FOOD & BIOBASED RESEARCH
WAGENINGEN UR

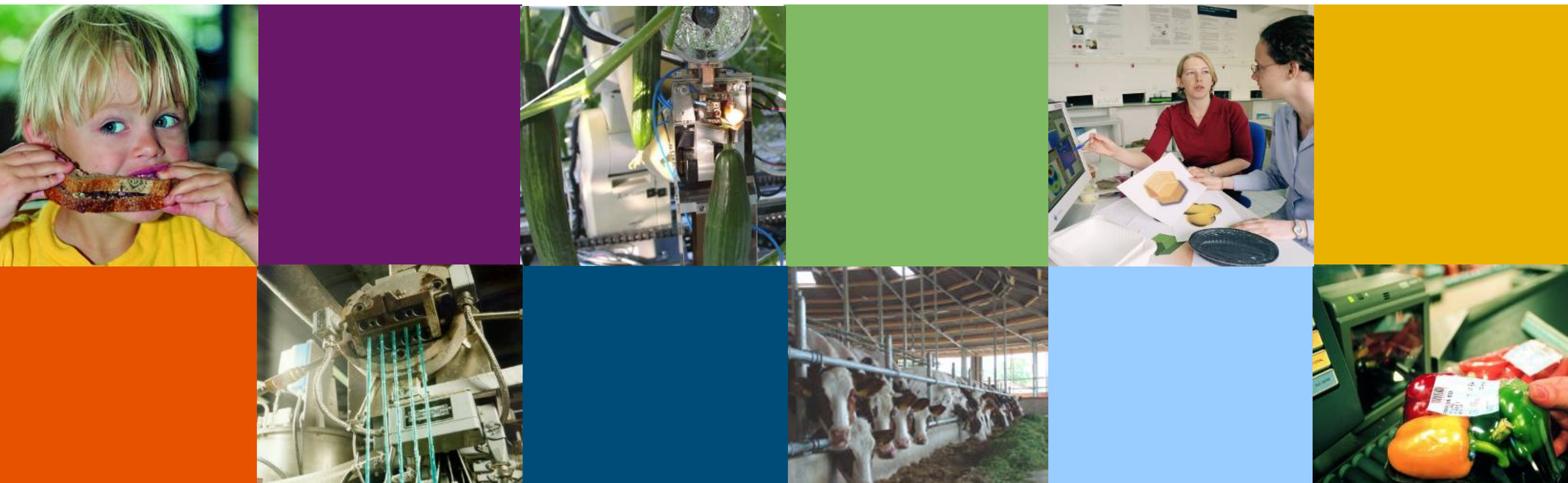
Software Description for SIEVE

A COMMIT/e-FoodLab report

Roeland van Batenburg, Don Willems, Jan Top

COMMIT/

Rapport nr. 1521



Colophon

Title	Software Description for SIEVE
Author(s)	Roeland van Batenburg, Don Willems, Jan Top
Number	Food & Biobased Research #
ISBN-number	
Date of publication	2014-04-15
Confidentiality	No
OPD code	
Approved by	Jan Top

Wageningen UR Food & Biobased Research
P.O. Box 17
NL-6700 AA Wageningen
Tel: +31 (0)317 480 084
E-mail: info.fbr@wur.nl
Internet: www.wur.nl

Abstract

This is a technical report of the method of the tool SIEVE, developed by Food & Biobased Research in the e-FoodLab project, part of COMMIT. SIEVE assists experts when creating a comprehensive set of documents relevant to their domain with relatively little effort. Such sets of documents can be used for reference by non-experts or as a source for literature studies. Using knowledge about the domain and labelled documents, the system presents the expert with possible relevant documents reducing the workload on the expert. SIEVE uses knowledge about the domain to generate queries (logical combinations of terms using AND and OR) and evaluate these with the labelled documents to find documents whose content is similar to the documents selected by the expert. We developed a user-friendly interface around the SIEVE method and used the tool for a project about protein innovations. The tool returned 106 documents in the first iteration, the expert evaluated 34% of these as relevant.

Content

1 Introduction

In the COMMIT/ supported e-FoodLab project we develop tools and software to improve the accessibility and quality of information, in particular within the food domain. Here we introduce a method that allows users to gain sets of document related to a domain they are interested in. It is implemented in SIEVE (**S**ift **I**ntelligently using **EV**aluation by **E**xperts). The goal of SIEVE is to support an expert when collecting a set of documents covering (part of) his or her domain. To demonstrate this method, we have applied it in the domain of food protein innovations.

1.1 Problem Context

When a large amount of information is available, it is difficult to select the right subset of information for a specific domain. This is a common problem when searching the web, but it also occurs when creating a comprehensive set of relevant documents from for example a large company-based repository. A possible solution might be to ask an expert from the relevant domain or field for a set of relevant documents. The expert has a working knowledge of the domain and can with some effort retrieve a set of documents that is important to the field. However, in most cases this set is still limited. Furthermore, the consulted expert cannot be expected to have read every document that might be relevant.

Comprehensive sets of documents can for example be used to create a website for end-users. These end-users know they want information about a specific domain, but they might not know the proper way to describe this domain. If they can use a website about this domain to search through documents related only to that domain, they can find the required information with far less effort. This marks the difference of the task addressed here with regular search tasks. In that case the goal is to answer a specific question using a query that preferably leads to one best document. The target for gathering a set of documents that covers a domain is more general and non-specific.

As our user scenario, imagine Johan as an expert in the field of protein innovations. For a portal about this subject, he has been asked to collect a set of publications. These documents will be presented to the visitors who can search through them. Johan has a good knowledge of the domain and can easily provide a little over a hundred relevant documents. These documents give a good overview of the field, but the portal wants to collect *all* relevant documents. In other words he needs to spend time on searching for many more relevant documents.

2 Methods

We assume that the expert initially provides a small set of manually collected documents covering the domain at hand. Our method aims to find documents in a larger repository of documents that are similar to this set. It does so by generating a query from a small domain vocabulary and using the initial document set as a reference set. We construct the query that best matches the set of known relevant documents, but also finds unknown relevant documents. From these the expert selects which documents are indeed relevant. These are added to the set of relevant documents. With this information the tool provides additional suggestions, the experts can repeat this until a sufficiently large set of documents is gathered or no more relevant documents can be found.

In principle the set of relevant documents can be sufficient input to search for similar documents. It is possible to generate a fingerprint for the documents using the important words [1, 2]. These methods however rely on extracting the important terms correctly from the documents, something which is not trivial. If we assume that a separate vocabulary of the domain is available, we can use this to better identify the important terms. It is feasible to make a vocabulary very quickly using associations[1]. Because the terms in the vocabulary are provided by an expert we know that these are the important ones. Then we simply have to identify which combinations of terms are typical for relevant documents.

Now the task of the tool is to generate an optimal query from terms in the vocabulary using the known relevant documents as an optimisation criterion. An ideal query would at least find all the relevant documents of the expert. That means we want a query with high recall with respect to the initial set. We also require high precision because then we know that the query is not too broad. However, since we want to find new documents as well, precision should not be one hundred percent. By setting a fixed number of new documents as a goal we can also make sure that the expert is not overwhelmed with new documents. We refer to Section 2.5 for more detail on this.

We can describe the goal of the method with the Venn diagram in Figure 1. Each set represents a set of documents. The Repository contains all known documents. The Domain contains the documents that are relevant for the domain. The Expert selects a subset of these documents. Sieve generates a Query that finds new documents. The Query matches a part of the known relevant documents and extends in that 'direction' selecting more documents in the domain. This 'direction' is given by the terms that are used in the query. For instance a query might contains terms about economics, resources and protein innovations. Then the 'direction' will be the economic aspects of protein innovation and the query will select documents about that subject. These main themes of the domain can be retrieved from the vocabulary and our method will

[1] *For this we have developed and applied ROC+, a tool to have non-IT experts quickly generate simple domain vocabularies. ROC+ is also an output of COMMIT/eFoodLab.*

generate different queries matching different ‘directions’, as seen in Figure 2. Another possible direction could be studies of the nutritional value of new proteins.

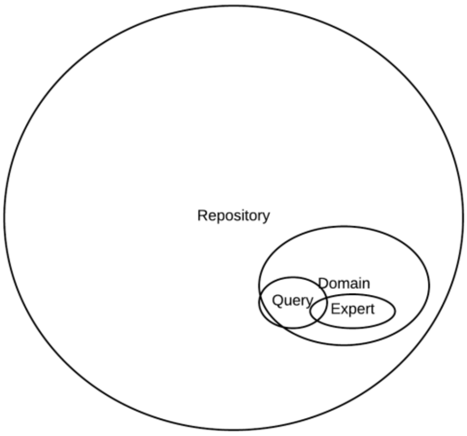


Fig 1. Venn diagram of the different sets of documents.

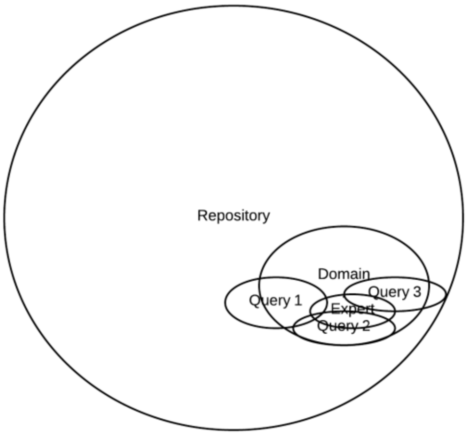


Fig 2. Venn diagram of the different sets of documents with multiple queries.

We will first describe in detail the different inputs we have (vocabulary, repository and relevant documents). Then we show how we generate possible queries and how we measure the performance of the different queries and the selection method of these queries and the resulting documents.

2.1 Vocabulary

To start off, the expert creates an vocabulary in the form of an ontology. An ontology has more structure than a simple vocabulary and we use this structure, therefore we will referrer to the vocabulary with the term ontology in this section. A knowledge expert interviews the expert to determine important concepts for the domain. These are entered into ROC, which suggests additional terms. After multiple iterations of this process the most important concepts in the field of protein innovations have been collected and organised using `rdfs:subClassOf` relations. Ideally, the ontology will contain all terms that are used in the domain. However, the method will work even with a few terms as long as these are common enough in the documents about this domain.

2.2 Repository

The repository is the source of new documents, this can be any search endpoint on which we can execute queries and retrieve documents. It can for instance be Google, Web of Science or a private index of documents.

2.3 Selected Documents

The relevant documents selected by the expert are indexed by a search engine in order to make searching them possible.

2.4 Query Generation

All possible queries can be constructed by combining any subset of ontology terms with the logical operators AND and OR. However, this will result in the huge number of possible queries that can never be executed with reasonable resources. For this purpose we use the structure of the ontology to group terms and apply the logical operators.

We assume that the ontology contains several subjects within the domain. A combination of these subjects using the AND-operator is the "direction" mentioned in the beginning of Section 2, this is further elaborated in Section 2.5. We define the subjects as the top level concepts in the ontology. Then we assume that all subconcepts and their labels are alternative or partial representations of the top level terms. Thus we can use the "OR"-operator to generate a query that matches any of these representations.
That way for example the concept "fish" would also match a document that contains the word "herring" (if that is the label of one of the subclasses of "fish"). So, if concept A has subclasses Z, Y and each class has two labels then the query for A would become (label_A1 OR label_A2 OR label_Z1 OR label_Z2 OR label_Y1 OR label_Y2). See Figure 3. With this expansion we only have to combine the thirteen top level concepts with AND to generate all possibilities.

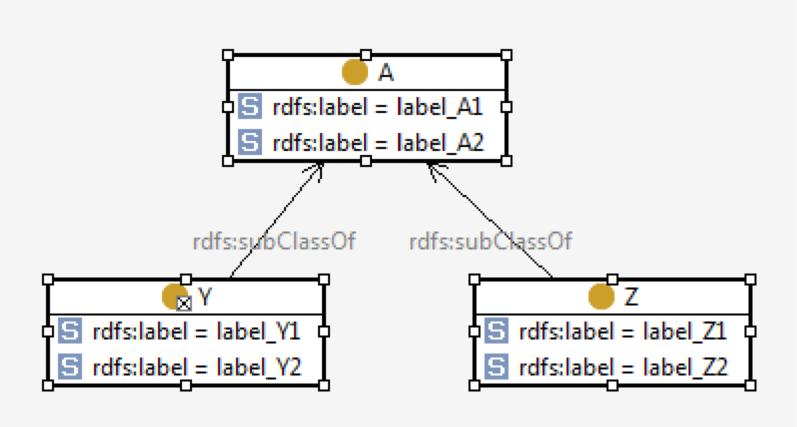


Fig 3. A very simply ontology showing top level concept A, subclasses Y and Z and their labels

2.5 Best-first search

In order to limit the number of queries we have to execute and evaluate, we can combine them in a tree and only expand promising parts. To be able to do this SIEVE requires two things, a tree structure and an evaluation method. For the tree structure we simply start with the most restrictive query and gradually drop terms, making them less restrictive. The most restrictive query requires all top level concepts to be present in the documents. On the second level we allow one concept to not be present (in the protein example there are thirteen possibilities then). For an illustration with 4 top level terms see Figure 3.

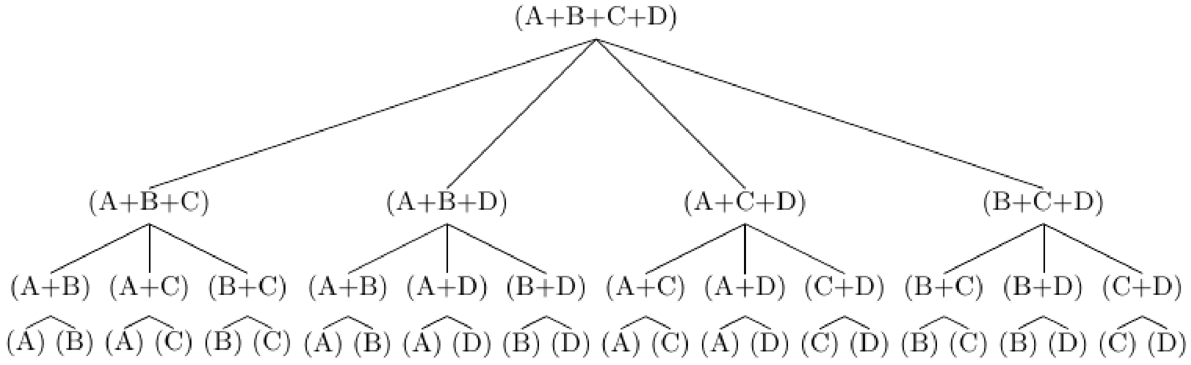

(A+B+C+D)
(A+B+C)
(A+B+D)
(A+C+D)
(B+C+D)
(A+B) (A+C) (B+C) (A+B) (A+D) (B+D) (A+C) (A+D) (C+D) (B+C) (B+D) (C+D)
(A) (B) (A) (C) (B) (C) (A) (B) (A) (D) (B) (D) (A) (C) (A) (D) (C) (D) (B) (C) (B) (D) (C) (D)

Fig 3. The search tree for four top level concepts.

To evaluate a query we look again at Figure 2. We want a query that covers as much of the domain as possible and includes as little of the rest of the repository as possible. However, in principle the only way to determine whether a document is in the domain is to ask the expert. This is far too labour intensive to be an efficient way to determine the relevance of all found documents.

As an alternative we can determine how good the query represents the documents selected by the expert. Our hypothesis is that if the new set closely matches the reference document set, then the other documents that it returns should also be relevant.

A good way to determine how closely the query matches the selected documents is by using the concepts *precision* and *recall*. Precision indicates how many of the returned documents are known to be relevant and recall indicates how many of the known relevant documents were returned. If both numbers are 100% then the query returns exactly the expert set, if they are both 0% then none of the found documents match with the reference documents. A precision of 50% indicates that half of the returned documents are in the expert set and a recall of 50% indicates that 50% of the documents in the expert set are matched by the query. We do not want a 100% precision (then we have nothing new), but prefer queries with a value as high as possible. We combine the precision and recall in the so-called F1-measure. This gives a value between 0 and 1 and gives equal weight to both precision and recall. To make sure we have a close match with the selected documents but also return new documents we keep selecting the highest performing nodes until we have retrieved a fixed number of new documents. This number should not be too high in order to give the expert the chance to influence the direction of the search and not to overwhelm him with documents to be evaluated.

Now that we can evaluate nodes, we have to walk through the tree. We used the best-first search algorithm. This simply expands the most optimal node in the tree and continues until no node with higher value exists. We have adapted the best-first search algorithm to retrieve multiple nodes from the tree. The algorithm searches through the tree until a local maximum is found and then continues searching ignoring that part of the tree. We want multiple results, as we decided to retrieve multiple combinations of AND-queries in order to be able to ignore the OR (on the highest level). Now that we have a way to search the tree, we can add the additional requirement that we want to retrieve new documents. We simply do this by letting the algorithm search through the tree until a minimum number of new documents has been retrieved. The total flow of walking through the tree is described in Figure 4.

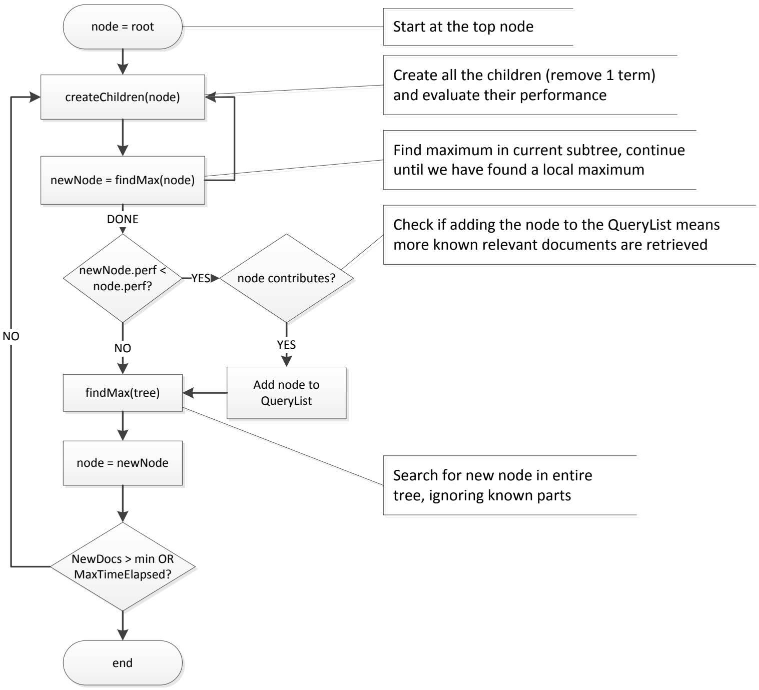

node = root
Start at the top node
createChildren(node)
Create all the children (remove 1 term) and evaluate their performance
newNode = findMax(node)
Find maximum in current subtree, continue until we have found a local maximum
DONE
newNode.perf < node.perf?
YES
node contributes?
Check if adding the node to the QueryList means more known relevant documents are retrieved
NO
NO
YES
findMax(tree)
Add node to QueryList
node = newNode
Search for new node in entire tree, ignoring known parts
NewDocs > min OR MaxTimeElapsed?
end

Fig 4. The flow through searching through the tree. For more details see below.

This is quite complex, to illustrate we will step through an example tree. We start with seven directions in the protein innovation domain: nutrient (N), source (S), meat alternative (M), product (P), brand (B), chain link (L) and consumer (C). First we create the seven possible children and evaluate their performance:

- N, S, M, P, B, L, C. perf=0.210
    - S, M, P, B, L, C. perf=0.190
    - N, M, P, B, L, C. perf=0.212
    - N, S, P, B, L, C. perf=0.189
    - N, S, M, B, L, C. perf=0.213
    - N, S, M, P, L, C. perf=0.067
    - N, S, M, P, B, C. perf=0.203
    - **N, S, M, P, B, L. perf=0.214**

Now comes the findMax step, this picks on this level the top performing node and expands that until a local maximum is found. This means that we expand the (`N, S, M, P, B, L`)-node:

```
-   N, S, M, P, B, L, C. perf=0.210
        o   S, M, P, B, L, C. perf=0.190
        o   N, M, P, B, L, C. perf=0.212
        o   N, S, P, B, L, C. perf=0.189
        o   N, S, M, B, L, C. perf=0.213
        o   N, S, M, P, L, C. perf=0.067
        o   N, S, M, P, B, C. perf=0.203
        o   N, S, M, P, B, L. perf=0.214
                ▪   S, M, P, B, L. perf=0.210
                ▪   N, M, P, B, L. perf=0.218
                ▪   N, S, P, B, L. perf=0.206
                ▪   N, S, M, B, L. perf=0.212
                ▪   N, S, M, P, L. perf=0.219
                ▪   N, S, M, P, B. perf=0.198
```

And we expand the highest performing node again:

```
-   N, S, M, P, B, L, C. perf=0.210
        o   S, M, P, B, L, C. perf=0.190
        o   N, M, P, B, L, C. perf=0.212
        o   N, S, P, B, L, C. perf=0.189
        o   N, S, M, B, L, C. perf=0.213
        o   N, S, M, P, L, C. perf=0.067
        o   N, S, M, P, B, C. perf=0.203
        o   N, S, M, P, B, L. perf=0.214
                ▪   S, M, P, B, L. perf=0.210
                ▪   N, M, P, B, L. perf=0.218
                ▪   N, S, P, B, L. perf=0.206
                ▪   N, S, M, B, L. perf=0.212
                ▪   N, S, M, P, L. perf=0.219
                        •   S, M, P, L. perf=0.051
                        •   N, M, P, L. perf=0.204
                        •   N, S, P, L. perf=0.043
                        •   N, S, M, L. perf=0.170
                        •   N, S, M, P. perf=0.212
                ▪   N, S, M, P, B. perf=0.198
```

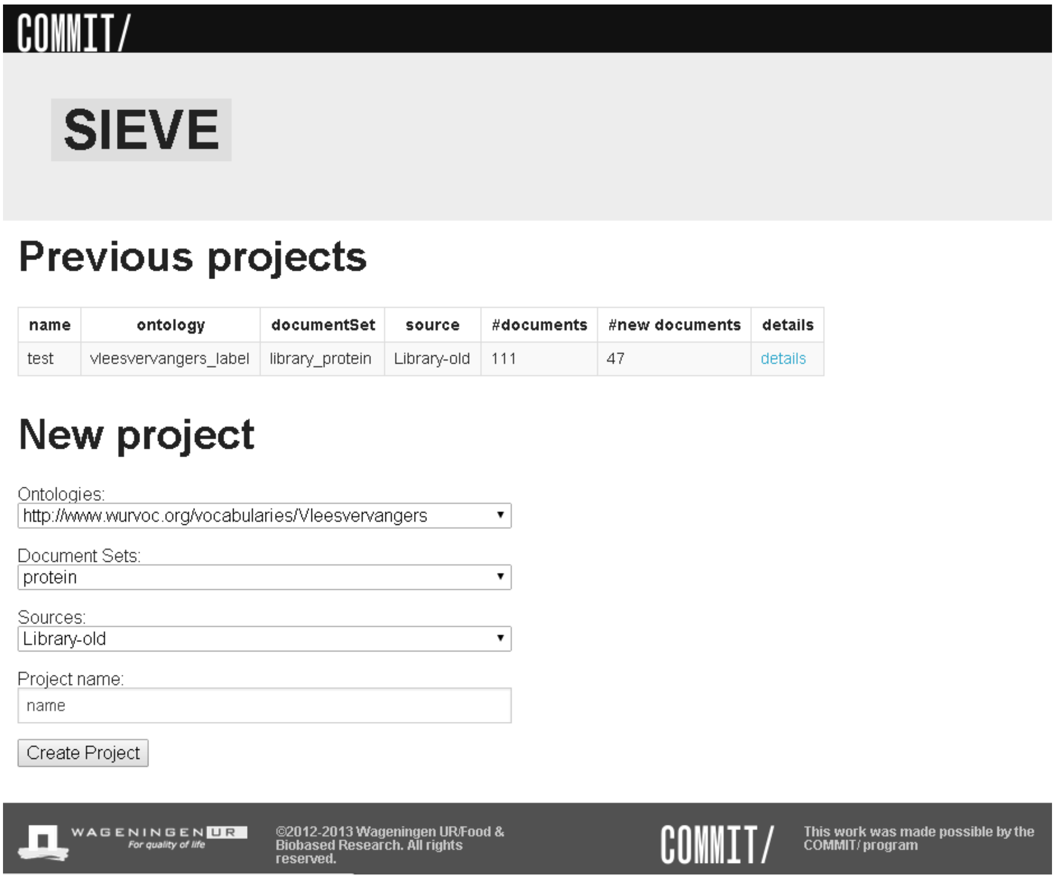
None of the children for the expanded node have a higher performance. So we stop expanding here and we check if the found node contributes to the set of queries we already have. In this case we find that this finds 20 known relevant documents of which 13 were not matched by previous queries. In addition it finds 75 new documents.

The algorithm continues and finds the maximum node that has not been expanded yet. It selects the (`N, M, P, B, L`)-node. But when it checks the number of new documents it finds that enough new documents have been selected by the queries and stops.

On a side node it is interesting to inspect the above tree. At the last expansion there are some large drops in performance for a few nodes (to 0.05 and 0.04), this is typical and indicates that these have lost some restrictive term and are now are returning far too many documents for them to be certainly relevant. For instance the node (`N, S, P, L`) returns 2101 documents. A total 50 of them are known to be relevant, but this is only a small percentage and the node is therefore penalized. From this we learn that removing some terms such as nutrient or meat alternative will make the query so generic that is matches all kind of other documents.

3 Implementation

We have implemented the above method in a tool called SIEVE, which is intended for use by domain experts. In this section we describe the features of SIEVE by displaying the different screens and describing what information the users see and what actions they can perform.



Previous projects

name	ontology	documentSet	source	#documents	#new documents	details
test	vleesvervangers_label	library_protein	Library-old	111	47	details

New project

Ontologies: http://www.wurvoc.org/vocabularies/Vleesvervangers

Document Sets: protein

Sources: Library-old

Project name: name

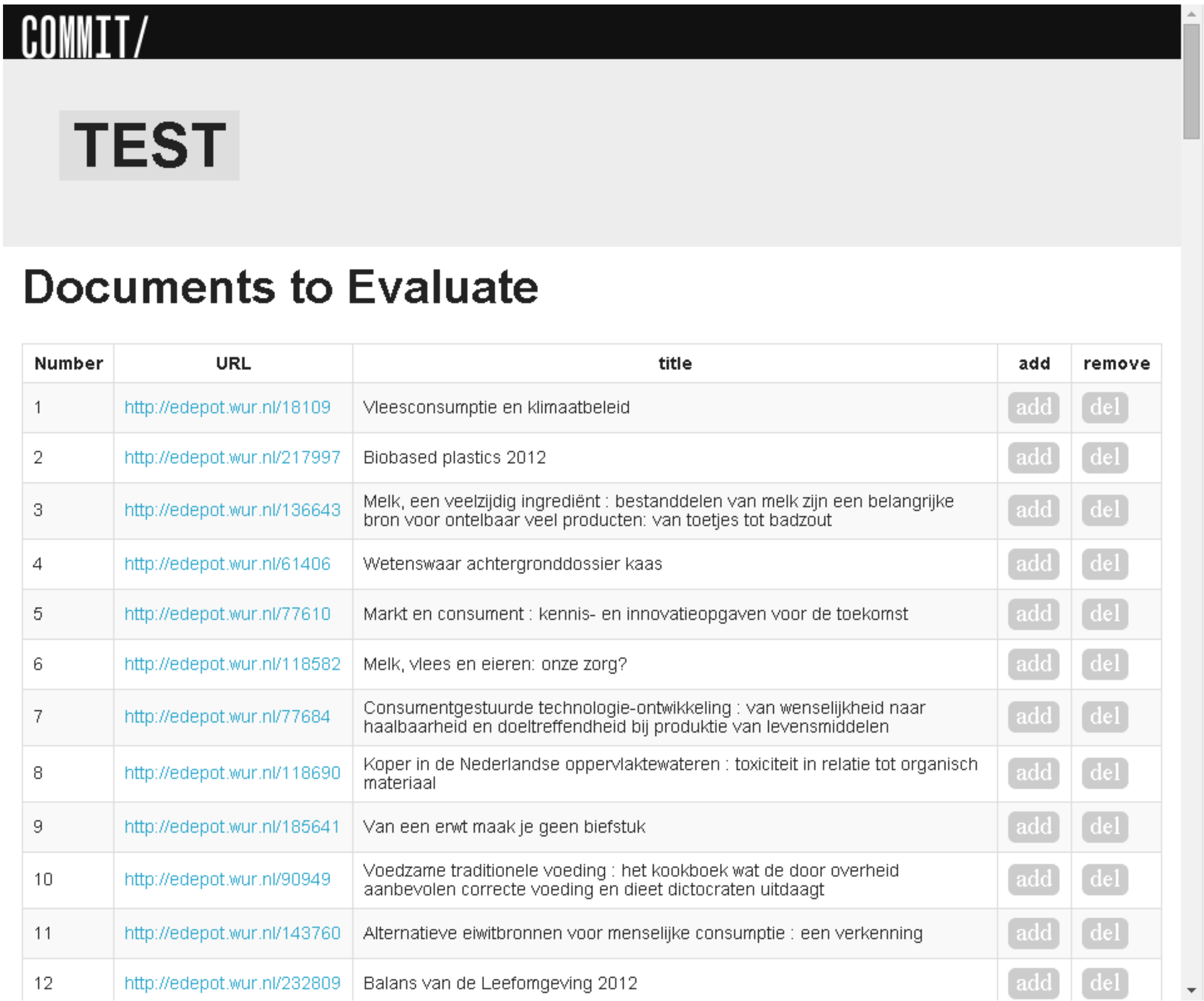
©2012-2013 Wageningen UR/Food & Biobased Research. All rights reserved.

This work was made possible by the COMMIT/ program

Figure 5: Main page, here users can manage their projects

The main page in Figure 5 shows an overview of the current projects including the number of relevant documents found and the number of suggested relevant documents. The link ‘details' in Figure 5 brings the user to a page on which the user can see four sections for the selected project, which are described in the following figures.

- Documents to Evaluate: new documents that SIEVE has found. The user can mark these as relevant or irrelevant. When all documents have been evaluated SIEVE presents the option to search for more documents. (Figure 6)
- All Documents: the documents which were marked as relevant or part of the initial run. (Figure 7)
- Rejected Documents: the documents which were marked as not relevant. (Figure 8)
- Configuration: contains the settings for this project (the ontology to be used, the document set and the source of documents).



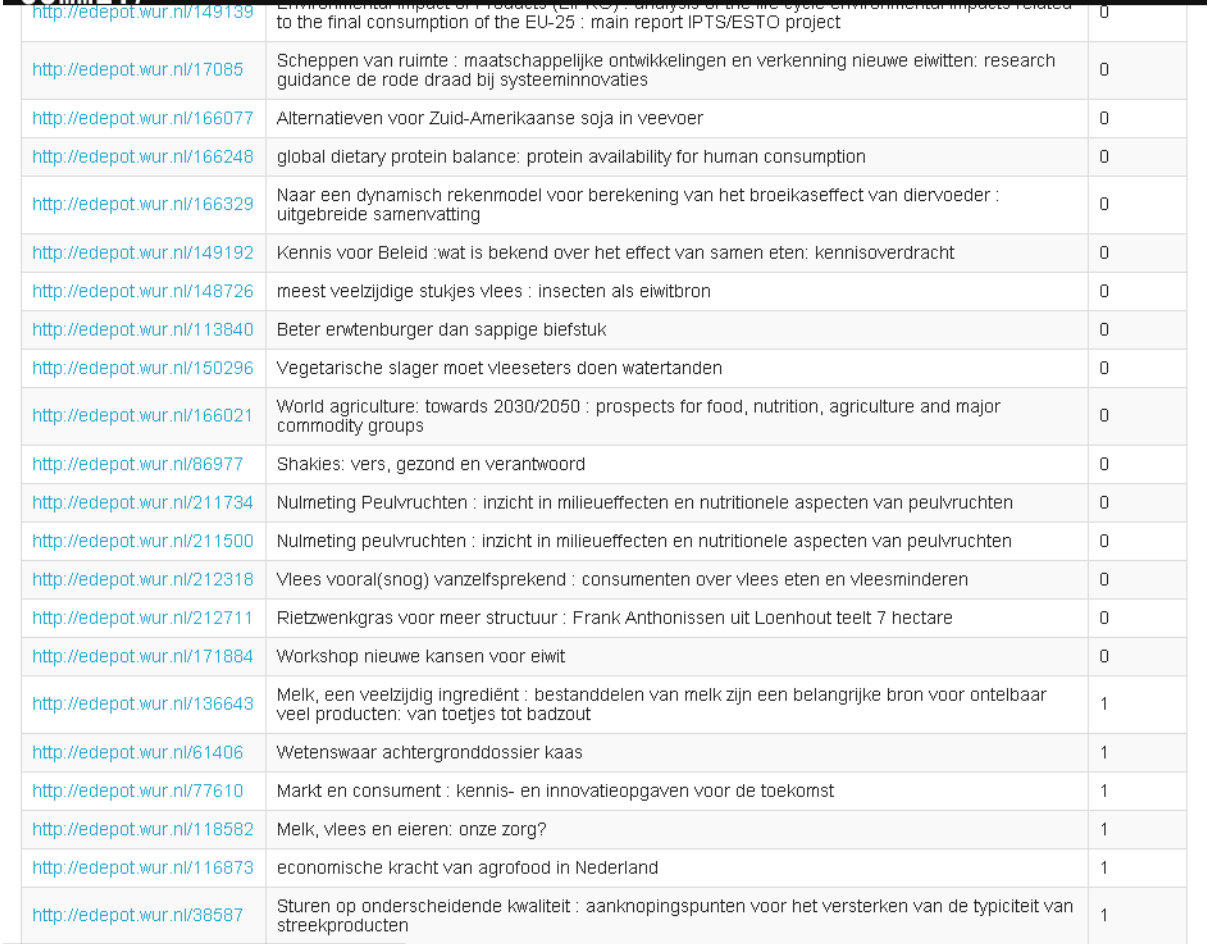
COMMIT/

TEST

Documents to Evaluate

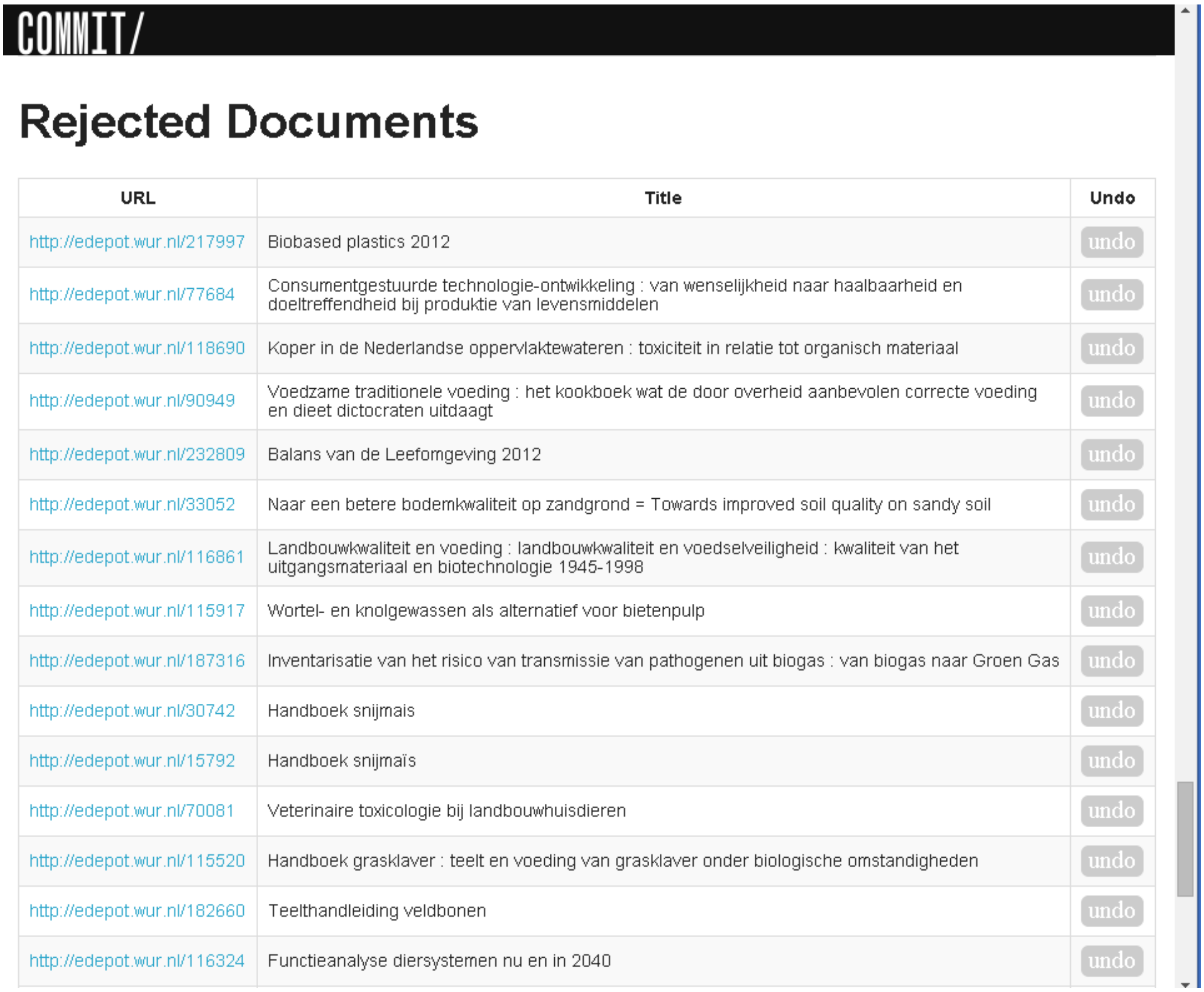
Number	URL	title	add	remove
1	http://edepot.wur.nl/18109	Vleesconsumptie en klimaatbeleid	add	del
2	http://edepot.wur.nl/217997	Biobased plastics 2012	add	del
3	http://edepot.wur.nl/136643	Melk, een veelzijdig ingrediënt : bestanddelen van melk zijn een belangrijke bron voor ontelbaar veel producten: van toetjes tot badzout	add	del
4	http://edepot.wur.nl/61406	Wetenswaar achtergronddossier kaas	add	del
5	http://edepot.wur.nl/77610	Markt en consument : kennis- en innovatieopgaven voor de toekomst	add	del
6	http://edepot.wur.nl/118582	Melk, vlees en eieren: onze zorg?	add	del
7	http://edepot.wur.nl/77684	Consumentgestuurde technologie-ontwikkeling : van wenselijkheid naar haalbaarheid en doeltreffendheid bij produktie van levensmiddelen	add	del
8	http://edepot.wur.nl/118690	Koper in de Nederlandse oppervlaktewateren : toxiciteit in relatie tot organisch materiaal	add	del
9	http://edepot.wur.nl/185641	Van een erwt maak je geen biefstuk	add	del
10	http://edepot.wur.nl/90949	Voedzame traditionele voeding : het kookboek wat de door overheid aanbevolen correcte voeding en dieet dictocraten uitdaagt	add	del
11	http://edepot.wur.nl/143760	Alternatieve eiwitbronnen voor menselijke consumptie : een verkenning	add	del
12	http://edepot.wur.nl/232809	Balans van de Leefomgeving 2012	add	del

Figure 6: Documents to Evaluate, each document can be marked as relevant (add) or not relevant (remove).



http://edepot.wur.nl/149139	Environmental impact of Products (EIPRO) : analysis of the life cycle environmental impacts related to the final consumption of the EU-25 : main report IPTS/ESTO project	0
http://edepot.wur.nl/17085	Scheppen van ruimte : maatschappelijke ontwikkelingen en verkenning nieuwe eiwitten: research guidance de rode draad bij systeeminnovaties	0
http://edepot.wur.nl/166077	Alternatieven voor Zuid-Amerikaanse soja in veevoer	0
http://edepot.wur.nl/166248	global dietary protein balance: protein availability for human consumption	0
http://edepot.wur.nl/166329	Naar een dynamisch rekenmodel voor berekening van het broeikaseffect van diervoeder : uitgebreide samenvatting	0
http://edepot.wur.nl/149192	Kennis voor Beleid :wat is bekend over het effect van samen eten: kennisoverdracht	0
http://edepot.wur.nl/148726	meest veelzijdige stukjes vlees : insecten als eiwitbron	0
http://edepot.wur.nl/113840	Beter erwtenburger dan sappige biefstuk	0
http://edepot.wur.nl/150296	Vegetarische slager moet vleeseters doen watertanden	0
http://edepot.wur.nl/166021	World agriculture: towards 2030/2050 : prospects for food, nutrition, agriculture and major commodity groups	0
http://edepot.wur.nl/86977	Shakies: vers, gezond en verantwoord	0
http://edepot.wur.nl/211734	Nulmeting Peulvruchten : inzicht in milieueffecten en nutritionele aspecten van peulvruchten	0
http://edepot.wur.nl/211500	Nulmeting peulvruchten : inzicht in milieueffecten en nutritionele aspecten van peulvruchten	0
http://edepot.wur.nl/212318	Vlees vooral(snog) vanzelfsprekend : consumenten over vlees eten en vleesminderen	0
http://edepot.wur.nl/212711	Rietzwenkgras voor meer structuur : Frank Anthonissen uit Loenhout teelt 7 hectare	0
http://edepot.wur.nl/171884	Workshop nieuwe kansen voor eiwit	0
http://edepot.wur.nl/136643	Melk, een veelzijdig ingrediënt : bestanddelen van melk zijn een belangrijke bron voor ontelbaar veel producten: van toetjes tot badzout	1
http://edepot.wur.nl/61406	Wetenswaar achtergronddossier kaas	1
http://edepot.wur.nl/77610	Markt en consument : kennis- en innovatieopgaven voor de toekomst	1
http://edepot.wur.nl/118582	Melk, vlees en eieren: onze zorg?	1
http://edepot.wur.nl/116873	economische kracht van agrofood in Nederland	1
http://edepot.wur.nl/38587	Sturen op onderscheidende kwaliteit : aanknopingspunten voor het versterken van de typiciteit van streekproducten	1

Figure 7: This shows a few lines of the found documents. For each document it is indicated in which iteration it was added. The initial documents have a zero and those with one are found by SIEVE and added by the expert in the first iteration



Rejected Documents

URL	Title	Undo
http://edepot.wur.nl/217997	Biobased plastics 2012	undo
http://edepot.wur.nl/77684	Consumentgestuurde technologie-ontwikkeling : van wenselijkheid naar haalbaarheid en doeltreffendheid bij produktie van levensmiddelen	undo
http://edepot.wur.nl/118690	Koper in de Nederlandse oppervlaktewateren : toxiciteit in relatie tot organisch materiaal	undo
http://edepot.wur.nl/90949	Voedzame traditionele voeding : het kookboek wat de door overheid aanbevolen correcte voeding en dieet dictocraten uitdaagt	undo
http://edepot.wur.nl/232809	Balans van de Leefomgeving 2012	undo
http://edepot.wur.nl/33052	Naar een betere bodemkwaliteit op zandgrond = Towards improved soil quality on sandy soil	undo
http://edepot.wur.nl/116861	Landbouwkwaliteit en voeding : landbouwkwaliteit en voedselveiligheid : kwaliteit van het uitgangsmateriaal en biotechnologie 1945-1998	undo
http://edepot.wur.nl/115917	Wortel- en knolgewassen als alternatief voor bietenpulp	undo
http://edepot.wur.nl/187316	Inventarisatie van het risico van transmissie van pathogenen uit biogas : van biogas naar Groen Gas	undo
http://edepot.wur.nl/30742	Handboek snijmais	undo
http://edepot.wur.nl/15792	Handboek snijmaïs	undo
http://edepot.wur.nl/70081	Veterinaire toxicologie bij landbouwhuisdieren	undo
http://edepot.wur.nl/115520	Handboek grasklaver : teelt en voeding van grasklaver onder biologische omstandigheden	undo
http://edepot.wur.nl/182660	Teelthandleiding veldbonen	undo
http://edepot.wur.nl/116324	Functieanalyse diersystemen nu en in 2040	undo

Figure 8: The section for Rejected Documents displays the documents found by SIEVE which were rejected by the expert

4 Experiment

4.1 Input

The expert selected 109 documents as known relevant documents. And we constructed an ontology of which some top concepts are depicted in Figure 9.

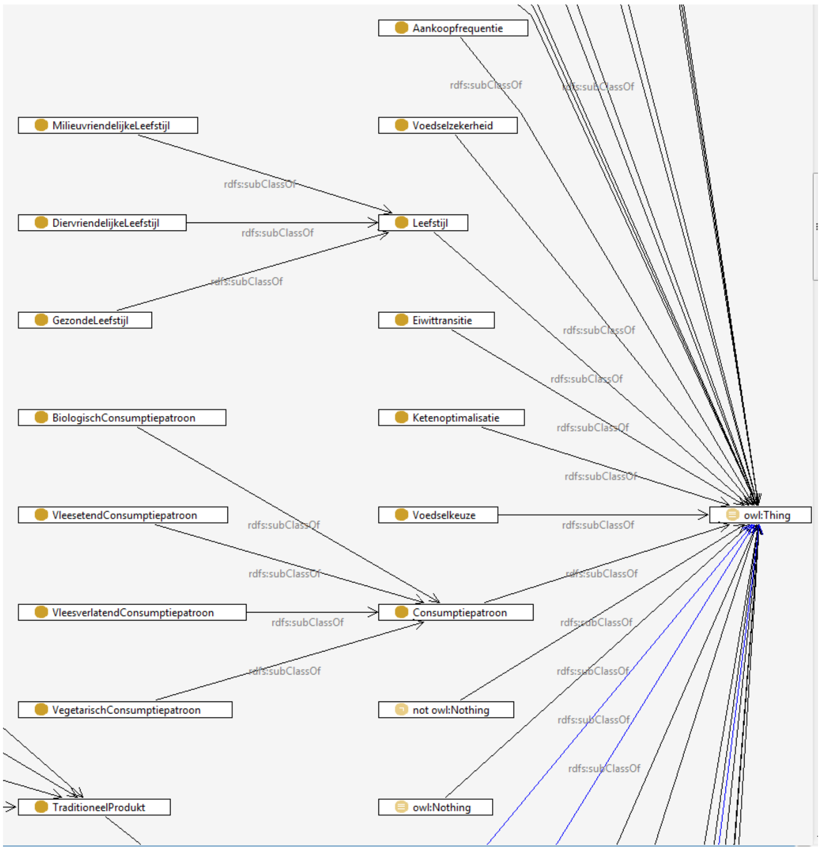


Fig 9. Some of the top concepts and some of their subclasses for the protein innovation domain

4.2 Result

When we applied the described system to the domain of protein innovations we got in the first run 106 new documents. They were gathered by three subqueries which matched 27 of the 109 known relevant documents. The three queries were:

- top concepts: nutrient, ingredient, meat alternative, product feature, brand, chain link. F1-measure = 0.21, 20 known relevant documents, 55 new documents.
- top concepts: ingredient, meat alternative, brand, chain link. F1-measure = 0.22, 22 known relevant documents, 55 new documents.
- top concepts: nutrient, ingredient, brand, chain link. F1-measure = 0.21, 25 known relevant documents, 96 new documents.

We asked our expert to determine the relevance of these 106 new documents and 36 of these were selected as relevant. This means that the set of known relevant documents was increased by 34%, where the expert only had to look at 106 documents.

In this experiment we have only a single iteration. Iteration is possible because of the reduced processing time gained by using query expansion and tree search. However, at this point we had no possibility to have an expert review the intermediate results. This will be part of future experiments, for example as part of an EU-project on agricultural knowledge for farmers.

5 Discussion and Future Work

Currently SIEVE is using the F1-measure to determine the performance of a query. This works as it is a representation of the need for both a high precision and recall. However, our actual goal is a maximal recall and a close but not exactly 100% precision. That will give us the query almost exactly matches the known relevant documents plus a few new documents. A new metric that describes this value would improve the method.

For the COMMIT/ project eFoodLab, this is part of a larger framework. The described tool uses a set of documents and an ontology to expand that set of documents. However, a good set of documents already contains a lot of the information required to build a good ontology. In our vision we see this as one direction in a two-way process in which we both improve the ontology and the set of documents. In a complete system the steps will be repeated with experts performing in-between checks of the ontology and the document set.

An improved ontology can also help the user to search or browse through the documents and even help the expert understand more about his domain. For instance, an expert might discover that he has forgotten about a subdomain of the domain in both the ontology and the documents because part of the new documents contain references to this subdomain.

Finally there are some possible improvements that could increase the relevance of the returned new documents. One way would be to make use of common techniques in document searching, such as using the relevant frequency of terms to determine the relevance instead of simply the presence of a term. In other COMMIT/ projects we have also worked on more advanced ways to determine the match of a document to an ontology.

References

1. Andrei Z. Broder, Steven C. Glassman, Mark S. Manasse, and Georey Zweig. Syntactic clustering of the web. *Computer Networks and ISDN Systems*, 29(8):1157-1166, 1997.
2. James W. Cooper, Anni R. Coden, and Eric W. Brown. Detecting similar documents using salient terms. In *Conference on Information and Knowledge Management: Proceedings of the eleventh international conference on Information and knowledge management*, volume 4, pages 245-251, 2002.

Acknowledgements

This report describes work within the eFoodLab project, part of COMMIT/.