Interactive search through iterative refinement

Muuo Wambua
University of Toronto
Toronto, Ontario
muuo@cs.toronto.edu

Stefania Raimondo
University of Toronto
Toronto, Ontario
sraimond@cs.toronto.edu

Jennifer Boger
University of Waterloo
Waterloo, Ontario
jboger@uwaterloo.ca

Jan Polgar
Western University
London, Ontario
jpolgar@uwo.ca

Hamidreza Chinaei
University of Toronto
Toronto, Ontario
chinaei@cs.toronto.edu

Frank Rudzicz
Toronto Rehabilitation Institute-UHN,
and University of Toronto
Toronto, Ontario
frank@cs.toronto.edu

ABSTRACT

In this paper we describe a means for disambiguating vague search queries by automatically asking users a series of clarifying questions. We demonstrate how a structure of appropriate questions can be generated by greedily choosing those that will yield the maximum information gain when answered. Data are manually annotated with indicator variables of interest to train a classifier that achieves 96% precision on our test set. We then use a modified version of the ID3 algorithm to select clarifying questions and present a functional prototype that employs our algorithm for query disambiguation within the context of caregiving in healthcare.

CCS CONCEPTS

• **Information systems → Query intent**; *Document filtering*;

KEYWORDS

iterative search; document filtering

ACM Reference Format:

Muuo Wambua, Stefania Raimondo, Jennifer Boger, Jan Polgar, Hamidreza Chinaei, and Frank Rudzicz. 2018. Interactive search through iterative refinement. In *Proceedings of 2nd International Workshop on Conversational Approaches to Information Retrieval (CAIR'18) (CAIR'18).* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Modern search engines place an enormous amount of information within reach, but can offer disappointing results when users are unable to form their queries with sufficient specificity. Delivering relevant and specific results is particularly important for matters of health and wellness. Consider, for example, family caregivers for individuals with dementia, who require highly specific, personalized, and accurate information, but are also often exhausted and limited in time [5, 28]. Given that most users do not venture past the first page of search results[4] or even the first three results [36], it is crucial to support carers and those with similar needs. The goal of this project is to support carers by automatically providing a series of questions which help clarify the intent of a user's initial query. While the focus of our work is supporting queries related to dementia and Alzheimer's disease, our technique is applicable to other domains.

We consider a means of clarifying search queries that modifies the ID3 algorithm [26] to automatically construct an optimum decision tree given a set of 'observations' (i.e., documents) and associated labels, to narrow down the search as quickly as possible. Each clarifying question reduces the number of search results, and each question corresponds to a branch in the decision tree. We find a sequence of questions by greedily searching for the question that provides the most information gain at each branch, given training data. The search terminates when a sufficiently small number of results remains relevant.

2 LITERATURE REVIEW

Query disambiguation, by determining the intended sense of a series of search terms, is an active area of research and is attempted through search personalization, query expansion, and clarification.

Personalization aims to tailor results to the user's interests by making use of user's short- (within session) and long-term (historic) browsing behaviour to disambiguate a user's query within a given session [3, 19]. For example, webpages previously or repeatedly visited by a user are often highly relevant to these or other similar users [12, 33]. Other systems have considered information external to searches, such as email messages, calendar items, or documents on the user's device [35]. On the other hand, users may be directly asked to provide a list of interests or to rate certain pages as relevant to one or more topic profiles [22], although this is often burdensome for the user and is often undesirable. The richer the representation of the user, the better [35]. The approach presented in this paper does not currently make use of this sort of automatic personalization.

Query expansion aims to rewrite the query so as to retrieve a smaller set of results [8]. For example, queries can be supplemented with words similar to the original query: Liu and Chu [17] used the Unified Medical Language System (UMLS) to append the original query with additional terms specifically related to the user's scenario, while more recent efforts by Roy et al. [30] have also attempted to use word embeddings to find semantically similar search terms. The additional search terms can also be produced by

CAIR'18, July 2018, Ann Arbor Michigan, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$0.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

relating queries to ontological categories [9] such that a query such as 'restaurants in Toronto' may be supplemented with the search terms 'café,pub,dinner'. These additional terms may also be derived from an initial set of retrieved documents, as in the pseudo-query reformulation of Diaz [11] and other pseudo-relevance feedback mechanisms. In our system, instead of expanding query results, we aim to clarify them through user input. However, these sorts of expansion techniques could be useful in our setting.

Methods which make use of ontologies are the most similar to the approach taken here, in that we also make use of a predefined set of relevant semantic categories with which we subdivide our results. Gauch et al. [12] began with an existing ontology (such as Yahoo's subject hierarchy [7]) and trained a classifier to automatically classify pages into ontological concepts which is then used to re-rank or filter results based on the user's profile of concepts. Categories can also be dynamically generated: Bordogna et al. [6] automatically clustered search results into subsets, automatically generating a proposed query for each.

However, none of these query expansion techniques used direct feedback from the users. While clarification is a common phenomenon in human communication [25], few information retrieval systems directly attempt to clarify a user's intended meaning. Asking a user for clarification can take many forms: Anastasiu et al. [1] and Igarashi et al. [13] explicitly asked the user to fill in additional contextual information related to their query; while Igarashi et al. [13], Bordogna et al. [6] and Soldaini et al. [31] presented the users with a list of potential improved queries - Bordogna et al. [6]'s being automatically generated to represent clustered subsets of search results, while Soldaini et al. [31]'s were developed using synonym mapping in the medical domain. Recently, however, interest has grown in "dynamic" search incorporating user feedback, specifically with the goal of handling complex search tasks that exist in professional domains [15, 38].

In natural conversation, queries are often clarified by posing questions back to the other party. Natural queries for clarification have been applied to question answering [10] and information retrieval systems with speech interfaces [21]. Misu and Kawahara [21] presented the user with questions from a pool of candidates, using information gain (IG) as the selection criterion. These questions were developed using a structured knowledge base and the system is applied to a small tech-support website. Indeed, Radlinski and Craswell [27] described a theoretical framework for conversational search – in which the user's true information need, which is often difficult to formulate, is dynamically clarified through back and forth presentation of options and ratings provided by the user.

Our work combines and expands these concepts by automatically selecting distinguishing questions whose answers are learned from webpages automatically.

3 EXPERIMENTAL SETUP

Our algorithm filters search results based on user responses to clarifying questions, presenting only pages which are tagged with the categories, or 'indicator variables', associated with the answers to those questions. 'Indicator variable' tags correspond to a Indicator:Value pair, such as *Audience:Researchers* or *Payment:Subscription.* Web pages are automatically tagged using a classifier trained on a labelled corpus of web pages. The question to present to the user is selected in order to maximize the possible information gain provided by their answer. The full system is shown in Figure 1.

3.1 Search

Search is achieved using Apache Solr[1], an open-source, full-text search engine. With it, we index 97,940 documents, obtained by crawling 7402 websites using Apache Nutch[2]. This process involved identifying a seed set of top international resources for dementia care, including major hospitals, Alzheimer's disease associations and societies, and relevant research institutions; from this seed set, additional pages were added in a breadth-first search to a predetermined depth. Indexed documents include all plain-text and PDF files found by the crawler.

Search queries are run against the Solr instance, and the set of search results consist of the top n relevant documents, ranked according to their Okapi BM25 scores [29]. Given a query Q (containing keywords $q_1, \ldots, q_n$), the BM25 score of a document D would be expressed as:

$$\text{score}(D, Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{\text{avgdl}}\right)} \quad (1)$$

where:

- $f(q_i, D)$ is the term frequency of q_i in the document D
- $\text{IDF}(q_i)$ is the inverse document frequency weight of the query term q_i
- $|D|$ is the length of document D in words
- avgdl is the average document length in the text collection.
- k_1 and b are free parameters either set to defaults or chosen through optimization.

3.2 Data collection

The set of 'indicator variables' was developed with focus groups [24] to determine the most important kinds of information to caregivers of individuals with dementia. These are provided in Table 1. Some examples include whether i) the sought resource relates to online information (or is associated with a physical location or service), ii) the progression and management of dementia, and iii) information required to manage the disease, despite not being specifically related to dementia.

For validation, we require a set of gold-standard, manually labelled indicator variables for the documents in the index. However, manually annotating indicator variables of ∼100,000 documents was infeasible; therefore, labels were generated by a human annotator for 34 sites that were determined to be relevant and representative by experts. As each website consisted of many sub-pages, this resulted in a corpus of 2520 labelled documents. The distribution of indicator variables is provided in Table 4.

3.3 Question selection

The clarification questions presented to the user are selected using a modified ID3 algorithm, which was designed to generate decision

[1]https://lucene.apache.org/solr/
[2]https://nutch.apache.org/

Table 1: Indicator Variables with their associated labels.

	Indicator variable	Values
ind-1	Audience	Patients, Caregivers, Health Care Professionals, Researchers
ind-2	Payment	Free, Subscription, Optional Subscription
ind-3	Location	Physical site, Web Only
ind-4	Forum	Yes, No
ind-5	Information Generation	Users, Institution
ind-6	Definitions of Disease	Yes, No, Sometimes
ind-7	Progression/Stages	Yes, No
ind-8	Medication/Treatment	Yes, No, Sometimes
ind-9	Diet/Nutrition	Yes, No, Sometimes
ind-10	Patient Neuropsyche	Yes, No
ind-11	Caregiver Neuropsyche	Yes, No
ind-12	Risk Factors, Warning	Yes, No
ind-13	Adult Day Care	Yes, No
ind-14	Community Help (unpaid)	Internal, External, Yes, No
ind-15	Community Help (paid)	Internal, External, Yes, No
ind-16	Driving/Other Activities	Yes, No
ind-17	Financial (Aid, Advice)	Yes, No
ind-18	Legal (Aid, Advice)	Yes, No

trees for generic tasks [26]. Given a set of documents C, the algorithm calculates the information gained in knowing the value of each attribute A_k. The attribute with the largest potential information gain is then used to split C into subsets according its value, producing a node in the decision tree. The algorithm then iterates recursively on the resulting subsets of C.

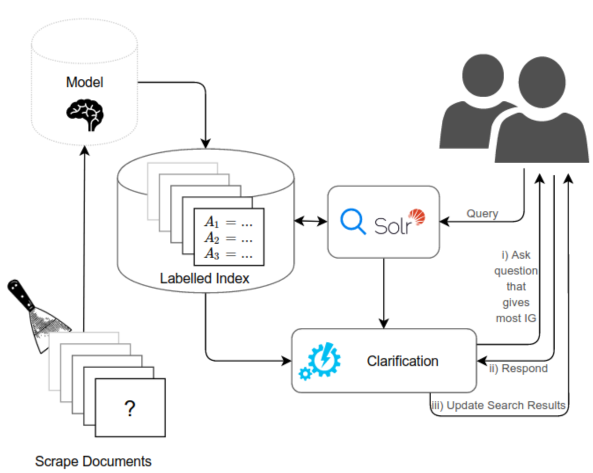


Figure 1: Query Clarification by means of Information Gain

However, we are not interested in generating a single decision tree *a priori* for all possible queries, but instead in producing a decision tree dynamically according to the user's preferences and initial input. Our system asks the user questions that specify values a_i to attributes A_k after each iteration. This is therefore a version of ID3 where all irrelevant branches are pruned after each iteration based on information from the user. At each step, we select a question to present to the user with the highest expected information gain.

In search, we are interested in knowing which documents in our working set, D, are relevant to the user's needs. The uncertainty we have about the possible relevance of each document can be expressed as a function of a discrete probability distribution $P(D)$ as shown in Equation 2, which is the likelihood that our user will find a desired document in our working-set. For simplicity, we also assume that the user is only interested in one particular document in D, and initially $P(D) = 1/|D|$, but this assumption is open to change.

$$\begin{aligned} H(D) &= -\sum_{d \in D} P(d) \log_2 P(d) \\ &= -\sum_{d \in D} \frac{1}{|D|} \log_2 \frac{1}{|D|} \end{aligned} \tag{2}$$

The uncertainty present after the user reveals an answer a to the question A_k can similarly be expressed as function of a probability density function $P(D \mid A_k = a)$, as shown in Equation 3.

$$H(D \mid A_k = a) = -P(D \mid A_k = a) \log_2 P(D \mid A_k = a) \tag{3}$$

Once again, for the sake of simplicity, we assume that the user's answer is accurate and the one relevant document is present in the set documents whose attribute $A_k = a$. This allows us to use a $P(D \mid A_k = a)$ as expressed in Equation 4.

$$P(D \mid A_k = a) = \begin{cases} \frac{1}{|D_{A_k=a}|} & \text{for } 0 < |D_{A_k=a}| \\ 0 & \text{for } |D_{A_k=a}| = 0 \end{cases} \tag{4}$$

The full information gain of obtaining any answer to a question corresponding to indicator-variable A_k is the difference between the initial uncertainty $H(D)$ and the average uncertainty after revealing some value a of A_k. The latter is found by taking into account $P(A_k = a)$, the probability that the user would give the answer a. There are a number of ways of estimating this value, including using past user interactions. However, since we lack sufficient historical data, we estimate it using the fraction of documents in the index corresponding to the provided answer, i.e.,

$$P(A_k = a) = \frac{|C_{A_k=a}|}{|C|} \tag{5}$$

The final expression for information gain is provided in Equation 6 and can be expressed as a function of document counts by plugging in Equations 2, 3, 4, and 5.

$$\begin{aligned} IG(D, A_k) &= H(D) - H(D \mid A_k) \\ &= H(D) - \sum_a P(A_k = a) H(D \mid A_k = a) \end{aligned} \tag{6}$$

We can now select the indicator variable A_k that results in the maximal IG, and prompt the user to provide an answer a to the clarifying question corresponding to that variable. We then update

D such that $D = D_{A_k=a}$, and repeat the process until we have a sufficiently small set of relevant search results, $|D|$.

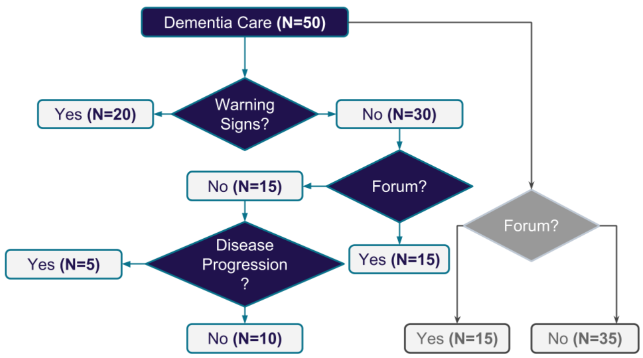


Figure 2: An example of the ID3 algorithm applied to query clarification (Derived from a session with the system). An initial query of 'Dementia Care' will yield a number of documents which can be further broken down by their relevance to concerns about warning signs or disease progression, and to connecting through forums.

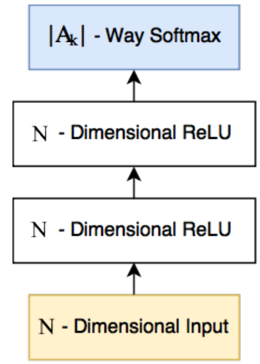


Figure 3: The neural network classification architecture: for each indicator variable A_k evaluated, the model is run with a N-dimensional vector input, where N is the number of features for a given feature set. It generates a one-vs-rest $|A_k|$ dimensional output, where $|A_k|$ is the number of values that can be assigned to A_k.

3.4 Document Classification

To perform the filtering based on indicator variables described in the previous section, we must be able to associate documents with the values of indicator variables. With our corpus of documents labelled with their respective indicator variables, we use supervised learning to obtain a model that automatically labels new documents with inferred indicator variables. Two thirds of the the labelled documents ($n = 1688$) was used for training, and the remainder for testing. We train a separate classifier for each indicator variable.

We experiment with four types of classifiers: naïve Bayes (NB, with Gaussian distributions and class priors reflecting the observed class frequencies), support vector machines (SVMs, with a linear kernel $K_{lin}(x_i, x_j) = {x_i}^T x_j$, regularization parameter $C = 1$), artificial neural networks (ANNs, with two hidden layers, as shown in Figure 3, trained with adam optimizer and categorical cross-entropy loss, batch size of 32, learning rate of 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$) and Joulin et al. [14]'s `fastText`. The latter is an efficient model for sentence classification that first learns an embedded representation of documents which are then fed into a multinomial logistic classifier. It has been competitive with state-of-the-art deep learning models and provides our baseline.

3.4.1 Feature Selection: The NB and SVM classifiers make use of lexical features that encode the probability of various unigrams and bigrams occurring in documents of a specific class. Specifically, we use tf-idf weighted unigram and bigram counts.

We consider the following five sets of features:

(1) **Individual token frequency:** Each document is represented by a vector where each element contains the number of times the unigram, or bigram, occurs in the document.
(2) **Token frequency with tf-idf weighting:** By using the product of tf, and idf, we penalize words that are present in most documents (e.g., "the", "a", "is").
(3) **Mean of global vectors (GloVe):** GloVe vectors [23] are distributed representations of words obtained by training aggregated global word-word co-occurrence statistics from a corpus. Here, we generate features by averaging GloVe vectors corresponding to each word occurring in the corpus.
(4) **Mean of word2vec vectors:** Word2vec [20] is an alternative model for learning word embeddings from raw text. We use a model that was trained on the Google News dataset.
(5) **Document (doc2vec) vectors:** Doc2vec [16] is an extension of word2vec that learns fixed-length vector representations of documents by predicting document contents.
(6) **Lexical-syntactic features:** These are obtained from syntactic parses and part-of-speech tagging of our corpus. These features, modeled after the work of Yancheva et al. [37], can be considered to reflect the "style" of the text which may vary depending on target audience or communicative goal and these include:
   (a) The relative frequency of context-free productions.
   (b) Phrase-type proportion, rate, and mean length.
   (c) Depth of syntactic parse trees.
   (d) Subordination and coordination phrase ratios.
   (e) Measures of word quality such as imageability, age-of-acquisition, familiarity, and transitivity.
   (f) Cosine distance between pairs of vectorized sentences within a document

4 RESULTS AND ANALYSIS

4.1 Indicator variable classification

Here, we evaluate the supervised classifiers that label documents with indicator variables, described in Section 3.4. We evaluate precision, recall, and F1 scores for each possible pair of feature and classifier type. In cases where indicator variables can take more than two values, the provided scores are the average for each value.

All combinations of feature-type and classifier-type yield high average scores, as shown in Table 2. However, a combination of tf-idf

Table 2: Precision, Recall and F1 scores averaged across the Indicator Variables.

		Mean		
Feature	**Model**	**Precision**	**Recall**	**F1**
Token f	GNB	0.81	0.78	0.79
	SVM	0.83	0.82	0.82
	NN	0.84	0.84	0.84
	KNN	0.77	0.77	0.76
tf-idf	GNB	0.83	0.79	0.81
	SVM	**0.96**	0.86	**0.89**
	NN	0.87	0.89	0.88
	KNN	0.72	0.69	0.67
Word2Vec	GNB	0.59	0.67	0.53
	SVM	0.77	0.63	0.64
	NN	0.86	0.84	0.85
	KNN	0.79	0.77	0.78
GloVe	GNB	0.60	0.66	0.54
	SVM	0.78	0.71	0.72
	NN	0.90	0.84	0.85
	KNN	0.79	0.76	0.772
Doc2Vec	GNB	0.61	0.65	0.57
	SVM	0.81	0.85	0.82
	NN	0.87	0.87	0.87
	KNN	0.85	0.84	0.85
Lex-Syn	GNB	0.59	0.65	0.53
	SVM	0.62	0.58	0.56
	NN	0.63	0.62	0.62
	KNN	0.53	0.47	0.38
FastText		0.80	**0.95**	0.84

weighted vectors and a linear SVM yield the best performance, with precision of 0.96 and recall of 0.89, with relatively little variation across indicator variables, as shown in Figure 4. The tf-idf weighted vectors outperform plain token-frequency vectors because they factor in the frequency of tokens in the corpus. This helps prevent the model from placing undue emphasis on frequently occuring, less informative words. The tf-idf SVM model outperforms `fastText`, having significantly higher precision and a higher F1 score.

Despite the small size of the training dataset, the ANN outperforms almost all models across all metrics, excepting SVM tf-idf precision and F1 and GNB Lex-Syn recall. On average, the difference is particularly stark for the word embedding models, Word2Vec and GloVe, suggesting that the neural network is able to better capure non-linear relationships between individual features in these feature sets than other models. The performance of the ANN may likely be further significantly improved by regularization: possible over-fitting to the training data may be prevented by early stopping and introducing dropout [32] layers to bias the network towards simpler models. However, even this simple implementation suggests the promise of ANNs for this task.

The relatively poor performance of the distributed vector representations, Word2vec and GloVe, may be due to the composition of a single vector for each document by averaging out the vector representations of its constituent words. Order-sensitive representations such as sequence models [16] and tree-structured LSTMs [34] have fared better at generating semantic sentence representations that account for difference in meaning as a result of differences in word order or syntactic structure.

Overall, the poorest classification is achieved for IVs 2, 6, 8, 9, and 14. Each of these IVs have highly imbalanced possible values with nuanced or potentially overlapping values (e.g. 'Sometimes'/'Yes'/'No' and 'Yes'/'No'/'External'/'Internal'). Improvements to these difficult to classify categories may be achieved with a larger more nuanced dataset.

4.2 Query clarification

4.2.1 Method. To evaluate query clarification, we consider how clarification would affect the rank of relevant documents and the number of search results returned. However, without a standard set of query/relevance judgments for our labelled websites, we must instead automate the query and search process. To do so, we randomly select documents from our index and generate queries that are relevant to them, but vague enough to also be relevant to other documents.

One way of generating these queries is to generate a word or sequence of words that summarize a document's contents. We evaluated two methods of generating relevant queries: keyword extraction and extractive summarization, since these would result in relatively abstract or general representations for the contents of the selected documents, and of other documents. Our methodology is similar to that described by Azzam et al. [2] in that we use co-reference chains to find either the words or sentences that are most representative of each document.

Both methods first involve performing co-reference resolution using Manning et al. [18]'s CoreNLP. For keyword extraction, we then find the co-reference chain with the most mentions in document (representing the most common entity therein), and select the most representative mention of that chain as our keyword. This keyword is selected with preference to proper noun mentions or mentions with more pre-modifiers. Extractive summarization, by contrast, finds the sentence traversed by the most coreference chains. Examples of both are provided in Table 3.

Naturally, we are only able to reliably resolve coreference chains for articles written in English.

4.2.2 Evaluation. Using the query generation described in Section 4.2.1, we randomly select 200 documents and extract both summarizing sentences, and keywords. Each of these documents has been labelled using the classifier with the best average F1 score, which was found to be a linear SVM trained with tf-idf feature vectors. These 200 documents are then narrowed down to lists of 50 and 82 by respectively manually cleaning out sentences and terms in languages besides English and ones that did not carry non-generic co-reference chains.

We then perform the following tasks for each document:

(1) Query Solr for the terms of interest.

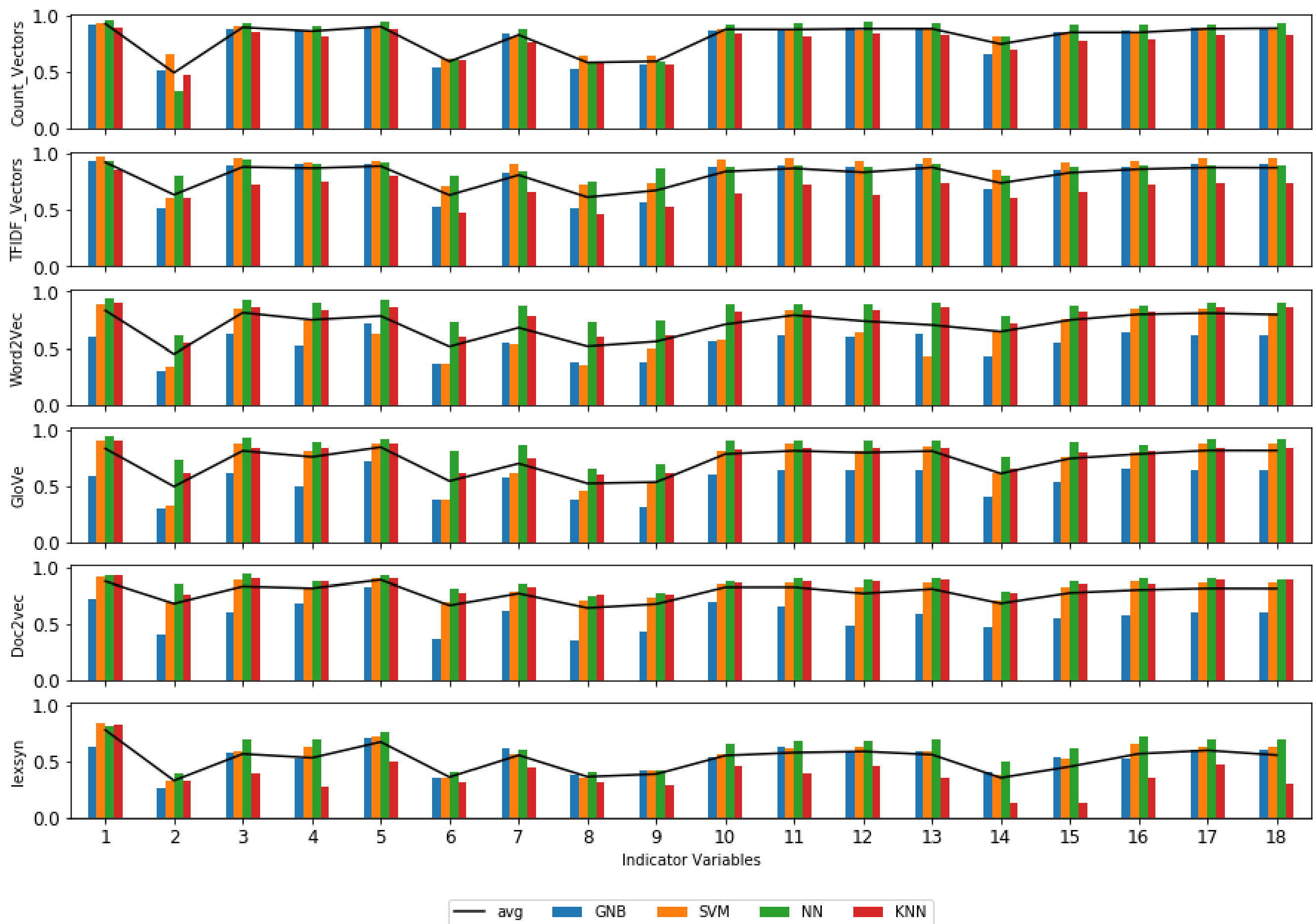


Figure 4: Accuracy of classification models for each feature set across indicator variables (described in Table 1). The average accuracy across models for a given indicator and feature set is shown by the black trendline.

(2) Find the indicator-variable with the most information gain, and assign it its appropriate value.
(3) Narrow the search results using the inferred answer to the question.
(4) Repeat steps 1-3 until either each possible question is asked or none of the remaining questions would yield any information gain.

The results are presented in Figures 5 and 6 for sentences and keywords, respectively. It is evident that, in both cases, clarification helps narrow down the search substantially in 1 to 8 iterations. However, a difference emerged between queries generated by keywords and sentences, with regards to the distribution of the ranking of documents. Queries generated by sentences resulted in the relevant documents ranking very high, while queries generated by keywords were initially more evenly distributed and were forced towards the top as more clarification was provided. Whether coherent phrases in natural language are more amenable to search than traditional keywords is the topic of ongoing research.

5 CONCLUSION

Caring for someone with dementia can be physically, emotionally, and financially difficult. Access to support, professional and otherwise, can reduce negative aspects of caregiving, and can involve products, information, and services to support dementia care. While it is often left to family caregivers to find various forms of support, locating appropriate support can be difficult, frustrating, and often futile, as family caregivers may not know what they are looking for or how to find it.

The current work suggests that it is possible to effectively reduce the quantity of relevant search results by asking clarifying questions of users. It is evident that the quality of query clarification is dependent on the quality and type of examples of indicator-variable assignment. A prototype of this project is publicly available via CARE-RATE [3], which is a website for simplifying information retrieval related to dementia caregiving in the community. We are in the process of testing this interface with target-users to examine the effect of query-clarification on user search-behaviour and to

[3]https://care-rate.herokuapp.com

Table 3: Examples of extracted keywords and sentences from web pages, along with short text excerpts (scraped on April 18, 2017).

Keywords: **people with dementia** *http://www.alz.org/sewi/in_my_community_20372.asp* The Alzheimer's Association Memories in the Making® program offers creative art expression for individuals with early to the middle stages of Alzheimer's disease. Even after people with dementia have lost the ability to use words, they are able to paint their thoughts, emotions and memories in a manner that is expressive and beautiful. Art becomes their voice...
Keywords: **the addicted patient** *http://nethealthbook.com/drug-addiction/* Physical dependence manifests itself in withdrawal symptoms. Another feature is psychological dependence. It means that the use translates into feelings of satisfaction and a desire to repeat the experience. There is a feeling of discontent and intense craving if the drug is withheld. Addiction is characterized by a lifestyle where...
Keywords: **prostate cancer** *http://aboutbrachytherapy.com/cancer-types/prostate-cancer/introduction-to-prostate-cancer/* Prostate cancer occurs when abnormal cells develop in the prostate gland, usually after the age of about 45, although this can vary from individual to individual. West Indian and African men are more likely to develop this disease compared to Caucasian (white) and Asian men...
Sentence: **Some people with dementia may encounter problems with their sight – in some cases, this includes having hallucinations.** *https://www.alzheimers.org.uk/site/scripts/documents_info.php%3FdocumentID=110* The Alzheimer's Association Some people with dementia may encounter problems with their sight – in some cases, this includes having hallucinations. This page looks at some of the difficulties and mistakes this can cause, and suggests ways of providing support for the person. Understanding...

validate our approach through evaluation of user browsing patterns and relevance judgments.

Ongoing work involves unsupervised methods to identify and assign indicator variables to documents in the index using methods such as latent Dirichlet allocation. This will allow us to ask clarifying queries about topics that were not previously identified manually and overcome the limitation of per-website (instead of per-page) labels of the manual labelling method. We also intend to implement question clarification using a partially-observable Markov decision process, for comparison, and to incorporate the uncertainty in the indicator variable classification into the question clarification step.

6 SUPPLEMENTAL MATERIAL

Figures 7 to 10 illustrate how a user would go about using CARE-RATE for iterative refinement of search queries. This user session corresponds to the flowchart illustrated in figure 2.

REFERENCES

[1] David C. Anastasiu, Byron J. Gao, Xing Jiang, and George Karypis. 2013. A novel two-box search paradigm for query disambiguation. *World Wide Web* 16, 1 (2013), 1–29. https://doi.org/10.1007/s11280-011-0154-0

[2] Saliha Azzam, Kevin Humphreys, and Robert Gaizauskas. 1999. Using Coreference Chains for Text Summarization. In *Proceedings of the Workshop on Coreference and Its Applications (CorefApp '99)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 77–84. http://dl.acm.org/citation.cfm?id=1608810.1608825

[3] Paul N. Bennett, Ryen W. White, Wei Chu, Susan T. Dumais, Peter Bailey, Fedor Borisyuk, and Xiaoyuan Cui. 2012. Modeling the Impact of Short- and Long-term Behavior on Search Personalization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. ACM, New York, NY, USA, 185–194. https://doi.org/10.1145/2348283.2348312

[4] Mehret S Birru, Valerie M Monaco, Lonelyss Charles, Hadiya Drew, Valerie Njie, Timothy Bierria, Ellen Detlefsen, and Richard A Steinman. 2004. Internet usage by low-literacy adults seeking health information: an observational analysis. *Journal of Medical Internet Research* 6, 3 (2004).

[5] Jennifer Boger, Frank Rudzicz, Hamid Chinaei, Sigrún Kristín Jónasdóttir, M. Wambua, and Jan Polgar. 2017. CARE-RATE: Initial development of an artificially intelligent online tool for connecting caregivers to relevant support. In *Rehabilitation Engineering and Assistive Technology Society of North America (RESNA)*.

[6] G. Bordogna, A. Campi, G. Psaila, and S. Ronchi. 2009. *Query Disambiguation Based on Novelty and Similarity User's Feedback*. Springer Berlin Heidelberg, Berlin, Heidelberg, 179–190. https://doi.org/10.1007/978-3-642-04957-6_16

[7] Anne Callery and Deb Tracy Proulx. 1997. Yahoo! cataloging the web. *Journal of Internet Cataloging* 1, 1 (1997), 57–64.

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

[9] Abdur R Chowdhury and Gregory S Pass. 2009. Query disambiguation. (July 14 2009). US Patent 7,562,069.

[10] Matthias Denecke and Norihito Yasuda. 2008. *Does This Answer Your Question?* Springer Netherlands, Dordrecht, 219–246. https://doi.org/10.1007/978-1-4020-6821-8_9

[11] Fernando Diaz. 2015. Pseudo-Query Reformulation. *arXiv:1507.03928 [cs]* (July 2015). http://arxiv.org/abs/1507.03928 arXiv: 1507.03928.

[12] Susan Gauch, Jason Chaffee, and Alexander Pretschner. 2003. Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems: An international journal* 1 (2003), 219–234.

[13] Hisakazu Igarashi, Charles G Bird, and Andrew Moedinger. 2013. Prompt for query clarification. (July 9 2013). US Patent 8,484,190.

[14] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of Tricks for Efficient Text Classification. *CoRR* abs/1607.01759 (2016). http://arxiv.org/abs/1607.01759

[15] Evangelos Kanoulas and Leif Azzopardi. 2017. CLEF 2017 Dynamic Search Evaluation Lab Overview. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction (Lecture Notes in Computer Science)*. Springer, Cham, 361–366. https://doi.org/10.1007/978-3-319-65813-1_31

[16] Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Eric P. Xing and Tony Jebara (Eds.), Vol. 32. PMLR, Bejing, China, 1188–1196. http://proceedings.mlr.press/v32/le14.html

[17] Zhenyu Liu and Wesley W Chu. 2007. Knowledge-based query expansion to support scenario-specific retrieval of medical free text. *Information Retrieval* 10, 2 (2007), 173–202.

[18] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*. 55–60. http://www.aclweb.org/anthology/P/P14/P14-5010

[19] Lilyana Mihalkova and Raymond Mooney. 2009. Learning to disambiguate search queries from short sessions. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 111–127.

[20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781

[21] Teruhisa Misu and Tatsuya Kawahara. 2006. Dialogue strategy to clarify user's queries for document retrieval system with speech interface. *Speech Communication* 48, 9 (2006), 1137 – 1150. https://doi.org/10.1016/j.specom.2006.04.001

[22] Michael Pazzani, Jack Muramatsu, and Daniel Billsus. [n. d.]. Syskill & Webert: Identifying Interesting Web Sites. ([n. d.]), 9.

[23] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[24] Jan Polgar and Frank Rudzicz. 2016. An online resource for caregivers of persons with dementia. *Gerontechnology* 15.

[25] Matthew Richard John Purver. 2004. *The theory and use of clarification requests in dialogue.* Ph.D. Dissertation. King's College London, Department of Computer Science.

[26] J. Ross Quinlan. 1986. Induction of decision trees. *Machine learning* 1, 1 (1986), 81–106.

[27] Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval (CHIIR '17)*. ACM, New York, NY, USA, 117–126. https://doi.org/10.1145/3020165.3020183

[28] Doering Riley, Jan Polgar, Frank Rudzicz, and Jennifer Boger. 2017. Designing CARE-RATE: An Online Assistive Tool for Dementia Caregivers. In *Proceedings of HCIxDementia: The Role of Technology and Design in Dementia Workshop at CHI 2017.*

[29] Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at TREC–3, In Overview of the Third Text REtrieval Conference (TREC–3). 109–126. https://www.microsoft.com/en-us/research/publication/okapi-at-trec-3/

[30] Dwaipayan Roy, Debjyoti Paul, Mandar Mitra, and Utpal Garain. 2016. Using Word Embeddings for Automatic Query Expansion. *CoRR* abs/1606.07608 (2016). http://arxiv.org/abs/1606.07608

[31] Luca Soldaini, Andrew Yates, Elad Yom-Tov, Ophir Frieder, and Nazli Goharian. 2016. Enhancing web search in the medical domain via query clarification. *Information Retrieval Journal* 19, 1 (2016), 149–173. https://doi.org/10.1007/s10791-015-9258-y

[32] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[33] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. 2004. Adaptive Web Search Based on User Profile Constructed Without Any Effort from Users. In *Proceedings of the 13th International Conference on World Wide Web (WWW '04)*. ACM, New York, NY, USA, 675–684. https://doi.org/10.1145/988672.988764

[34] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. In *Association for Computational Linguistics (ACL)*.

[35] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2018. Personalizing Search via Automated Analysis of Interests and Activities. *SIGIR Forum* 51, 3 (Feb. 2018), 10–17. https://doi.org/10.1145/3190580.3190582

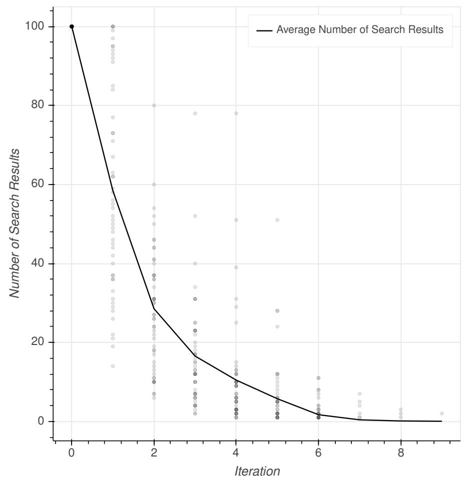
[36] Alexander JAM Van Deursen and Jan AGM Van Dijk. 2009. Using the Internet: Skill related problems in users' online behavior. *Interacting with computers* 21, 5 (2009), 393–402.

[37] Maria Yancheva, Kathleen Fraser, and Frank Rudzicz. 2015. Using linguistic features longitudinally to predict clinical scores for Alzheimer's disease and related dementias. In *6th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT)*. sn, 134.

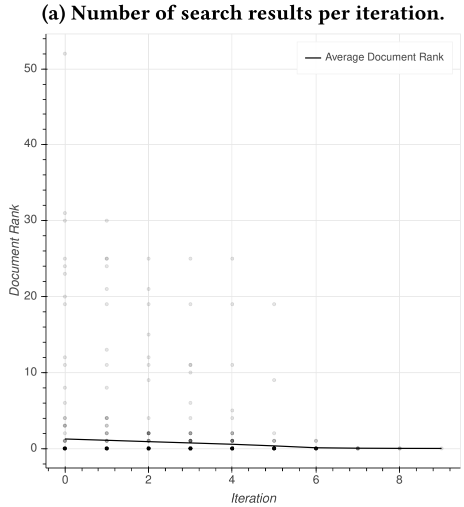
[38] Grace Hui Yang, Zhiwen Tang, and Ian Soboroff. 2017. TREC 2017 Dynamic Domain Track Overview. In *The Twenty-Sixth Text REtrieval Conference (TREC 2017) Proceedings*. Gaithersburg, Maryland, 16. https://trec.nist.gov/pubs/trec26/papers/Overview-DD.pdf

Table 4: Frequency of Indicator Variable Values in Labelled Corpus

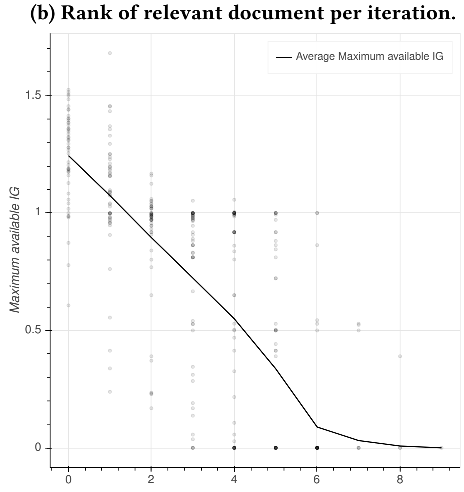
Indicator Variable	Value	*f*	%
ind-1	Patients	1921	76.2
	Caregivers	2280	90.5
	Health Care Professionals	1024	40.6
	Researchers	1097	43.5
ind-2	Subscription	14	0.6
	Free	2506	99.4
	Optional Subscription	24	1
ind-3	Web only	1284	51
	Physical site	1236	49
ind-4	No	1684	66.8
	Yes	836	33.2
ind-5	Institution	2414	95.8
	Users	530	21
ind-6	No	238	9.4
	Yes	2258	89.6
	Sometimes	24	1
ind-7	No	564	22.4
	Yes	1956	77.6
ind-8	No	535	21.2
	Yes	1961	77.8
	Sometimes	24	1
ind-9	No	821	32.6
	Yes	1675	66.5
	Sometimes	24	1
ind-10	No	823	32.7
	Yes	1697	67.3
ind-11	No	1126	44.7
	Yes	1394	55.3
ind-12	No	658	26.1
	Yes	1862	73.9
ind-13	No	1279	50.8
	Yes	1241	49.2
ind-14	External	535	21.2
	No	1086	43.1
	Yes	60	2.4
	Internal	1082	42.9
ind-15	External	900	35.7
	No	702	27.9
	Internal	1161	46.1
ind-16	No	1290	51.2
	Yes	1230	48.8
ind-17	No	1228	48.7
	Yes	1292	51.3
ind-18	No	1228	48.7
	Yes	1292	51.3



(a) Number of search results per iteration.

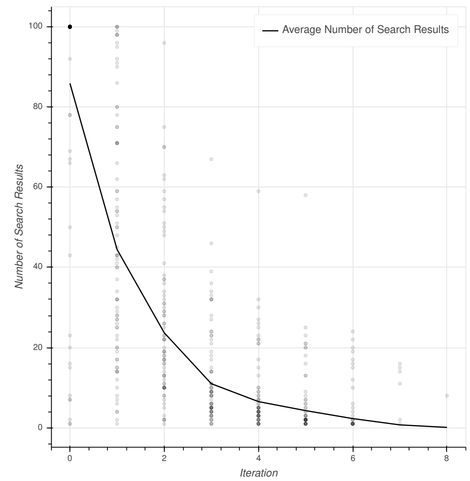


(b) Rank of relevant document per iteration.

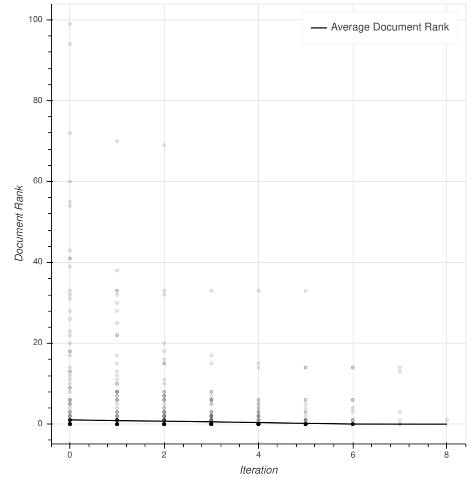


(c) Maximum available IG per iteration.

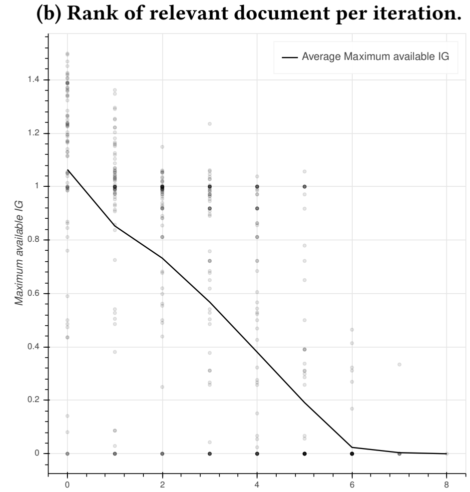
Figure 5: Change in number of search results, rank and maximum available information gain per clarification iteration; using sentences as search queries.



(a) Number of search results per iteration.



(b) Rank of relevant document per iteration.



(c) Maximum available IG per iteration.

Figure 6: Change in number of search results, rank and maximum available information gain per clarification iteration; using keywords as search queries.

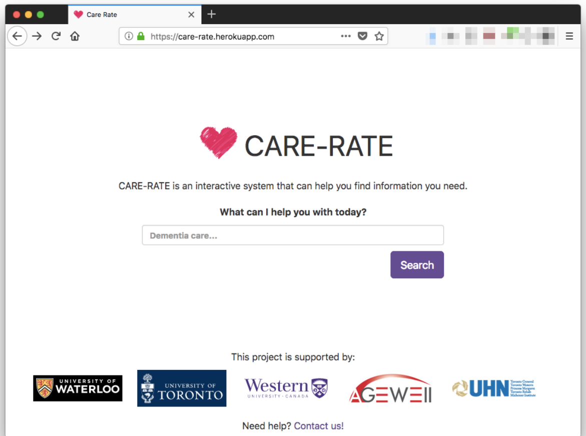


Figure 7: Landing page

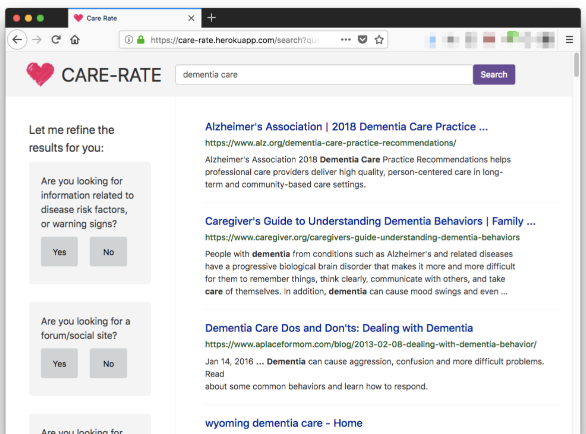


Figure 8: After providing a search query, users are presented with an unfiltered set of search results, and a list of clarifying questions ranked by their information gain.

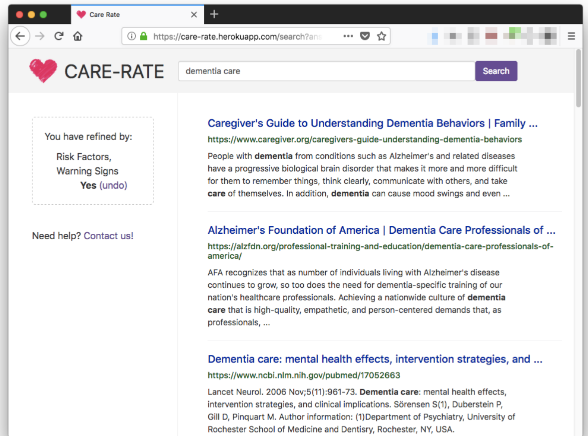


Figure 9: Answering 'yes' to the first question yields a state where the search results can not be narrowed down further. If desired, the user may revise their answer (by clicking 'undo') and return to the previous screen.

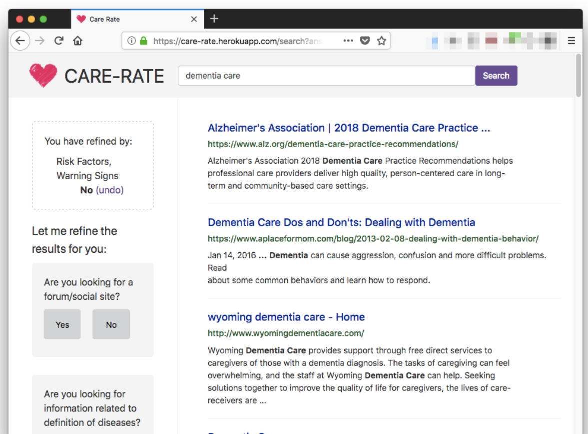


Figure 10: However, answering 'no' leads them down a path where other clarifying questions can provide further information. If desired, a question may be dismissed (by clicking an 'x' that appears at the corner of the box).