Boolean query

Input : Boolean query from original review (q), set of query transformations (T) and original review's relevance judgements (R_{orig})

Output : Updated Boolean query (q^*)

Initialise : $q^* \leftarrow q$

while $true$ **do**
  // Step one: Boolean Query Transformation
  // Generate set of updated queries by applying all possible transformations
  $\hat{Q} \leftarrow \{\}$
  for $t \; in \; T$ **do**
    for $clause \; c \; in \; q^*$ **do**
      if $t \; can \; be \; applied \; to \; c$ **then**
        $\hat{Q} \leftarrow \hat{Q} \cup t(q_c^*)$ // where $t(q_c^*)$ denotes transformation t applied to clause c of q^*
      end
    end
  end
  // Step two: Boolean Query Selection
  // Evaluate each transformed query and select the highest scoring for the next iteration
  for $\hat{q} \; in \; \hat{Q}$ **do**
    $Compute \; f(\hat{q}|R_{orig})$ // Where f is some scoring function based on R_{orig}
  end
  $q' = argmax_{\hat{q} \in \hat{Q}} \, f(\hat{q}|R_{orig})$
  // if performance of the best new query is the same as the base query then the query cannot
  // be improved
  if $f(q'|R_{orig}) \leq f(q^*|R_{orig})$ **then**
    $break$
  end
  $q^* \leftarrow q'$
end
return q^*

The individual steps of the approach are now described in further detail.

Step one: Boolean Query Transformation

The first step of the algorithm applies a set of query transformations to generate new queries from the current one. Three types of transformation are applied.

(a) Operator Substitution. This transformation replaces one query operator with another. For example, disjunction with conjunction (e.g. (blind$ or mask$).ti. → (blind$ and mask$).ti.) or altering a restriction field (e.g. (blind$ or mask$).ti. → (blind$ or mask$).ti,ab.). (In the second example .ti,ab. indicates that both the title and abstract are searched for the terms rather than just the title.)

A set of useful operator substitution transformations were identified during preliminary experiments: .tw.→.ti., .tw.→.ti,ab., .ti,ab.→.ti., .ti,ab.→.tw., .ti.→.tw., .ti.→.ti,ab., .ab.→.ti,ab., .ab.→.ti., .sh.→exp *, and→or and or→and. Some of these transformations were used in previous work [20]: logical operator replacement (and→or and or→and) and four field restrictions (.ti,ab.→.ti., .ti.→.ti,ab., .ab.→.ti,ab. and .ab.→.ti.). The remaining transformations were developed for this study. Additional transformation types were also explored but not found to improve performance, including three field restriction transformations: .af.→.ti,ab., .af.→ .ti. and .af.→ .tw..

(b) Query Expansion. This transformation adds new elements to the query and has not been applied in previous work on query transformation. Terms that tend to occur in included studies are identified using the Log-likelihood statistic and added to the query [24]. The set of studies retrieved for the original review is partitioned into relevant and irrelevant subsets

based on the relevance judgements of the original review and the score for each term computed as:

$$Log\_Liklehood = 2 \times \left(O_{rel} \times \log\frac{O_{rel}}{E_{rel}} + O_{irrel} \times \log\frac{O_{irrel}}{E_{irrel}}\right) \quad (1)$$

where O_{rel} and O_{irrel} are the observed frequency of the term in different subsets of the collection (e.g. relevant and irrelevant studies). E_{rel} and $E_{irrel}$are the term's expected frequencies, calculated as:

$$E_{rel} = N_{rel} \times \frac{O_{rel} + O_{irrel}}{N_{rel} + N_{irrel}} \quad , \quad E_{irrel} = N_{irrel} \times \frac{O_{rel} + O_{irrel}}{N_{rel} + N_{irrel}} \quad (2)$$

where N_{rel} and N_{irrel} represent the size of the sub-corpus (i.e. relevant and irrelevant studies). Terms are assigned high Log-likelihood scores when their observed frequency is (much) higher than the expected frequency within a sub-corpus. Log-likelihood scores are used to identify the five terms that are most closely associated with the relevant studies. These terms are then used to form a set of transformations in which the terms are added to a query clause using the logical or operator and .tw. as the restriction field.[1] For example, the terms packaging, blister, pack, calendar and medication are identified as the top five terms that identify relevant studies for the review CD005025 [8]. Transformations include adding the first term alone (Reminder Systems/ → Reminder Systems/ or packaging.tw.), adding the second term alone (Reminder Systems/ → Reminder Systems/ or blister.tw.), adding the top two terms (Reminder Systems/ → Reminder Systems/ or packaging.tw. or blister.tw.), adding the top three terms (Reminder Systems/ → Reminder Systems/ or packaging.tw. or blister.tw. or pack.tw.) and so on.

[1] The full list of transformations that can be added to each clause is add 1st term, add 2nd term, add 3rd term, add 4th term, add 5th term, add 1st and 2nd terms, add 1st to 3rd terms, add 1st to 4th terms and add all 5 terms.

(c) Query Reduction: The final transformation method simply deletes a clause from the Boolean query. For example, removal of the second clause

1. Reminder Systems/		1. Reminder Systems/
2. exp Patient Compliance/	→	2. Treatment Refusal/
3. Treatment Refusal/		

The transformed queries produced during each iteration differ from the query selected during the previous iteration by a single clause. A total of 21 transformation types are used, leading to up to 21×*c* transformed queries being produced during each iteration (where *c* is the number of clauses in the query selected during the previous iteration). However, this is an upper bound value because not all transformations types applicable to all clauses. For example, the operator substitution .tw.→.ti,ab. cannot be applied to a clause that does not contain the .tw. restriction field.

Step two: Boolean Query Selection

The set of transformed queries generated during step one are evaluated by assessing them against the relevance judgements produced for the original review. Each transformed query is run against MEDLINE and the list of studies it retrieves is returned. The query is then assessed using the following objective function which favours improvements in recall over improvements in precision:

$$f(\hat{q}) = recall(\hat{q}|R_{orig}) \times 100 + precision(\hat{q}|R_{orig}) \quad (3)$$

where $\hat{q}$ is the transformed query and R_{orig} the relevance judgements from the original review. Recall and precision are calculated as follows:

$$recall(\hat{q}|R_{orig}) = \frac{Number\ of\ relevant\ studies\ in\ R_{orig}\ retrived\ by\ \hat{q}}{Total\ number\ of\ relevant\ studies\ in\ R_{orig}} \quad (4)$$

$$precision(\hat{q}|R_{orig}) = \frac{Number\ of\ relevant\ studies\ in\ R_{orig}\ retrived\ by\ \hat{q}}{Total\ number\ of\ studies\ retrived\ by\ \hat{q}} \quad (5)$$

As can be seen from the equation 3, the objective function always assigns a higher score to a query that produces an improvement in recall to one that improves precision. This is due to the nature of the search problem in systematic reviews where high recall is important since the goal is to identify all potentially relevant studies. However, retrieving a large number of irrelevant studies increases the screening effort required by reviewers, and it is therefore beneficial to ensure that the precision of queries is as high as possible. It would be straightforward to adapt the approach proposed here to favour a different balance of recall and precision by using a different objective function.

The transformed query that produces the highest score is then chosen for the next iteration. If there are multiple queries with the same highest score, then one is chosen at random. If there is no difference between performance of the highest scoring query and the query from the previous iteration, then the algorithm stops.

EXPERIMENTS

Experiments were carried out using a publicly available dataset containing 22 reviews from the Cochrane Library of Systematic Reviews [25]. All reviews are intervention type, the most frequently occurring in Cochrane, which evaluate the efficacy of a healthcare intervention for a specific disease.

The dataset includes information about original and updated versions of the reviews including the review title, Boolean query (in OVID format) and set of included PMIDs. For each review the majority of the included PMIDs are identified using the Boolean query but additional studies are often identified using alternative techniques such as hand searching key journals

and examination of the references lists of included studies. The gold standard dataset used in this paper includes all the relevant studies which are available on PubMed regardless of whether they were identified using the Boolean query or by other methods. Note that this means that the query used for the review may not achieve full recall since it may not retrieve all studies included in the review or an update.

PMIDs included after abstract level screening were used for the experiments since the goal of this work is to develop queries that are applied to databases of scientific abstracts, such as PubMed, and only very few studies are included after content level screening for some reviews. Between 1 and 46 studies are included in the updated reviews after abstract level screening. Further information about the dataset can be found in the supplementary material (see Tables S1 and S2).

Approaches are evaluated using the set of studies included in the update as a gold standard list of relevant studies. It is worth noting that this information is not available to the proposed approach which only makes use of information about the studies considered for inclusion in the original review.

Approach 1: Proposed Method

The first approach that was applied is the method proposed in this paper (see METHOD section). Transformed queries are run against MEDLINE using the Entrez package from biopython.org to retrieve studies for the updated version of the review. Publication dates are used to identify studies published since the previous version of the review. To run the queries against MEDLINE, the OVID format Boolean queries are converted to a single-line PubMed format query. This is carried out automatically using a Python script created specifically for this purpose.

Approach 2: Baseline

A baseline approach was implemented which used the Boolean query from the original review to retrieve studies for updated review without any transformation. The original Boolean query is run against MEDLINE and the set of the studies that match the query retrieved. The aim of this approach is to assess performance when the query developed for the original review is re-used for the update, which is common practise within the systematic review community.

Approach 3: Oracle

An oracle approach was also implemented that is similar to the proposed method (see METHOD section) with the exception that performance of the transformed query is assessed using the relevance judgements for the updated review (R_{update}) rather than for the original, i.e. using the objective function:

$$f(\hat{q}) = recall(\hat{q}|R_{update}) \times 100 + precision(\hat{q}|R_{update}) \quad (6)$$

The oracle approach represents an unrealistic scenario since it has access to the relevance judgements for the updated review. However, it provides context for the results of the proposed method by placing an upper bound on the results that are possible by transforming queries for review updates.

Approach 4: Restricted Transformation Types

The final approach is a version of the proposed method (Approach 1) in which only a single transformation type is applied. Three versions of this approach were explored (one for each transformation type): (1) operator substitution, (2) query expansion and (3) query reduction.

RESULTS AND DISCUSSION

Results are shown in Table 1. Recall and precision scores are shown for each method, both for each review individually and averaged across all reviews. Averages are weighted by the number of abstracts in each review to place more weight on reviews where there are larger numbers of abstracts to be screened. The iteration of the algorithm that produced the final query is also shown for the proposed method, oracle and restricted transformation type approaches. This information is not included for the baseline method which is simply the unmodified query from the original review.

Considering average performance, the proposed method produces queries that improve upon those used for the original review (baseline) both in terms of recall and precision. The increase in recall (10.3%) represents a marked increase in the number of relevant studies that are identified for review updates. Although the precision of the queries produced by the proposed method is still low (0.7%) this is more than double the precision obtained using the original queries, thereby halving the set of studies that need to be considered during the expensive manual screening process. Performance of the oracle method demonstrates the challenge of developing high precision queries while also maintaining recall. Results of the restricted transformations approach indicate that using only one type of transformation generally produces queries that are more effective than the original query, but the improvement is much smaller than other approaches, indicating the importance of using different types of query transformation.

More generally, using queries produced by the proposed method leads to increased recall for seven of the 22 reviews and the same recall for another 14. However, recall was reduced for a single review (CD007428), from 0.667 to 0.556. There were nine relevant abstracts for this

review and this change represented a single document being missed. Precision increased for 13 reviews without any reduction in recall.

Table 1. Recall and precision results for each review in the update dataset. Values in boldface denote results improved when comparing with the baseline.

									Restricted Transformations								
	Baseline		Proposed Method			Oracle			Operator Substitution			Query Expansion			Query Reduction		
Review	Recall	Precision	Recall	Precision	iteration	Recall	Precision	iteration	Recall	Precision	iteration	Recall	Precision	iteration	Recall	Precision	iteration
CD000155	0.3333	0.0182	0.3333	**0.0206**	8	0.3333	**0.0667**	11	0.3333	0.0012	13	0.3333	0.0145	3	0.3333	0.0007	4
CD000160	1.0000	0.0005	1.0000	**0.0108**	15	1.0000	**0.0047**	12	1.0000	**0.0008**	4	1.0000	0.0003	2	1.0000	**0.0034**	13
CD000523	1.0000	0.0526	1.0000	**0.0909**	8	1.0000	**1.0000**	5	1.0000	0.0233	8	1.0000	0.0500	2	1.0000	0.0238	8
CD001298	0.0000	0.0000	**0.5882**	**0.0010**	17	**0.5882**	**0.0154**	2	**0.5882**	**0.0003**	13	**0.5882**	**0.0006**	3	**0.2353**	**0.0054**	20
CD001552	1.0000	0.0021	1.0000	**0.0043**	12	1.0000	**0.0444**	10	0.5000	**0.0152**	11	1.0000	0.0021	1	1.0000	**0.0039**	11
CD002064	1.0000	0.0833	1.0000	**1.0000**	9	1.0000	**0.5000**	5	1.0000	**0.1429**	6	1.0000	**0.0909**	2	0.0000	0.0000	9
CD004069	0.8889	0.0068	0.8889	0.0068	1	**1.0000**	**0.0089**	5	0.8889	0.0068	1	0.8889	0.0068	1	0.8889	0.0068	1
CD004214	0.0000	0.0000	**0.5000**	**0.0010**	2	**0.5000**	**0.0010**	2	**0.5000**	**0.0004**	3	**0.5000**	**0.0010**	2	0.0000	0.0000	1
CD004241	0.6000	0.0116	0.6000	0.0022	20	0.6000	**0.1765**	6	0.6000	0.0005	15	0.6000	0.0052	4	0.6000	0.0002	12
CD004479	0.7500	0.0189	0.7500	0.0013	2	0.7500	**0.0211**	2	0.7500	0.0149	3	0.7500	0.0001	2	0.7500	0.0013	2
CD005025	0.4130	0.0139	**0.6304**	0.0017	18	**0.7391**	0.0008	12	0.3913	0.0018	9	**0.5652**	0.0014	4	0.2609	0.0028	16
CD005055	0.6667	0.0033	0.6667	**0.0102**	7	**1.0000**	**0.0083**	6	0.6667	**0.0114**	5	0.6667	0.0016	2	**1.0000**	0.0001	4
CD005083	0.2222	0.0160	**0.5556**	0.0025	3	**0.5556**	**0.0403**	11	0.2222	0.0160	1	**0.5556**	0.0098	2	**0.5556**	0.0025	3
CD005128	0.5556	0.0007	0.5556	**0.0066**	7	0.5556	**0.0066**	6	0.5556	**0.0036**	7	0.5556	0.0007	1	0.5556	**0.0066**	7
CD005426	0.0000	0.0000	0.0000	0.0000	11	0.0000	0.0000	1	0.0000	0.0000	1	0.0000	0.0000	2	0.0000	0.0000	1
CD006839	0.6667	0.0204	0.6667	**0.2000**	9	**1.0000**	**0.3333**	11	0.6667	**0.1000**	7	0.6667	0.0204	1	0.6667	**0.1818**	8
CD006902	0.5000	0.0365	**0.8000**	0.0014	4	**0.8000**	0.0014	4	0.4000	0.0019	9	**0.8000**	0.0014	2	**0.6000**	0.0074	6
CD007020	0.2500	0.0156	0.2500	**0.0192**	3	0.2500	**0.0233**	4	0.2500	0.0083	3	0.2500	0.0147	2	0.2500	0.0052	2
CD007428	0.6667	0.0270	0.5556	**0.0568**	11	**0.7778**	**0.0538**	7	0.6667	**0.0435**	5	0.6667	0.0221	2	**0.7778**	0.0104	4
CD008392	1.0000	0.0014	1.0000	**0.0135**	10	1.0000	**0.0159**	12	1.0000	**0.0065**	7	1.0000	0.0013	2	1.0000	**0.0136**	10
CD010089	0.5000	0.0004	**0.7500**	**0.0021**	5	**0.7500**	**0.0021**	5	0.5000	**0.0008**	7	**0.7500**	**0.0005**	2	**0.7500**	**0.0009**	4
CD010847	0.6667	0.0755	**0.8333**	0.0007	11	**1.0000**	0.0003	8	0.6667	**0.1081**	5	0.6667	0.0755	1	0.6667	0.0154	3
Weighted Average	**0.566**	**0.003**	**0.669**	**0.007**	**9**	**0.691**	**0.012**	**7**	**0.571**	**0.004**	**7**	**0.666**	**0.002**	**2**	**0.641**	**0.005**	**7**

Restricting transformations to a single type leads to a reduction in performance compared to using all types. Operator substitution transformations have the lowest performance with average recall and precision only slightly higher than the baseline. Query expansion transformations are able to achieve almost as high recall as when all three transformation types are combined, but at the expense of precision. Perhaps surprisingly applying only the simple query reduction transformations is more effective than applying operator substitution transformations, leading to improvements in both precision and recall. However, recall drops for more reviews when only a single transformation type is used compared with all types: two for query reduction and three for operator substitution.

Figures 2 and 3 show the weighted average recall and precision scores for each iteration. The figures show the maximum number of iterations applied by each method (e.g. 12 for the oracle), although note that the number of iterations applied to an individual review may be lower (e.g. see Table 1). Overall, improvements in recall (compared with the baseline) appear to be generated during the first iteration while subsequent iterations help to improve precision. The effect is particularly pronounced for the oracle but can still be observed for other methods.

Table 2 shows an analysis of the transformation types used by the various approaches. The table indicates the number of times each transformation was selected to generate the modified query. The transformation type applied most frequently by the proposed method and oracle was remove line (query reduction). The frequent use of this transformation may be explained by the fact removing lines from queries makes them less restrictive and the objective function used to score queries prefers ones that maximise recall (i.e. are less restrictive). On the other hand, the transformation types used most frequently by the operator substitution approach were or→and, .tw.→.ti. and .ti,ab.→.ti.. All of these transformations lead to more restrictive queries

thereby increasing the possibility of missing relevant studies. This is reflected by the low recall achieved using this transformation type (see Table 1).

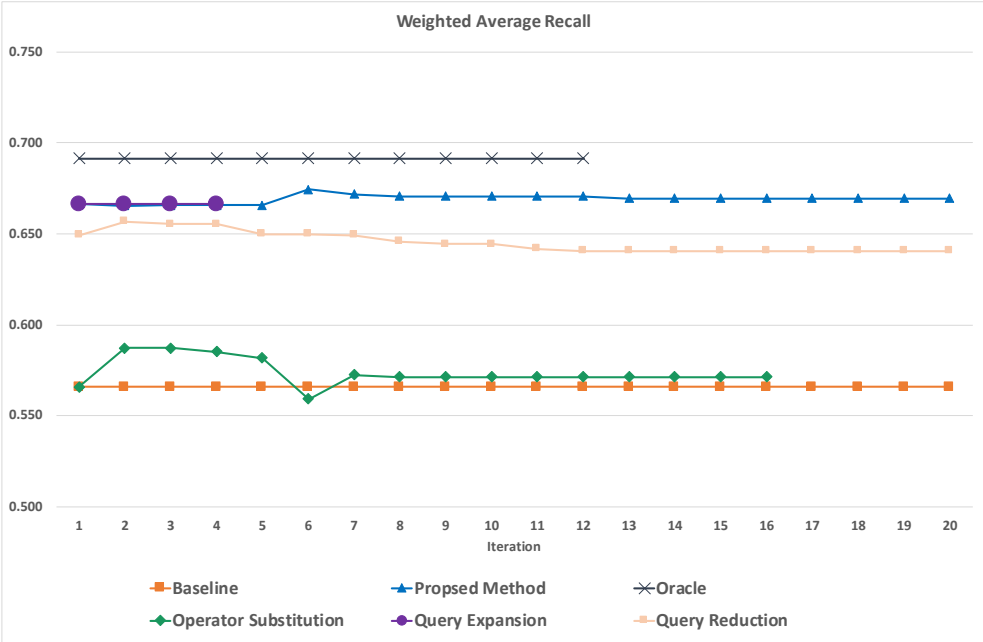

Figure 2. Weighted average recall scores for the various approaches for each iteration. Baseline approach included for comparison.

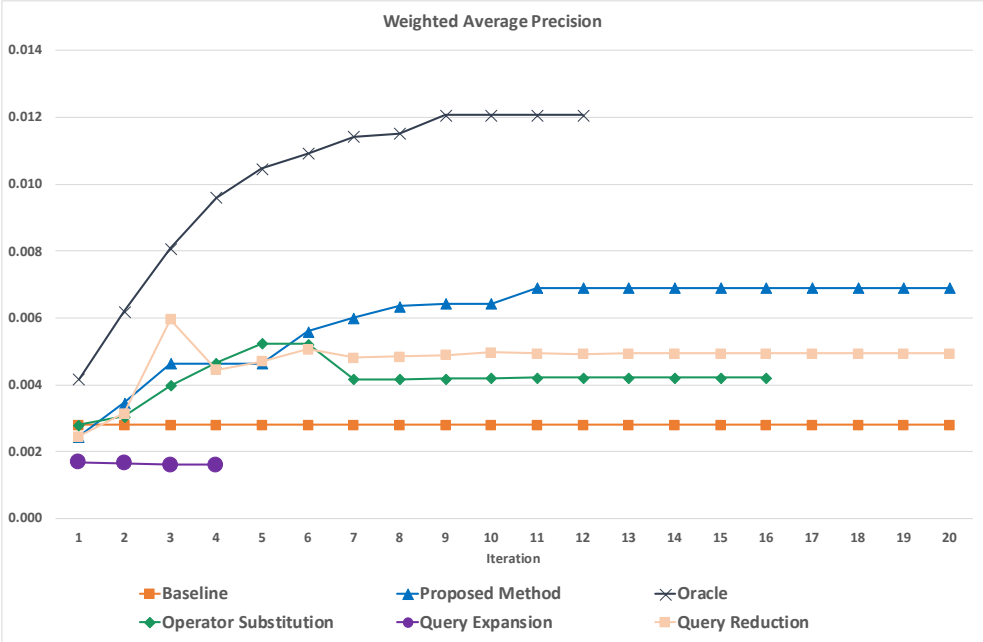

Figure 3. Weighted average precision scores for the various approaches for each iteration. Baseline approach included for comparison.

Table 2. Analysis of transformation types used in the proposed method, oracle and restricted transformations approaches. The numbers represent how many times each transformation has been used through all iterations.

				Restricted Transformations		
Transformation Category	Transformation Type	Proposed Method	Oracle	Operator Substitution	Query Expansion	Query Reduction
Operator Substitution	.tw. → .ti.	6	8	42	-	-
	.tw. → .ti,ab.	2	2	1	-	-
	.ab. → .ti.	0	0	0	-	-
	.ab. → .ti,ab.	0	0	0	-	-
	.ti,ab. → .ti.	16	19	36	-	-
	.ti,ab. → .tw.	0	1	0	-	-
	.ti. → .tw.	0	0	3	-	-
	.ti. → .ti,ab.	0	0	1	-	-
	and → or	2	1	11	-	-
	or → and	1	2	46	-	-
	.sh. → exp *	0	1	0	-	-
Query Expansion	1st top term	5	1	-	14	-
	2nd top term	3	1	-	7	-
	3rd top term	5	5	-	6	-
	4th top term	3	0	-	10	-
	5th top term	2	3	-	1	-
	1st and 2nd top terms	1	0	-	0	-
	1st to 3rd top terms	0	0	-	2	-
	1st to 4th top terms	0	0	-	0	-
	1st to 5th top terms	0	0	-	0	-
Query Reduction	remove line	146	102	-	-	146
Total		192	146	140	40	146

The original Boolean query is returned when the approach is unable to identify a transformation that improves performance. This happens for one review when the proposed method and oracle approaches are used, three for the operator substitution and query reduction transformation types and five for the query expansion transformation type.

Figure 4 shows an example of a baseline Boolean query used for an original review and the transformed query produced by the proposed method (Approach 1). For this review, the algorithm ran for nine iterations with two types selected: operator substitution (use .ti. restriction for clauses 4, 8 and 16) and query reduction (removal of clauses 1, 2, 3, 5 and 7). The transformed query improved precision by 92% without any reduction in recall.

Original query	Transformed query
1 randomized controlled trial.pt. 2 controlled clinical trial.pt. 3 randomized.ab. 4 placebo.ab. 5 drug therapy.fs. 6 randomly.ab. 7 trial.ab. 8 groups.ab. 9 or/1-8 10 exp animals/ not humans.sh. 11 9 not 10 12 exp Motor Neuron Disease/ 13 (moto\$1 neuron\$1 disease\$1 or moto?neuron\$1 disease\$1).mp. 14 ((Lou Gehrig\$1 adj5 syndrome\$1) or Lou Gehrig\$1) adj5 disease).mp. 15 Charcot disease.tw. 16 amyotrophic lateral sclerosis.tw. 17 or/12-16 18 Insulin-Like Growth Factor I/ 19 (rhIGF-1 or rhigf or rhigf-1 or insulin-like).mp. 20 11 and 17 and 19	~~1 randomized controlled trial.pt.~~ ~~2 controlled clinical trial.pt.~~ ~~3 randomized.ab.~~ 4 placebo.ti. ~~5 drug therapy.fs.~~ 6 randomly.ab. ~~7 trial.ab.~~ 8 groups.ti. 9 or/1-8 10 exp animals/ not humans.sh. 11 9 not 10 12 exp Motor Neuron Disease/ 13 (moto\$1 neuron\$1 disease\$1 or moto?neuron\$1 disease\$1).mp. 14 ((Lou Gehrig\$1 adj5 syndrome\$1) or Lou Gehrig\$1) adj5 disease).mp. 15 Charcot disease.tw. 16 amyotrophic lateral sclerosis.ti. 17 or/12-16 18 Insulin-Like Growth Factor I/ 19 (rhIGF-1 or rhigf or rhigf-1 or insulin-like).mp. 20 11 and 17 and 19

Figure 4. Example of the original Boolean query (on the left) and the transformed Boolean query after nine iterations by our proposed model (on the right) with highlighted lines represent the clauses transformed by the model (review CD002064 [26]).

In summary, the results of this study indicate that Boolean query transformations can improve the retrieval performance for the review update in term of recall and precision. The proposed model can produce queries that retrieve more relevant studies and reduce the workload required by researchers by half.

CONCLUSION

This study proposed a method to automatically refine Boolean queries for the study selection stage of systematic review updates. The proposed approach generates a set of transformed queries using three methods: operator substitution, query expansion and query reduction. The best query is then selected using an objective function that considers both recall and precision. The method improves the original query both in terms of recall and precision. It produces queries that are able to identify relevant studies that would not be retrieved using the query from the original review.

Results demonstrated that information available from the original review, particularly the relevance judgements, can be used to produce queries that are more effective than the ones used for the original review. The method proposed here has the potential to assist researchers

conducting updates of systematic reviews by supporting them to produce queries that both identify more relevant studies and reduce the number of studies that need to be screened, thereby reducing the workload required to ensure that reviews remain up to date.

The experiment described here was carried out on one type of systematic review (i.e. intervention reviews) since suitable datasets are not available for other review types. In future it would be interesting to explore performance of the approaches described here to other types of review. Other areas of potential interest include the exploration of additional transformation types (e.g. based on word embeddings or UMLS) and alternative objective functions.

FUNDING STATEMENT

Amal Alharbi was funded by the Royal Embassy of Saudi Arabia, Cultural Bureau in London.

COMPETING INTERESTS STATEMENT

The authors have no competing interests to declare.

CONTRIBUTORSHIP STATEMENT

Amal Alharbi and Mark Stevenson conceived the study. Amal Alharbi carried out the experiments. Both authors contributed to and approved the final manuscript.

REFERENCES

[1] Cohen A, Ambert K, and McDonagh M. A prospective evaluation of an automated classification system to support evidence-based medicine and systematic review. AMIA Annual Symposium Proceedings, 2010: 121–125, 2010.

[2] Karimi S, Pohl S, Scholer F, et. al. Boolean versus ranked querying for biomedical systematic reviews. BMC medical informatics and decision making, 10(1):1–20, 2010.

[3] McGowan J and Sampson M. Systematic reviews need systematic searchers. Journal of the Medical Library Association, 93(1):74, 2005.

[4] Bastian H, Glasziou P, and Chalmers I. Seventy-five trials and eleven systematic reviews a day: How will we ever keep up? PLOS Medicine, 7(9):1–6, 09 2010.

[5] Shojania K, Sampson M, Ansari M, et. al. How quickly do systematic reviews go out of date? a survival analysis. Annals of Internal Medicine, 147:224–234, 2007.

[6] Page M, Shamseer L, Altman D, et. al. Epidemiology and reporting characteristics of systematic reviews of biomedical research: A cross-sectional study. PLOS Medicine, 13(5):1–30, 2016.

[7] Michelson M and Reuter K. The significant cost of systematic reviews and meta-analyses: A call for greater involvement of machine learning to assess the promise of clinical trials. Contemporary clinical trials communications, page 100443, 2019.

[8] Mahtani KR, Heneghan CJ, Glasziou PP, et. al. Reminder packaging for improving adherence to self- administered long-term medications. Cochrane Database of Systematic Reviews, (9), 2011.

[9] Cohan A. Optimizing feature representation for automated systematic review work prioritization. AMIA Annual Symposium Proceedings, (1):121–125, 2008.

[10] Martinez D, Karimi S, Cavedon L, et. al. Facilitating biomedical systematic reviews using ranked text retrieval and classification. Proceedings of the 13th Australasian Document Computing Symposium, (1):53–60, 2008.

[11] Cohen A, William R Hersh, Kim Peterson, and Po-Yin Yen. Reducing workload in systematic review preparation using automated citation classification. Journal of the American Medical Informatics Association, 13(2):206–19, 2006.

[12] Kilicoglu H, Demner-Fushman D, Rindflesch TC, et. al. Towards automatic recognition of scientifically rigorous clinical research evidence. AMIA Annual Symposium Proceedings, 16:25–31, 2009.

[13] Wallace BC, Trikalinos TA, Lau J, et. al. Semi- automated screening of biomedical citations for systematic reviews. BMC Bioinformatics, 11(1):55, 2010.

[14] Kim S and Choi J. An SVM-based high-quality article classifier for systematic reviews. Journal of Biomedical Informatics, 47:153–159, 2014.

[15] Miwa M, Thomas J, O'Mara-Eves A, and Ananiadou S. Reducing systematic review workload through certainty-based screening. Journal of Biomedical Informatics, 51:242–253, 2014.

[16] Lagopoulos A, Anagnostou A, Minas A, et. al. Learning-to- rank and relevance feedback for literature appraisal in empirical medicine. In Proceedings of the International Conference of the Cross-Language Evaluation Forum for European Languages, pages 52–63, Avignon, France, 2018.

[17] Wallace BC, Small K, Brodley CE, et. al. Toward modernizing the systematic review pipeline in genetics: efficient updating via data mining. Genetics in medicine, 14(7):663–669, 2012.

[18] Khabsa M, Elmagarmid A, Ilyas I, et. al. Learning to identify relevant studies for systematic reviews using random forest and external information. Machine Learning, 102 (3):465–482, 2016.

[19] Cohen A, Ambert K, and McDonagh M. Studying the potential impact of automated document classification on scheduling a systematic review update. BMC Medical Informatics and Decision Making, 12(1): 33, 2012.

[20] Scells H and Zuccon G. Generating better queries for systematic reviews. In The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 475–484, 2018.

[21] Kim Y, Seo J, and Croft W. Automatic Boolean query suggestion for professional search. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information - SIGIR '11, pages 825–834, 2011.

[22] Graf E, Azzopardi L, and Rijsbergen K. Automatically generating queries for prior art search. In Proceedings of the Workshop of the Cross-Language Evaluation Forum for European Languages, volume 6241, pages 480–490, Berlin, Heidelberg, 2010.

[23] Harris G, Panangadan A, and Prasanna V. Interactive Query Refinement for Boolean Search. In Proceeding SemADoc workshop, Fort Collins, Colorado USA, 2014.

[24] Alharbi A and Stevenson M. Improving ranking for systematic reviews using query adaptation. In Experimental IR Meets Multilinguality, Multimodality, and Interaction CLEF2109, pages 141–148, Lugano, Switzerland, 2019. Springer International Publishing.

[25] Alharbi A and Stevenson M. A dataset of systematic review updates. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19, pages 1257–1260. ACM, 2019.

[26] Beauverd M, Mitchell JD, Wokke JHJ, et. al. Recombinant human insulin-like growth factor I (rhIGF-I) for the treatment of amyotrophic lateral sclerosis/motor neuron disease. Cochrane Database of Systematic Reviews, (11), 2012.