An Interactive Relevance Feedback Interface for Evidence-Based Health Care

Ivania Donoso-Guzmán
Pontificia Universidad Católica de Chile
Santiago, Chile
indonoso@uc.cl

Denis Parra
Pontificia Universidad Católica de Chile
Santiago, Chile
dparras@uc.cl

ABSTRACT

We design, implement and evaluate EpistAid, an interactive relevance feedback system to support physicians towards a more efficient citation screening process for Evidence Based Health Care (EBHC). The system combines a relevance feedback algorithm with an interactive interface inspired by Tinder-like swipe interaction. To evaluate its efficiency and effectiveness in the citation screening process we conducted a user study with real users (senior medicine students) using a large EBHC dataset (Epistemonikos), with around 400,000 documents. We compared two relevance feedback algorithms, Rocchio and BM25-based. The combination of Rocchio relevance feedback with the document visualization yielded the best recall and F-1 scores, which are the most important metrics for EBHC document screening. In terms of cognitive demand and effort, BM25 relevance feedback without visualization was perceived as needing more physical and cognitive effort. EpistAid has the potential of improving the process for answering clinical questions by reducing the time needed to classify documents, as well as promoting user interaction. Our results can inform the development of intelligent user interfaces for screening research articles in the clinical domain and beyond.

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Graphical User Interfaces; H.3.3 Information Search and Retrieval: Information filtering

Author Keywords

Evidence-Based Health Care; Intelligent User Interfaces; Information Filtering; Visualization; Human in the Loop.

INTRODUCTION

Evidence-Based Health Care (EBHC) is a medical practice approach that emphasizes the use of research evidence to justify a medical treatment. Sackett et al. defined it as *"the conscientious, explicit, and judicious use of current best evidence in making decisions about the care of individual patients"* [51]. EBHC has produced a large impact in the practice of medicine, since applying the knowledge gained from large clinical trials to patient care promotes consistency of treatment and optimal outcomes, in contrast to solely relying on anecdotal cases [50].

Clinicians have established a process to practice evidence based health care [28] which can be summarized as follows: (i) first they pose a question, then (ii) seek studies related to it, (iii) select documents which are relevant to the question, (iv) evaluate the selected documents according to a their methodological quality, (v) next they perform a meta-analysis, (vi) to finally answer the initial clinical question. Despite its growing importance in health care, the process of finding articles which answer a medical question is currently very time consuming and can take several months [4, 5, 12, 17, 37]. This situation can be problematic because in practice, many healthcare related decisions must be made quickly [54]. Moreover, with the explosion of scientific knowledge being published, it is difficult for clinicians to stay updated on the latest best medical practices [12, 17]. Another issue in this area is that the cost of not having all the relevant documents for a given question is very high. Missing a few documents could mean that the answer of the medical question can be wrong, and in consequence, clinicians could treat patients with outdated protocols.

In order to solve the aforementioned issues, some researchers have tried to reduce the time spent in this task using automatic machine learning techniques [41]. However, these algorithms still produce several false positives and negatives, and since the cost of not including a relevant document is high, this task is still performed by clinicians and requires further research to be fully automated [14]. Nowadays, there are some systems designed to support clinicians in the process of collecting, organizing, and searching for scientific evidence such as Embase [18], Covidence [1], and *Epistemonikos* [49]. These systems serve mostly as search engines, but they do not provide advanced functionalities to reduce the workload of the article screening process.

In this context, the main objectives of this research are to design, implement and evaluate a more efficient way to find and screen documents for answering a medical question. We propose *EpistAid*, a system which combines well-established relevance feedback algorithms with a Tinder-like user interface designed to find articles relevant to a medical question, using *Epistemonikos*' database as ground truth.

IUI'18, March 7–11, 2018, Tokyo, Japan

ISBN 978-1-4503-4945-1/18/03…$15.00

DOI: https://doi.org/10.1145/3172944.3172953

By identifying our main task as *citation screening to efficiently classify relevant articles for a clinical question*, we aim to answer the following research questions:

RQ1. Can we expect large differences in performance between relevance feedback algorithms? We answer this question with an off-line analysis over a dataset of close to 400,000 documents.

RQ2. Does an interface in combination with an algorithm perform better than the algorithm alone? We design, implement and evaluate with a user study an interactive Tinder-like user interface which allows physicians to find documents related to a clinical question.

RQ3. Can a document corpus visualization improve the task's outcome? We investigate whether a 2D visualization of the corpus related to a clinical question improves the document screening process.

RQ4. Are there other factors that can affect the task's outcome, such as user expertise or familiarity with visualization? We analyze if there are other factors such as reading skills in English or expertise in EBCH that could affect the citation screening performance.

Contributions. We contribute to the area of intelligent user interfaces applied to Evidence Based Health Care (EBHC) by (i) introducing an interactive interface which supports clinicians in classifying documents for EBHC, (ii) presenting an off-line evaluation of algorithms to classify documents using a real EBHC database; (iii) performing a user study with actual medicine students, not only an off-line evaluation, as much previous research has reported. We found that Rocchio combined with a 2D visualization yielded the best performance in terms of recall, and that previous expertise on EBHC and sufficient command in the documents' language has also a significant effect.

THE PROCESS TO ANSWER CLINICAL QUESTIONS

EBHC requires collecting a list of articles which provide the evidence to answer a clinical question such as "Is there a relationship between vaccines with thimerosal and autism?" The two main types of articles used are systematic reviews (SR) and primary studies (PS). PS is an umbrella term that includes any study design, qualitative or quantitative, where data is collected from individuals or groups of people. On the other side, the main objective of a SR is to synthesize primary studies. Collecting articles to answer a clinical question is iterative, since it involves the manual process of curating documents. The process starts by selecting a seed SR, based on a clinical question. If the seed SR has been digitalized, the screening process checks the cited PS (PS_{cited}), and then continues with other SR which cited documents in PS_{cited}. In summary:

The problem. Find a list a papers relevant to a clinical question by: (i) Identifying a seed SR and PS_{cited}, next (ii) removing cited papers PS_{cited} not related to the clinical question, and then (iii) adding new SR and PS strongly related to the clinical question. This process can take several months, especially (iii), since it involves physicians manually searching and screening papers in several databases without clear guidelines for building queries.

Our solution. We call our solution *EpistAid*. We propose a series of methods combined with an interactive user interface with the aim of reducing the effort of document search and screening. Our solution involves investigating relevance feedback algorithms such as Rocchio and BM25, [35], as well as visualizing documents via dimensionality reduction over the text of the articles [13].

RELATED WORK

Citation screening

Several approaches have been proposed to reduce the workload associated with the task of citation screening in EBHC. Automatic classification has been explored by several authors [3, 4, 10, 11, 23, 29, 32, 33, 38, 57, 62]. They have compared known classifiers (Naive Bayes, Support Vector Machines, K-Nearest Neighbors) with different sets of features (title, abstract, and MESH [1] terms). Their results were promising, but there were large differences in performance depending on the dataset.

There is also research in active learning [37, 58, 60, 61]. These studies have used certainty and uncertainty approaches to select what documents should be shown to the user. Like automatic classification, they have used different sets of features and also sampling techniques to improve results. These authors only performed off-line evaluations and used a few medical questions. The datasets in these studies have 3,000 to 10,000 documents, which is unrealistically small, considering that only Epistemonikos has close to 400,000 documents. In the area of information retrieval, relevance feedback was used in [23, 27] to classify documents. Data Visualization was also used to do citation screening. A visual text mining tool was presented in [21, 22], which used both text and the citation network as input.

To the best of our knowledge, there are two software that include tools to help filter documents for a systematic review. Wallace et al. introduced *Abstrack* in [59] a system that uses active learning and a simple interface to classify documents. Recently, *SWIFT-Review* was presented in [26]. This is a software that ranks documents according to a set of documents previously classified. Both systems require users to enter a reduced list of papers to be screened.

A common characteristic among the works surveyed, is that most research on this topic has been done using small datasets. Olorisade et al. [42] analyzed the quality of the research made in the area and states that *"More than half of the studies used a corpus size of below 1,000 documents for their experiments while corpus size for around 80% of the studies was 3,000 or fewer documents"*. Using these small datasets does not seem appropriate in this area because they are not comparable to the size of current medical databases[2].

In our research, we use *Epistemonikos*' database which has nearly 400,000 documents. For this reason, this is the first

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

[2] https://www.nlm.nih.gov/bsd/index_stats_comp.html

Figure 1: *EpistAid* document screening interface. (A) Navigation Bar shows information about the session and provides buttons to control it. (B) Suggested Documents' Bin contains the documents the algorithm deems relevant. (C) Relevant Documents' Bin and (G) Non-Relevant Documents' Bin contains the documents classified as relevant or non-relevant by the user. (E) Documents Visualization area shows documents as figures in a 2D chart. (F) Document Detail shows document meta-data. (D) Actions Timeline shows the actions performed by the user sorted by time.

research in the area made on a real, large and noisy database. With respect to filtering and classification techniques, for our research we chose relevance feedback over active learning, since the former mimics the actions done by users when they search for documents: users look for documents, see whether they are relevant or not, reformulate the search query and so on, until they think they have found all related evidence.

Controllable User interfaces

Ideally, we would like to create an automatic system to solve the problem of finding relevant research articles for answering clinical questions. However, domain experts usually like to have more control than non-experts on systems supported by intelligent algorithms [31, 44]. This is the case for physicians looking for papers to answer a medical question. Research has shown that controllable interfaces increase satisfaction in recommender and search systems, because they increase transparency and trust [8, 25, 30, 45, 24, 56, 2]. This type of interface also increases user engagement and leads to better user experience [44].

Following these previous works, some authors have created interfaces that support information filtering. For instance, di Sciascio et al. [15] proposed a new interface to present search results. In this interface all results can be ranked according to words present in the documents. Users can assign a weight to each word they select and the documents will be re-ranked accordingly. More recently, Peltonen et al. [47] presented an interface to support relevance feedback that has a visualization of topics and keywords. Users can give positive feedback as well as negative feedback. They found that for certain types of difficult information seeking tasks, negative feedback could benefit an exploratory system even when a good deal of positive feedback is available. Beltran et al. [6] presented a new interface with swipe gestures (Tinder-like) that allows users to classify documents in two groups. This intelligent system creates bins in each group that allow users to justify their classification without needing to write. One common feature of these systems is having a human-in-the-loop (HITL) aspect in the process, that is, they require human interaction to achieve the goal. In EBHC users need to find articles in a way they can understand and trust the results. For this reason we propose a controllable and transparent information filtering system designed for the practice of EBHC, inspired by controllable recommender system interfaces.

EPISTAID

EpistAid evolves from an initial design [16], which was introduced but never evaluated. Here we present *EpistAid* in detail focusing on three aspects: (i) User interface, layout and visual components, (ii) Interactions, where we describe our design based on Schneidermann's visual Information-Seeking mantra [53], and (iii) Algorithms, which support the intelligence behind the filtering process.

User interface

EpistAid is implemented as a web application. The user interface was developed using D3.js [9], dragula.js (**`https://bevacqua.github.io/dragula/`**) and Bootstrap [43]. The GUI

layout, shown in Figure 1, has 7 views and is described as follows:

(A) Navigation Bar. The navbar shows the search's title and the control buttons: search for more documents, save and continue later, finish the search and get help. It also shows how much time the user has left for document screening – functionality needed for the user study. Users do not input a query in the search process. They simply start by picking a *systematic review* related to the clinical question to be answered, and *EpistAid* generates the initial and subsequent queries using relevance feedback. More details are presented in the algorithms coming section **Algorithms**.

(B) Suggested Documents' Bin. The system will look for additional documents related to the clinical question and will display them in this bin. Each document is represented by a rectangle. When clicking on the document, its details will appear on **(F)** and its related mark in visualization **(E)** will increase its size. The documents are sorted from left to right according to their score in the *relevance model* of the document set.

(C) Relevant Documents' Bin and (G) Non-Relevant Documents' Bin. These bins have the documents that have already been classified by the user as relevant or non-relevant. Documents inside the bins don't have any particular order and users can simply place documents on them by drag-and-drop from the (B) panel.

(E) Documents Visualization. This area shows the documents as figures in a 2D chart. Its purpose is to provide an overview of the documents that are in any of the bins (suggested, relevant and non-relevant), and to let the user explore the content based on proximity among documents. Since we represent this set of documents as a document-term matrix (DTM) using a vector space model [35], we perform dimensionality reduction over this DTM to represent each document with a low-rank vector of two dimensions. %deIvania: aca hay solo 4 metodos We chose 5 different dimensionality reduction algorithms: Principal Components Analysis (PCA), Linear Discriminant Analysis (LDA), Latent Semantic Analysis (LDA), Multidimensional Scaling (MDS) [19] and the recent t-distributed Stochastic Neighbor Embedding (t-SNE) [34]. Users can choose the type of dimensionality reduction they prefer in order to eventually visualize the documents in the two dimensional (2D) chart.

In the 2D chart, primary studies (PS) are represented by squares (■, ■, ■) and systematic reviews (SR) by circles (●, ● , ●). The color is used to discriminate the current status of the document: relevant (■, ●), non-relevant (■, ●) or unknown (■, ●). We decided to use these colors (teal, orange, and purple) in order to provide a colorblind safe pallete.

For selecting a subset of documents for more detailed exploration, the user can draw a *brush*. The brushing interaction enables the user to navigate through all the documents by subsetting them based on their positions in the 2D projections. The selected documents are highlighted in all bins (relevant, non-relevant, suggested) and in the *Actions Timeline*. It is also possible to zoom in and out on the visualization.

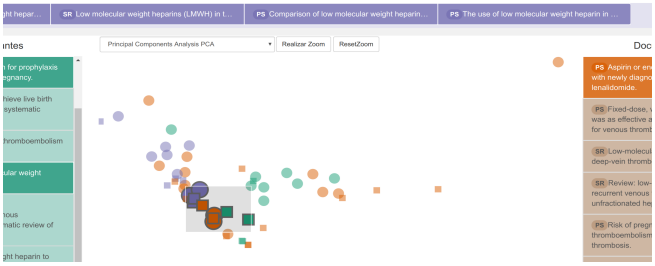


Figure 2: Brush applied in the visualization. It shows how the selected documents are highlighted in the corresponding bins.

(D) Actions Timeline. This panel shows all the documents the user has classified so far, sorted chronologically by time of user labeling. Each item in this view is colored based on its labeled class.

(F) Document Detail. For a document selected in any of the panels, this area shows its meta-data (title, abstract, type of study, publication year and authors). Its goal is to offer the user the option to review the documents in the same way they usually do with a traditional interface.

Justifying the design choices

To justify the visualization we use the framework introduced by Tamara Munzner in [39]. This framework has three layers: *why* is the task being performed, *what* data is shown in the views, and *how* is the visualization idiom constructed in terms of design choices.

Why. The interface allows users to *explore* the set of documents and *identify* documents of certain types (SR or PS), which could be relevant or not. Users have to *analyze* to *produce annotations* on each document. The annotation is the relevance they assign to the document, given the clinical question. In the visualization, users explore without a specific *target* (document) or a pre-defined corpus hierarchy.

What. This visualization is showing document objects which are initially represented in a tabular format, where each document is a row and each column is a word of the corpus. Each cell contains the frequency of that word in the document. After performing dimensionality reduction, each document is represented with a 2D vector, which allows us to display it in the corresponding view. In the visualization, the document's relative positions provide users more information about the document they want to classify. For example, if a document is surrounded by relevant documents, it is very likely that the document is relevant. For this reason, including non-relevant documents was important, since it allowed users to see how close the unknown documents are to the non-relevant ones.

How. We *encoded* all items (documents) using color *hue* for relevance, *shape* for different types of publication and *size* to identify what is being selected. Users can *filter* the data by *selecting* documents using a brush, as seen in Figure 2. Also, users can *change* the position of documents using a dimensionality reduction algorithm and *navigate* using the pan and zoom features. The position channel is given by the result of the dimensionality reduction, so a document will not have

the same position given different dimensionality reduction algorithms.

Interactions

Our interaction design is based on the visual Information-Seeking Mantra: overview first, zoom and filter, then details-on-demand [53].

Overview first. We implement the *overview-first* functionality with the *Documents Visualization* where users can see a summary of the documents that are in any of the bins in the 2D projection resultant from a dimensionality reduction over the term-document matrix. Users can also see both bins: relevant and non-relevant documents already classified.

Zoom and Filter: selecting documents. The *brush* tool described in the previous section allows users to subset documents from the 2D *Documents Visualization*. These documents will be highlighted in all panels, so the user can see their titles, as seen in Figure 2. Users can also zoom in the visualization to look a small quantity of papers and select them by clicking on them or using the brush tool.

Details on demand. When users click a document from any of the bins, *Actions Timeline* or *Documents Visualization* the system provides additional details displayed in *Document Detail*.

The *Tinder*-like inspiration of the interface is implemented in a way that users drag/swipe items to the left if they think they are relevant and to the right if they want to classify as non-relevant.

Algorithms

The application back-end was programmed in Python 3 using Scikit-Learn [46], Pandas [36] and Scipy & Numpy [55]. The code is available at `https://github.com/indonoso/EpisteAid`, a public repository. The documents were represented using a bag-of-words model [35], where the term weights used were based on Term-Frequency.

The main aspect of our "human-in-the-loop" algorithmic procedure starts with modeling the medical question as a *query*, which is updated iteratively when users provide feedback of relevant and non-relevant documents with respect to the clinical question being answered. We tested two algorithms: Rocchio [52], that computes a new query by weighing the words in relevant and non-relevant documents, and BM25, which scores each document based on the frequency of its words in the document and query, as well as based on some global parameters [35].

For both models we define a query q_i as a vector of words, as $\overrightarrow{q_i} = \{w_1, w_2, \ldots, w_n\}$ where n is the number of words in the corpus and w_j is the frequency of the word j. In our system the initial query q_0 is made from the words in the title and abstract from an initial document which represents the source for searching documents relevant to the clinical question. Since this document is a chosen systematic review, we call it *Seed SR*.

For replicability, we disclose the parameters of our algorithms. In Rocchio relevance feedback, parameters were $\alpha = \beta = \delta = 0.25$ and $\gamma = 0.5$. For BM25, parameters used were $k_1 = 1.7$, $k_2 = 1.2$ and $i = 25$.

Type of publication	Articles in database
Primary Study (PS)	277, 967
Systematic Review (SR)	73,040
Overview	1, 229
Structured Summary of PS	1,351
Structured Summary of SR	37,779

Table 1: Statistics of Epistemonikos' document database.

Epistemonikos Documents Dataset

Epistemonikos is a collaborative database which stores research articles that provide the best evidence according to the EBHC principles [49]. Since the evidence comes from scientific literature, this information is collected from specialized online sites such as PubMed and Cochrane, among other 24 sources of scientific information [20]. In this research, we used a dump of their database by December 2016. The database contains around 390,000 documents of five types. Table 1 shows the number of items per publication type. For simplification, anything which is not a PS (excepting for the structured summary of PS), is considered a Systematic Review (SR).

In the Epistemonikos dataset, clinical questions and documents are related through *evidence matrices*, the basic information unit in *Epistemonikos*. Although it is basically a list of research papers, it is called a matrix because it is shown to users in a matrix format, where the rows are Systematic Reviews (SR) and the columns are Primary Studies (PS) cited within those SR. According to data collected by Epistemonikos between 2010 and 2016, physicians require 2-6 months to get from an initial matrix version M_0 to a final revised M_f. They remove non-related articles from the list and then manually search for other articles until convergence. In our experiments, an Epistemonikos' evidence matrix corresponds to a medical question with its relevant documents. All documents that are not listed in the matrix are considered non relevant. The content of the seed systematic review (Seed SR) is used to form the initial query q_0. Epistemonikos currently has over 1,500 evidence matrices in their database, created using Epistemonikos' current document screening process.

Text processing. Each document was represented by the words in its title and abstract. Tokenization was performed by splitting the text by non alphanumeric characters (spaces, tabs, etc.). We then removed all English stopwords in the NLTK package [7]. Many articles used acronyms of medical concepts with less than three characters, and the same acronym is used in different medical domains with different meanings. For this reason we removed any word that had less than three characters. All words were stemmed using *Porter Stemmer* from the same package. This process resulted in a final corpus of $158,457$ words. We removed all documents that did not have tokens of the corpus, resulting in $391,364$ documents.

	Question	Specialty	Context	Relevant docs.
1	Echinacea for common cold	General Medicine	Echineacea is a flower. There are some web sites that say that it is useful for common cold.	54
2	Safety of Low-Molecular-Weight heparin during pregnancy	Internal Medicine	Low-molecular-weight heparin is anticoagulant medication. This question inquires whether it is safe to use during pregnancy, because of the possible risk to the baby.	112

Table 2: Detail of medical questions used in the user study, with their respective number of relevant documents.

EVALUATION METHODOLOGY

We conducted two types of evaluations. The first one, an offline evaluation to compare the performance of both relevance feedback algorithms. The second one, a user study to investigate whether differences in algorithms would be equally reflected in a user interface, or could be moderated by the effect of visualization and interaction.

Offline Evaluation Protocol

We split the document dataset in train (70%) and test (30%) randomly. The train set was used to tune the parameters of each relevance feedback algorithm (Rocchio and BM25), while the test was used to double check for potential problems in the models, such as overfitting. The experiment consisted of a simulation of the relevance feedback process. We selected $1,067$ clinical questions for which we knew the respective seed SR as well as all the relevant documents. For each clinical question, using the seed SR we issued an initial query q_0, using the words present on its title and abstract, and for the documents returned by the algorithms we assumed having a *simulated expert* able to exactly tell relevant from non-relevant documents. The relevance feedback algorithm then would reissue queries $q_1, q_2, ...$ based on the feedback of the simulated expert, so for each query attempt we calculated recall and AP (average precision), two well known metrics in information retrieval [35]. In this way, we could see how much to expect on average for the physicians in the user study, and we could also have a better idea of the performance of Rocchio and BM25 before conducting the user study.

User Study Protocol

We conducted a user study to compare performance of users considering four conditions, based on the relevance feedback algorithm (Rocchio vs. BM25) and on the interface (with and without 2D visualization of documents, i.e., box E in Figure 1). We designed a 2x2 mixed-subjects experiment, where each user had to screen documents in two out of four conditions (For instance, BM25 with visualization and then Rocchio without visualization). The conditions and their sequence were counterbalanced in order to have similar number of users on each one. The user study sequence for every user was:

1. Answer a pre-study survey with questions related to their previous knowledge on the topics (see Table 2), their experience with EBHC, with reading research in English language, and with data visualizations.
2. In the assigned condition, find documents related to one medical question (shown in Table 2). They could obtain documents to label by clicking on the button "search for more documents". In each iteration, users had to provide *relevance feedback* to at least 5 documents to be able to search for more documents.
3. Answer a post-study survey with questions related their satisfaction with the system. Moreover, we measured cognitive effort using the NASA-TLX survey.
4. repeat points 2. and 3. with another assigned *interface-algorithm* condition.

To evaluate the algorithm we paid special attention to recall [48] because in EBCH, having all the relevant documents for a research question is more important that having high precision [41]. However, since we could not afford the participation of physicians in our study for more than two weeks, we made a compromise by studying how much recall they could get by using *EpistAid* for 30 minutes over a period of 10 days.

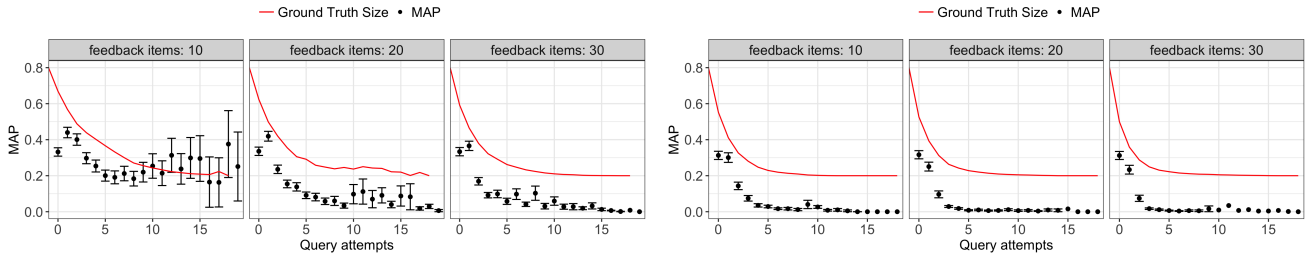
RESULTS

Offline Evaluation Results

Both models have a similar behavior on how they respond to parameters, but overall Rocchio performs clearly better than BM25. Rocchio shows close to 30% better recall than BM25 under different values of the parameter *feedback items* (for how many items the simulated expert provides feedback). Both algorithms show a similar behavior result in terms of Mean Average Precision.

Recall. Figure 4 shows the results of accumulated recall for Rocchio and BM25, respectively. Results were averaged over the clinical questions in the test set. We see a similar trend in both algorithms, where increasing the number of items for which feedback is given in each query attempt (*feedback items:* 10, 20, 30), improves significantly the maximum recall obtained. Other common pattern between these algorithms is that the top accumulated recall is reached after 5-6 query attempts. After that, no significant improvement in recall is observed. With respect to differences, Rocchio clearly outperforms BM25. Based on this offline evaluation, having an expert user labeling the documents we can expect recall values close to 0.4 with Rocchio, while only close to 0.3 in the case of BM25. Considering these results, and since in this area recall is more important than precision [41], we decided to show 30 documents for obtaining user feedback in the user study.

Mean Average Precision@k. Results of Mean Average Precision are shown in Figures 3 (a) for Rocchio and (b) for BM25. We evaluated Average Precision (AP) on each query attempt, and by averaging over all clinical questions, we obtained Mean AP (MAP) at different values of the variable *query attempts* (x-axis in the aforementioned figures). It is important to notice



(a) Rocchio (b) BM25

Figure 3: Offline evaluation. MAP@k for (a) Rocchio and (b) BM25 where *feedback items* is the amount of documents with relevance feedback in each query attempt. The red line shows the percentage of relevant documents left in the ground truth after each query attempt.

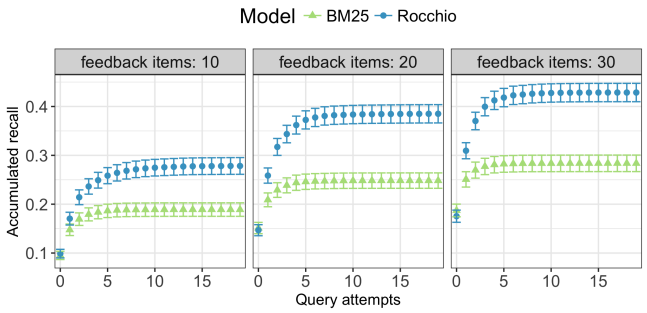


Figure 4: Offline evaluation. Accumulated recall for BM25 and Rocchio, where *feedback items* is the amount of documents the simulation gave feedback in each iteration.

that after each query, when relevant items are identified, we decrease the size of the ground truth. This effect is presented as a decreasing red curve in each plot of Figure 3. In this way we represent the fact that when a relevant document is already found, the probability of finding other relevant documents in subsequent queries diminishes, and is likely to obtain smaller values of MAP.

We observe that MAP yields their maximum values (0.35-0.42) between the first two queries, and then it falls significantly after the second query attempt, probably indicating that after the second query it becomes significantly more difficult to find relevant documents. This observation is supported by the quick drop of the ground truth size (the red line in Figure 3) especially with number of *feedback items* = 30, which indicates that within the first five queries the users find most of the relevant documents they could potentially find by these relevance feedback methods.

User Study Results

Demographics. The study had a total of $N = 22$ subjects, 19 of them were students in their latest year of the medicine program at PUC Chile, and the other 3 user subjects were professors. In terms of experience, 19 of them had never written a systematic review. Moreover, with the exception of one subject, all subjects had created at least one *evidence matrix* (list of papers which answer a clinical question), and

Model / Interface	**Time (in secs.)**	**Docs. class. p/query**	**Docs. class. p/session**	**Nbr. of queries**
BM25 / Non-Vis	1443.9 ± 107.2	21.2 ± 1.3	107.9 ± 17.7	5.1 ± 1.2
BM25 / Vis	1280.1 ± 117.2	20.4 ± 1.5	100.4 ± 16.7	4.9 ± 0.7
Rocchio / Non-Vis	1115.7 ± 80.3	19.8 ± 1.6	88.1 ± 11.9	4.5 ± 0.8
Rocchio / Vis	1267.7 ± 87.7	26.3 ± 0.9	116.3 ± 22.6	4.4 ± 0.9

Table 3: Interaction statistics in each condition studied (mean $\pm$ SE). *Docs. class.* stands for *Documents Classified*.

			Seen documents			Ground Truth
Model	**Interface**	**N**	**Recall**	**Precision**	**F-1 score**	**Recall**
BM25	Non-Vis	12	$.66 \pm .08$	$.52 \pm .06$	$.58 \pm .07$	$.20 \pm .04$
BM25	Vis	11	$.71 \pm .06$	$.64 \pm .04$	$.64 \pm .03$	$.18 \pm .02$
Rocchio	Non-Vis	11	$.65 \pm .08$	$.73 \pm .02$	$.65 \pm .05$	$.21 \pm .04$
Rocchio	Vis	12	$.77 \pm .06$	$.67 \pm .01$	$.70 \pm .03$	$.23 \pm .03$

Table 4: Mean ($\pm$ SE) recall, precision and F-1 score (per user) considering only documents seen by users in the session, and recall considering all documents in the ground truth.

16 had created at least two. Finally, 86% reported being able to read in English language without problems.

Engagement and interaction statistics. In order to measure user interaction and engagement between the four conditions, we compared four metrics which results are displayed in Table 3: time spent on the interface (Time), average number of documents classified per query (Docs. class. p/query), average number of documents classified per session (Docs. class. p/session), and average number of queries issued per session (Nbr. of queries).

We observe that in terms of time, users spent more on the conditions with BM25 ($M = 1443.91$ and $M = 1280.1$) than on the conditions with Rocchio ($M = 1115.7$ and $M = 1267.67$). In terms of queries issued, people also issued more queries on average in the BM25 conditions ($M = 5.1$ and $M = 4.9$) compared to Rocchio ($M = 4.5$ and $M = 4.4$). However, in terms of number of papers classified, list-wise and session-wise, people were more productive in the Rocchio condition with visualization ($M = 116.3$ session-wise and $M = 26.33$ list-wise). The condition with the fewest interactions was Rocchio without 2D document visualization ($M = 88.1$ session-wise and $M = 19.8$ list-wise). Another interesting metric is that in

Question	Model	Interface	M	SE	SD
The suggested documents were relevant	BM25	Non-vis	3.42	0.34	1.16
	BM25	Vis	3.60	0.34	1.07
	Rocchio	Non-vis	4.20	0.13	0.42
	Rocchio	Vis	3.25	0.35	1.22
The suggested documents were diverse	BM25	Non-vis	4.25	0.13	0.45
	BM25	Vis	4.00	0.45	1.41
	Rocchio	Non-vis	4.60	0.22	0.70
	Rocchio	Vis	4.33	0.14	0.49
I understood why the documents were recommended	BM25	Non-vis	3.50	0.36	1.24
	BM25	Vis	3.80	0.42	1.32
	Rocchio	Non-vis	4.40	0.22	0.70
	Rocchio	Vis	3.50	0.31	1.09
The system didn't miss any relevant documents	BM25	Non-vis	2.67	0.31	1.07
	BM25	Vis	3.40	0.45	1.43
	Rocchio	Non-vis	4.00	0.42	1.33
	Rocchio	Vis	3.42	0.34	1.16
I would use the system again	BM25	Non-vis	3.50	0.38	1.31
	BM25	Vis	3.60	0.37	1.17
	Rocchio	Non-vis	4.20	0.20	0.63
	Rocchio	Vis	3.75	0.28	0.97
The system was easy to use	BM25	Non-vis	**4.33***	0.26	0.89
	BM25	Vis	3.90	0.38	1.20
	Rocchio	Non-vis	**4.70***	0.15	0.48
	Rocchio	Vis	3.50	0.31	1.09
I believe the system needs a recommender system	BM25	Non-vis	3.67	0.26	0.89
	BM25	Vis	3.30	0.42	1.34
	Rocchio	Non-vis	3.70	0.26	0.82
	Rocchio	Vis	3.92	0.19	0.67
I didn't realize how the time passed	BM25	Non-vis	3.75	0.33	1.14
	BM25	Vis	3.50	0.34	1.08
	Rocchio	Non-vis	3.70	0.33	1.06
	Rocchio	Vis	3.25	0.30	1.06
I would recommend the system to a collegue	BM25	Non-vis	3.33	0.38	1.30
	BM25	Vis	3.40	0.37	1.17
	Rocchio	Non-vis	3.90	0.23	0.74
	Rocchio	Vis	3.42	0.31	1.08

Table 5: Post-session survey. The only significant difference was found in the question *the system was easy to use*

most experiments (91%), users reported they used at least title or abstract to classify the articles.

Performance Metrics. Table 4 presents the results of recall, precision and F-1 score [35] at the end of the session, averaged per user. These metrics are calculated based on the documents presented to the users during the session. We also calculated recall considering the actual items in the ground truth (112 relevant documents for evidence matrix A and 54 relevant documents for evidence matrix B). Considering all the metrics, the best combination of interface and algorithm was the use of 2D document visualization with the Rocchio relevance feedback algorithm, since it has the best F-1 score considering the items seen during the session ($M = 0.7$) and the best recall with respect to the ground truth, the evidence matrices ($M = 0.23$). The interface that seemed to have the worst general performance was BM25 without visualization, specially in terms of precision, and in terms of recall considering the whole ground truth.

Post-Session Survey. Results of the post-session survey are presented in Table 5. We conducted 2x2 between subjects ANOVA over each of the 9 questions and we only found one statistically significant result. This result was over the question *The system was easy to use*. We found no significant interaction effect, $p = .19$, but we found an effect of the interface variable. Conducting a Tukey HSD post-hoc analysis,

	BM25		Rocchio	
Variable	Non-Vis	Vis	Non-Vis	Vis
Effort	48.9 ± 6.8	31.8 ± 6.0	27.6 ± 5.1	31.7 ± 5.1
Frustration	$\mathbf{46.3}^{*} \pm 6.2$	$\mathbf{32.8}^{*} \pm 6.4$	18.0 ± 5.1	27.2 ± 6.4
Mental Demand	47.8 ± 5.2	36.6 ± 9.1	28.1 ± 6.4	41.8 ± 5.3
Performance	55.3 ± 5.5	57.3 ± 9.9	72.5 ± 6.3	63.7 ± 6.9
Physical Demand	32.8 ± 6.3	17.3 ± 7.8	18.4 ± 6.4	31.3 ± 7.9
Temporal Demand	30.3 ± 5.3	25.7 ± 8.3	21.1 ± 4.5	30.8 ± 5.6

Table 6: NASA-TLX results grouped by algorithm and interface. BM25 model was significantly more frustrating.

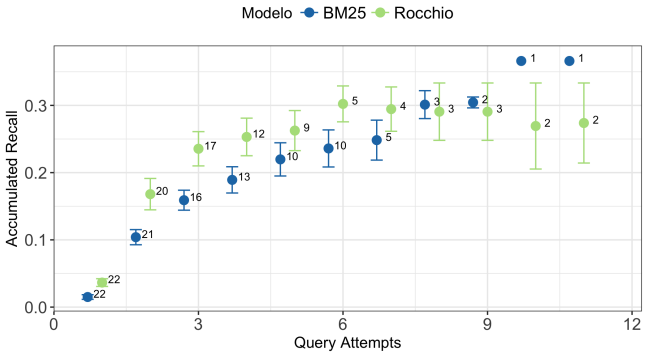


Figure 5: User study. Accumulated recall at different query attemps. Error bar depicts standard error.

we found that interface with no visualization ($M = 4.5$) was perceived as significantly easier than using an interface with visualization ($M = 3.7$), $p = .007$.

Perception of Effort. We used the NASA TLX to measure the perception of effort, results are presented in Table 6. Among the six variables measured, by conducting a 2x2 between subjects ANOVA we found a significant result on *Frustration*. We found no interaction effect between algorithm (BM25, Rocchio) and interface (vis, non-vis), $p = .07$, but we found a significant effect of the algorithm model. In particular, BM25 ($M = 39.6$) produced significantly more frustration than Rocchio ($M = 22.6$), $p = .008$.

DISCUSSION

In the following section, we answer the research questions stated in the introduction.

RQ1. Can we expect large differences in performance between relevance feedback algorithms?

The offline evaluation showed Rocchio clearly outperforming BM25 in both recall and MAP. We observed a similar behavior during the user study, but the difference was not as large as in the offline evaluation. For understanding this behavior, one aspect to consider is how much feedback people needed to provide in order to receive relevant documents. During the user study, Rocchio yields high recall faster than BM25 (Figure 5). After the second query attempt, the mean accumulated recall reached by Rocchio is significantly better than BM25, but the difference decreases as more query attempts are made. Then if the users are persistent, they can reach good results with either

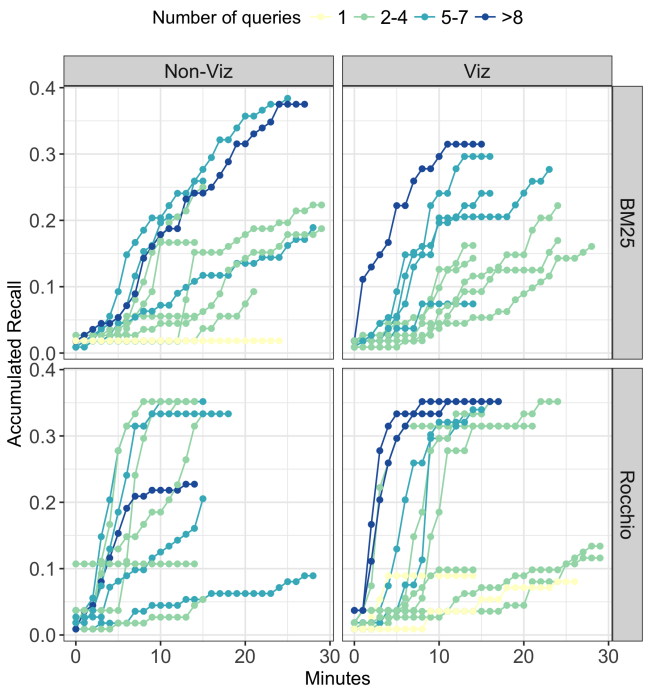


Figure 6: Accumulated recall at different minutes of the session. Each line represents a user. Rocchio achieves higher recall faster than BM25.

method, but if they try no more than 6 queries, they will more likely obtain better results with Rocchio.

Another perspective can be seen by analyzing the recall at different algorithm and interface (with an without visualization), as a function of session time, what we show in Figure 6. The plot indicates that many users under the Rocchio algorithm can reach their top recall within 5 minutes, meanwhile users under BM25 achieve it not before than 10-15 minutes. However, this results also in shorter sessions for Rocchio users, they feel satisfied with their result and stop exploring, while many users under BM25 keep exploring, even reaching recall levels close to 0.4. The results indicate that persistent users can leverage the interface to improve the weak offline results of BM25, compared to Rocchio users.

RQ2. Does an interface in combination with an algorithm perform better than the algorithm alone?

The offline evaluation showed that Rocchio reached more than 30% of the recall of BM25 (Figure 4). The important decrease of this difference during the user study indicates an effect of the interface. To dig deeper into this result, we performed the same offline experiment only considering the two clinical questions in the user study. We see, in Figure 7, that BM25 is clearly outperformed by Rocchio and even more, in question 1 it yields a $recall = 0$ since it never returns a relevant document.

One explanation we have for this result is that in the offline evaluation, the simulated user judges documents simply as relevant or not, without considering levels of relevancy. On the other side, in the user study, the *human in the loop* can infer

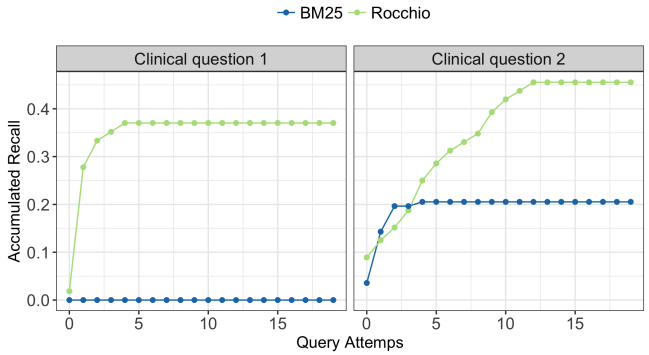


Figure 7: Offline evaluation conducted on the two questions of the user study. Accumulated recall at different query attemps. BM25 does not find relevant documents for question 1. Rocchio performs two times better than BM25 in question 2.

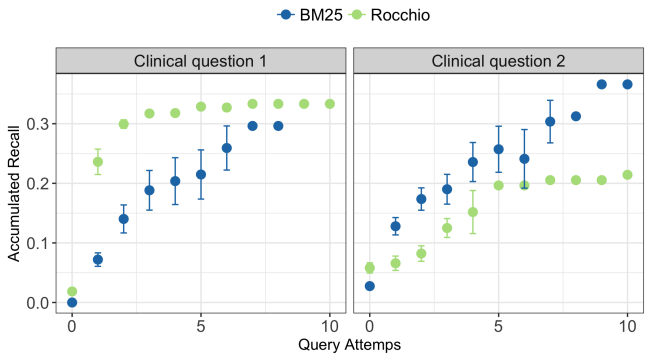


Figure 8: Accumulated recall in the user study, split by clinical question.

degrees or relevance, labeling a "barely relevant document" as relevant in order to diversify the results for the upcoming iteration until an actual relevant document is found. In summary, the use of an interactive interface favors BM25 making the difference with Rocchio smaller in terms of performance metrics. We then split results of the user study according to the clinical questions and we analyzed the actual user performance in Figure 8. Unlike the offline evaluation (Figure 7), for clinical question 1 Rocchio yields similar results, but BM25 users obtained $recall > 0$, and those who issued 6 or more queries can reach up to Rocchio level. In clinical question 2, people using BM25 actually reach better recall then Rocchio, specially users using 7 or more queries.

RQ3. Can a visualization of the documents increase performance or engagement?

Our results shown in the previous section, specially those in Table 3 and Table 4, show a trend towards better recall in visualization conditions when considering the documents seen by users during the session. However, this trend is not the same in terms of precision. Results shown in Table 4 indicate that the top levels of precision are found with Rocchio algorithm in an interface without visualization.

Since the success of the task of document screening depends more on recall than on precision, we focus on this aspect. To

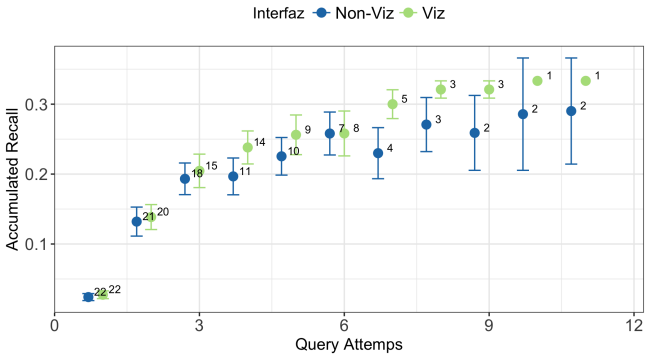


Figure 9: Accumulated recall at different query attempts. Error bar depicts standard error. The number by the circle indicates number of users N.

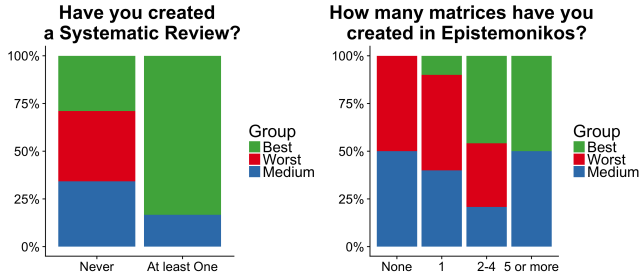


Figure 10: Experience with Evidence Based Health Care.

get a deeper understanding on the effect of visualization in terms of recall, we analyzed the average accumulated recall by query attempt and by interface, shown in Figure 9. We see that in conditions with visualization, after the 6th query attempt, the accumulated recall significantly improves over non-visualization conditions. After the 9th query attempt the difference disappears, but only 3 users reach that number so no statistical tests are valid at that point.

One possible explanation for this is that visualization helps users see relations between more documents holistically, which decreases the chances of missing documents. It happens specially at further query attempts because in each iteration more documents are added to the system, so there is more information to improve the discriminative utility of the visualization. Some comments of users support this explanation: a user said *"The visualization was useful to find similar studies"*, and another user stated *"I used the visualization to estimate how useful a study could be"*. The visualization does not significantly reduce cognitive load (Table 6) but it also promotes engagement. More users stayed longer than 15 minutes when using the visualization (Figure 6).

RQ4. Are there other factors that can affect performance?
To better understand the differences in user performance for the task of document screening, we split users in three groups based on their results on recall: *best*, *middle* and *worst*. These groups were separated by maintaining the same number of experiment sessions in each group. The thresholds were: (i) *worst*: less than 0.12 recall, (ii) *middle*: more than 0.12 and less than 0.30 recall, and (iii) *best*: more than 0.30 recall. There were 14 sessions in groups *worst* and *middle*, and 16 sessions within *best*.

During the pre-study survey, the experience of the users was measured in three topics: general academic experience, experience working in Evidence Based Health Care, as well as experience with data visualizations. In this latest aspect we found no differences among groups. Having the ability to read research in English (as opposed to reading any kind of text in English language) does affect recall. Users who strongly agreed with the statement *"I can read research in English"*, had significantly ($p < .01$) better performance than those who did not.

With respect to experience with EBHC, Figure 10 shows the experience among the aforementioned recall-performance groups. Having worked in the creation of a Systematic Review helped to get better results ($p < .05$). Users that had created two o more evidence matrices had better recall than those who had never created one or had only created one matrix ($M = 0.23$ vs. $M = 0.13$, $p < .01$).

CONCLUSION

In this article we have investigated whether an interactive relevance feedback user interface could help physicians in the process of screening documents to answer a medical question. We have introduced *EpistAid*, our proposed solution, and we have evaluated it with an offline simulation, as well as with a user study considering a large dataset and real users of a EBHC system, medical doctors and medicine students.

We found that the algorithm used in the process is not only relevant for performance metrics, but also for perception of cognitive demand. Rocchio relevance feedback combined with a visualization of documents was found to be better than the other conditions in terms of recall and F-1 score for medical document screening. We also discovered that a good command on reading research in English language was an important factor. This finding might seem negligible, but it supports the current efforts by Epistemonikos on translating articles into different languages. Experience with working in EBHC was also found to be an important variable, which supports the need for training physicians in this type of research activity.

In future work it would be interesting to test other style of algorithms, in particular active and reinforcement learning. The current system and evaluation had the limitation of making physicians work independently, so a system which promotes collaborative work for answering clinical questions could be of great help. Finally, we also think that adding explanations [40] to justify why a document is recommended as relevant is an important idea for research considering that we will continue working with a *human in the loop* paradigm.

Acknowledgments

We acknowledge the help of Gabriel Rada and Daniel Perez from Epistemonikos foundation. Authors were supported by the Chilean research agency Conicyt, Fondecyt grant#11150783.

REFERENCES

1. Clive E. Adams, Stefanie Polzmacher, and Annabelle Wolff. 2013. Systematic reviews: Work that needs to be done and not to be done. *Journal of Evidence-Based Medicine* 6, 4 (2013), 232–235. DOI: http://dx.doi.org/10.1111/jebm.12072
2. Ivana Andjelkovic, Denis Parra, and John O'Donovan. 2016. Moodplay: Interactive Mood-based Music Discovery and Recommendation. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization (UMAP '16)*. ACM, New York, NY, USA, 275–279. DOI:http://dx.doi.org/10.1145/2930238.2930280
3. Tanja Bekhuis and Dina Demner-Fushman. 2010. Towards automating the initial screening phase of a systematic review. *Studies in Health Technology and Informatics* 160, PART 1 (2010), 146–150. DOI: http://dx.doi.org/10.3233/978-1-60750-588-4-146
4. Tanja Bekhuis, Eugene Tseytlin, Kevin J. Mitchell, and Dina Demner-Fushman. 2014. Feature engineering and a proposed decision-support system for systematic reviewers of medical evidence. *PLoS ONE* 9, 1 (2014), 1–10. DOI: http://dx.doi.org/10.1371/journal.pone.0086277
5. Elaine M Beller, Joyce Kee-Hsin Chen, Una Li-Hsiang Wang, and Paul P Glasziou. 2013. Are systematic reviews up-to-date at the time of publication? *Syst Rev* 2 (2013), 36. DOI: http://dx.doi.org/10.1186/2046-4053-2-36
6. Juan Felipe Beltran, Ziqi Huang, Azza Abouzied, and Arnab Nandi. 2017. Don'T Just Swipe Left, Tell Me Why: Enhancing Gesture-based Feedback with Reason Bins. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces (IUI '17)*. ACM, New York, NY, USA, 469–480. DOI: http://dx.doi.org/10.1145/3025171.3025212
7. Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python* (1st ed.). O'Reilly Media, Inc.
8. Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. 2012. TasteWeights: A Visual Interactive Hybrid Recommender System. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, New York, NY, USA, 35–42. DOI: http://dx.doi.org/10.1145/2365952.2365964
9. Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. D3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (Dec. 2011), 2301–2309. DOI: http://dx.doi.org/10.1109/TVCG.2011.185
10. Sungbin Choi, Borim Ryu, Sooyoung Yoo, and Jinwook Choi. 2012. Combining relevancy and methodological quality into a single ranking for evidence-based medicine. *Information Sciences* 214 (2012), 76–90. DOI:http://dx.doi.org/10.1016/j.ins.2012.05.027
11. Aaron M Cohen. 2006. An effective general purpose approach for automated biomedical document classification. *AMIA ... Annual Symposium proceedings / AMIA Symposium. AMIA Symposium* (2006), 161–165.
12. Aaron M Cohen, Clive E Adams, John M Davis, Clement Yu, Philip S Yu, Weiyi Meng, Lorna Duggan, Marian McDonagh, and Neil R Smalheiser. 2010. Evidence-based medicine, the essential role of systematic reviews, and the need for automated text mining tools. *Proceedings of the ACM international conference on Health informatics - IHI '10* (2010), 376. DOI: http://dx.doi.org/10.1145/1882992.1883046
13. Steven P Crain, Ke Zhou, Shuang-Hong Yang, and Hongyuan Zha. 2012. Dimensionality reduction and topic modeling: From latent semantic indexing to latent dirichlet allocation and beyond. In *Mining text data*. Springer, 129–161.
14. V Dhar. 2016. When to trust robots with decisions, and when not to. *Harvard Business Review* (2016).
15. Cecilia di Sciascio, Vedran Sabol, and Eduardo E. Veas. 2016. Rank As You Go: User-Driven Exploration of Search Results. In *Proceedings of the 21st International Conference on Intelligent User Interfaces (IUI '16)*. ACM, New York, NY, USA, 118–129. DOI: http://dx.doi.org/10.1145/2856767.2856797
16. Ivania Donoso-Guzmán. 2017. EpistAid: An Interactive Intelligent System for Evidence-based Health Care. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces Companion (IUI '17 Companion)*. ACM, New York, NY, USA, 177–180. DOI: http://dx.doi.org/10.1145/3030024.3038281
17. Julian H. Elliott, Tari Turner, Ornella Clavisi, James Thomas, Julian P. T. Higgins, Chris Mavergames, and Russell L. Gruen. 2014. Living Systematic Reviews: An Emerging Opportunity to Narrow the Evidence-Practice Gap. *PLoS Medicine* 11, 2 (2014), e1001603. DOI: http://dx.doi.org/10.1371/journal.pmed.1001603
18. Elsevier. 2016. *Embase, Biomedical evidence is essential.* http://store.elsevier.com/embase
19. Daniel Engel, Lars Hüttenberger, and Bernd Hamann. 2012. A survey of dimension reduction methods for high-dimensional data analysis and visualization. In *OASIcs-OpenAccess Series in Informatics*, Vol. 27. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
20. Epistemonikos. 2016. *Epistemonikos database methods.* http://www.epistemonikos.org/en/about_us/methods
21. Katia R. Felizardo, Gabriel F. Andery, Fernando V. Paulovich, Rosane Minghim, and Jose C. Maldonado. 2012. A visual analysis approach to validate the selection review of primary studies in systematic reviews. *Information and Software Technology* 54, 10 (2012), 1079–1091. DOI: http://dx.doi.org/10.1016/j.infsof.2012.04.003
22. Katia Romero Felizardo, Simone R S Souza, and José Carlos Maldonado. 2013. The use of visual text mining to support the study selection activity in systematic literature reviews: A replication study. In *Proceedings - 2013 3rd International Workshop on Replication in Empirical Software Engineering Research, RESER 2013*. DOI: http://dx.doi.org/10.1109/RESER.2013.9
23. J. J. García Adeva, J. M. Pikatza Atxa, M. Ubeda Carrillo, and E. Ansuategi Zengotitabengoa. 2014. Automatic text classification to support systematic reviews in medicine. *Expert Systems with Applications* 41, 4 PART 1 (2014), 1498–1508. DOI: http://dx.doi.org/10.1016/j.eswa.2013.08.047
24. Shuguang Han, Daqing He, Jiepu Jiang, and Zhen Yue. 2013. Supporting Exploratory People Search: A Study of Factor Transparency and User Control. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management (CIKM '13)*. ACM, New York, NY, USA, 449–458. DOI: http://dx.doi.org/10.1145/2505515.2505684
25. Yoshinori Hijikata, Yuki Kai, and Shogo Nishida. 2012. The relation between user intervention and user satisfaction for information recommendation. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*. ACM, 2002–2007.
26. Brian E Howard, Jason Phillips, Kyle Miller, Arpit Tandon, Deepak Mav, Mihir R Shah, Stephanie Holmgren, Katherine E Pelch, Vickie Walker, Andrew A Rooney, Malcolm Macleod, Ruchir R Shah, and Kristina Thayer. 2016. SWIFT-Review: a text-mining workbench for systematic review. *Systematic reviews* 5 (2016), 87. DOI: http://dx.doi.org/10.1186/s13643-016-0263-z
27. Siddhartha Jonnalagadda and Diana Petitti. 2013. A new iterative method to reduce workload in systematic review process. *International journal of computational biology and drug design* 6, 1-2 (2013), 5–17. DOI:http://dx.doi.org/10.1504/IJCBDD.2013.052198
28. Khalid S Khan, Regina Kunz, Jos Kleijnen, and Gerd Antes. 2003. Five steps to conducting a systematic review. *Journal of the Royal Society of Medicine* 96, 3 (2003), 118–121. DOI: http://dx.doi.org/10.1258/jrsm.96.3.118
29. Halil Kilicoglu, Dina Demner-Fushman, Thomas C. Rindflesch, Nancy L. Wilczynski, and R. Brian Haynes. 2009. Towards Automatic Recognition of Scientifically Rigorous Clinical Research Evidence. *Journal of the American Medical Informatics Association* 16, 1 (2009), 25–31. DOI:http://dx.doi.org/10.1197/jamia.M2996
30. Bart P. Knijnenburg, Svetlin Bostandjiev, John O'Donovan, and Alfred Kobsa. 2012. Inspectability and Control in Social Recommenders. In *Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys '12)*. ACM, New York, NY, USA, 43–50. DOI: http://dx.doi.org/10.1145/2365952.2365966

31. Bart P Knijnenburg and Martijn C Willemsen. 2011. Each to His Own : How Different Users Call for Different Interaction Methods in Recommender Systems. *Proceedings of the 5th ACM conference on Recommender systems - RecSys '11* (2011), 141–148. DOI: http://dx.doi.org/10.1145/2043932.2043960

32. Alexandre Kouznetsov and Nathalie Japkowicz. 2010. *Using Classifier Performance Visualization to Improve Collective Ranking Techniques for Biomedical Abstracts Classification*. Springer Berlin Heidelberg, Berlin, Heidelberg, 299–303. DOI: http://dx.doi.org/10.1007/978-3-642-13059-5_33

33. Alexandre Kouznetsov, Stan Matwin, Diana Inkpen, Amir H Razavi, Oana Frunza, Morvarid Sehatkar, Leanne Seaward, and Peter O'Blenis. 2009. *Classifying Biomedical Abstracts Using Committees of Classifiers and Collective Ranking Techniques*. Springer Berlin Heidelberg, Berlin, Heidelberg, 224–228. DOI: http://dx.doi.org/10.1007/978-3-642-01818-3_29

34. Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605.

35. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.

36. Wes McKinney. 2010. Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*, Stéfan van der Walt and Jarrod Millman (Eds.). 51–56.

37. Makoto Miwa, James Thomas, Alison O'Mara-Eves, and Sophia Ananiadou. 2014. Reducing systematic review workload through certainty-based screening. *Journal of Biomedical Informatics* 51 (2014), 242–253. DOI:http://dx.doi.org/10.1016/j.jbi.2014.06.005

38. Yuanhan Mo, Georgios Kontonatsios, and Sophia Ananiadou. 2015. Supporting systematic reviews using LDA-based document representations. *Systematic Reviews* 4, 1 (2015), 172. DOI: http://dx.doi.org/10.1186/s13643-015-0117-0

39. Tamara Munzner and Eamonn Maguire. 2015. *Visualization analysis and design*. CRC Press, Boca Raton, FL. https://cds.cern.ch/record/2001992

40. Ingrid Nunes and Dietmar Jannach. 2017. A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction* 27, 3-5 (2017), 393–444.

41. 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. DOI: http://dx.doi.org/10.1186/2046-4053-4-5

42. Babatunde K. Olorisade, Ed de Quincey, Pearl Brereton, and Peter Andras. 2016. A Critical Analysis of Studies That Address the Use of Text Mining for Citation Screening in Systematic Reviews. In *Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering (EASE '16)*. ACM, New York, NY, USA, Article 14, 11 pages. DOI: http://dx.doi.org/10.1145/2915970.2915982

43. Mark Otto and Jacob Thornton. 2016. *Bootstrap*. http://getbootstrap.com/

44. Denis Parra and Peter Brusilovsky. 2015. User-controllable Personalization. *Int. J. Hum.-Comput. Stud.* 78, C (June 2015), 43–67. DOI:http://dx.doi.org/10.1016/j.ijhcs.2015.01.007

45. Denis Parra, Peter Brusilovsky, and Christoph Trattner. 2014. See What You Want to See: Visual User-Driven Approach for Hybrid Recommendation. In *Proceedings of the 19th international conference on Intelligent User Interfaces*. ACM, 235–240.

46. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

47. Jaakko Peltonen, Jonathan Strahl, and Patrik Floréen. 2017. Negative Relevance Feedback for Exploratory Search with Visual Interactive Intent Modeling. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces (IUI '17)*. ACM, New York, NY, USA, 149–159. DOI:http://dx.doi.org/10.1145/3025171.3025222

48. David Martin Powers. 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. (2011).

49. Gabriel Rada, Daniel Perez, and Daniel Capurro. 2013. Epistemonikos: a free, relational, collaborative, multilingual database of health evidence. *Studies in health technology and informatics* 192 (2013), 486–490. DOI: http://dx.doi.org/10.3233/978-1-61499-289-9-486

50. David L Sackett and William MC Rosenberg. 1995. The need for evidence-based medicine. *Journal of the Royal Society of Medicine* 88, 11 (1995), 620–624.

51. David L Sackett, William MC Rosenberg, JA Muir Gray, R Brian Haynes, and W Scott Richardson. 1996. Evidence based medicine: what it is and what it isn't. *Bmj* 312, 7023 (1996), 71–72.

52. G. Salton. 1971. *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

53. B. Shneiderman. 1996. The eyes have it: a task by data type taxonomy for informatio nvisualizations. *Proceedings 1996 IEEE Symposium on Visual Languages* (1996), 336–343. DOI: http://dx.doi.org/10.1109/VL.1996.545307

54. James Thomas, John McNaught, and Sophia Ananiadou. 2011. Applications of text mining within systematic reviews. *Research Synthesis Methods* 2, 1 (2011), 1–14. DOI: http://dx.doi.org/10.1002/jrsm.27

55. S. van der Walt, S. C. Colbert, and G. Varoquaux. 2011. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science Engineering* 13, 2 (March 2011), 22–30. DOI: http://dx.doi.org/10.1109/MCSE.2011.37

56. Katrien Verbert, Denis Parra, Peter Brusilovsky, and Erik Duval. 2013. Visualizing recommendations to support exploration, transparency and controllability. In *Proceedings of the 2013 international conference on Intelligent user interfaces*. ACM, 351–362.

57. Byron C. Wallace, Kevin Small, Carla E. Brodley, Joseph Lau, Christopher H. Schmid, Lars Bertram, Christina M. Lill, Joshua T. Cohen, and Thomas A. Trikalinos. 2012b. Toward modernizing the systematic review pipeline in genetics: efficient updating via data mining. *Genetics in Medicine* 14, 7 (2012), 663–669. DOI: http://dx.doi.org/10.1038/gim.2012.7

58. Byron C Wallace, Kevin Small, Carla E Brodley, Joseph Lau, and Thomas a Trikalinos. 2010. Modeling Annotation Time to Reduce Workload in Comparative Effectiveness Reviews Categories and Subject Descriptors Active Learning to Mitigate Workload. *Proceedings of the 1st ACM International Health Informatics Symposium. ACM,* (2010), 28–35. DOI:http://dx.doi.org/10.1145/1882992.1882999

59. Byron C. Wallace, Kevin Small, Carla E. Brodley, Joseph Lau, and Thomas A. Trikalinos. 2012a. Deploying an Interactive Machine Learning System in an Evidence-based Practice Center: Abstrackr. In *Proceedings of the 2Nd ACM SIGHIT International Health Informatics Symposium (IHI '12)*. ACM, New York, NY, USA, 819–824. DOI: http://dx.doi.org/10.1145/2110363.2110464

60. BC C Wallace, K Small, CE E Brodley, and TA A Trikalinos. 2011. Who Should Label What? Instance Allocation in Multiple Expert Active Learning. *Sdm* (2011), 176–187. DOI: http://dx.doi.org/10.1137/1.9781611972818.16

61. Byron C Wallace, Thomas a Trikalinos, Joseph Lau, Carla Brodley, and Christopher H Schmid. 2010. Semi-automated screening of biomedical citations for systematic reviews. *BMC bioinformatics* 11, 55 (2010), 55. DOI:http://dx.doi.org/10.1186/1471-2105-11-55

62. Wei Yu, Melinda Clyne, Siobhan M Dolan, Ajay Yesupriya, Anja Wulf, Tiebin Liu, Muin J Khoury, and Marta Gwinn. 2008. GAPscreener: An automatic tool for screening human genetic association literature in PubMed using the support vector machine technique. *BMC Bioinformatics* 9 (2008), 205. DOI: http://dx.doi.org/10.1186/1471-2105-9-205