A Model for Mining Relevant and Non-redundant Information

Laura Langohr
Department of Computer Science and
Helsinki Institute for Information Technology HIIT
University of Helsinki, Finland
laura.langohr@cs.helsinki.fi

Hannu Toivonen
Department of Computer Science and
Helsinki Institute for Information Technology HIIT
University of Helsinki, Finland
hannu.toivonen@cs.helsinki.fi

ABSTRACT

We propose a relatively simple yet powerful model for choosing relevant and non-redundant pieces of information. The model addresses data mining or information retrieval settings where relevance is measured with respect to a set of key or query objects, either specified by the user or obtained by a data mining step. The problem addressed is not only to identify other relevant objects, but also ensure that they are not related to possible negative query objects, and that they are not redundant with respect to each other.

The model proposed here only assumes a similarity or distance function for the objects. It has simple parameterization to allow for different behaviors with respect to query objects. We analyze the model and give two efficient, approximate methods. We illustrate and evaluate the proposed model on different applications: linguistics and social networks. The results indicate that the model and methods are useful in finding a relevant and non-redundant set of results.

While this area has been a popular topic of research, our contribution is to provide a simple, generic model that covers several related approaches while providing a systematic model for taking account of positive and negative query objects as well as non-redundancy of the output.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Relevance feedback , Retrieval models*

1. INTRODUCTION

Identification and retrieval of interesting objects is a classical but non-trivial task in data mining and information retrieval. We address a class of problems where one or more query objects has been identified, and the problem is to identify other objects that are relevant with respect to the query objects, but non-redundant with respect to each other. We build on the generic assumption that only a distance (or proximity) measure between objects is available, and we are not concerned whether it is based on similarities between objects or their attributes, similarities of their contexts, their co-occurrences, their relations, or something else. Consider the following examples.

Recommendation systems typically identify new products (e.g., songs) that are either similar to products currently liked by the user or, in collaborative recommendation systems, have been liked by similar users. At the same time, the set of recommendations should have variance. That is, the system should recommend k products, relevant to a given product and non-redundant to each other.

In a *text mining* setting, the user or a program might want to get an overview of different uses or contexts of given terms. For instance, given *root* as a query term, words *plant*, *equation*, and *word* constitute a representative set of terms that co-occur with *root* but represent different contexts (botany, mathematics, and linguistics, respectively). For the two query terms *branch* and *root* the terms *tree*, *mathematics*, and *languages* represent contexts in which both query terms occur. For instance, *trees* have *branches* and *roots*, other *plants* again might have only *roots*.

While relevance and redundancy have been addressed in numerous applications before (see Section 2 for related work), we are not aware of a general approach to find relevant and non-redundant objects based on distance alone. We propose and formulate the problem of identifying and retrieving a non-redundant set of relevant objects (Section 3) without restricting it to a specific application area.

We start by discussing distance (or proximity) based functions that define relevance with respect to one or more query objects. We then propose how to allow *negative query objects* in the definition of relevance, to specify which neighborhoods are less relevant. Noting that redundancy between objects can be based on a similar effect of repellance as that of negative query objects, we propose to treat these effects technically in the same way. The result is a relatively simple function that tries to find a balance between relevance with respect to positive query objects, avoidance of neighborhoods of negative query objects, and mutual non-redundancy of objects in the result.

We show that the problem of finding a non-redundant set of relevant objects is submodular and propose simple algorithms for it in Section 4. In Section 5 we report on preliminary experimental results on word relations and senses and with article co-authorships. In Section 6 we conclude with some notes about the results and future work.

SAC'12 March 25-29, 2012, Riva del Garda, Italy.

2. RELATED WORK

Identifying a set of relevant objects (typically documents) is a classical problem in information retrieval (IR). In typical settings, the selection is primarily based on the information contents of objects. Our problem differs from the main body of IR literature in the following aspects. (1) In our work the objects are not assumed to have attributes or other content. (2) Relevance is based solely on a given proximity function. (3) Queries are specified by objects, not by keywords. We next briefly review some previous work related especially to negative query terms and finding a non-redundant, representative set of results.

Negative query terms and redundancy are well-known in IR. Often, documents containing a negative query term are simply ignored [5], or they are assumed to be least interesting for the user [8]. IR measures that take into account both relevance and redundancy include negative feedback [19] and mixture modeling [20].

Carbonell and Goldstein [1] propose an incremental retrieval method, similar to ours: incrementally find a document that has high relevance to the query, but contains minimal similarity to previously selected documents. Several subsequent approaches are based on "information nuggets" of documents [2, 9], on cumulative gain measures [7, 2], on multiple application specific measures [6], or learn the ranking from diverse orderings [17] to mention only a few.

The problem of finding non-redundant or representative objects has been addressed in numerous other contexts, too. For instance, Lappas et al. [11] use social graphs to find a subset of experts (individuals) who answer the skill requests and can collaborate with each other. Other applications use representatives to eliminate irrelevant and redundant examples in databases to be analyzed by data mining algorithms [18, 15]. Clustering is often used to find representatives, see, e.g. [4, 21, 10].

The model we propose differs from the previous work by providing relatively simple but very flexible and generic measures for finding relevant but non-redundant objects. Our model only relies on a distance or proximity function between objects, and does not assume any other contents or properties for the objects. Potential targets thus range from mining or retrieving atomic concepts to documents and other complex structures whose distance is based on their contents. Rather than proposing complex and specific techniques, we aim to make a step towards a more generic model that covers and unifies some of the previous work. In particular, it shows how to systematically extend relevance to cover negative query objects and non-redundancy.

3. THE MODEL

In order to identify relevant and non-redundant objects, we need to be able to quantify relevance with respect to given query objects, as well as mutual non-redundancy between objects. In this section, we formalize these concepts.

Let V be a set of objects and $d : V \times V \to \mathbb{R}^+$ be a distance measure for objects in V. Alternatively, a proximity (i.e. similarity) function $s : V \times V \to \mathbb{R}^+$ can be given. We will assume that either one is given and identify the other one simply with the inverse $s(u, v) = 1/d(u, v)$ for all $u, v \in V$, except that $s(u, u) = \infty$. We assume that the proximity and distance functions are positive and symmetric. We then define relevance, irrelevance, and non-redundancy as follows.

Relevance. The *relevance* of an object $u \in V$ with respect to a positive query object $q \in V$ is defined directly as their proximity:

$$rel_P(u, q) = s(u, q) = 1/d(u, q). \quad (1)$$

Given a set $Q_P \subset V$ of (positive) query objects, an object is usually considered to be more relevant if it is relevant to all query objects. For instance, in the branch-root example the concepts *tree*, *mathematics*, and *languages* are connected to both query terms, *branch* and *root*.

A flexible relevance function can be obtained from the Minkowski or p-norm distance $(\sum_{q \in Q_P} d(u, q)^p)^{1/p}$ [13]. As is well known, with $p = 1$, the p-norm distance is the sum of the distances, and with $p = \infty$, it is their maximum. In general, with larger values of p, larger distances dominate the function more.

Since the p-norm is a distance but we want to measure relevance, we define relevance of object u with respect to a set Q_P of query objects simply as the inverse of the p-norm:

$$rel_P(u, Q_P) = \Big(\sum_{q \in Q_P} d(u, q)^\alpha\Big)^{-\frac{1}{\alpha}} \quad (2)$$

where $\alpha \geq 1$. Equation 1 clearly is a special case of this definition when $Q_P = \{q\}$.

For the sake of illustration, consider a set V of points on a plane and the Euclidean distance $d(u, v)$ between points. Different behaviors of relevance for different values of α are shown in Figure 1 (a)–(d). In Panel (a), with $\alpha = 1$, the sum of distances to query points determines the relevance. As a result, all points on the line between the two query points have an equal, highest relevance. Panel (b) illustrates how $\alpha = 4$ emphasizes larger distances and, in effect, favors points that are more equally distant to both query points. Panels (c)–(d) show a similar effect for the case of three query points. There, relevance is centered already for a smaller value of α due to a larger number of query points.

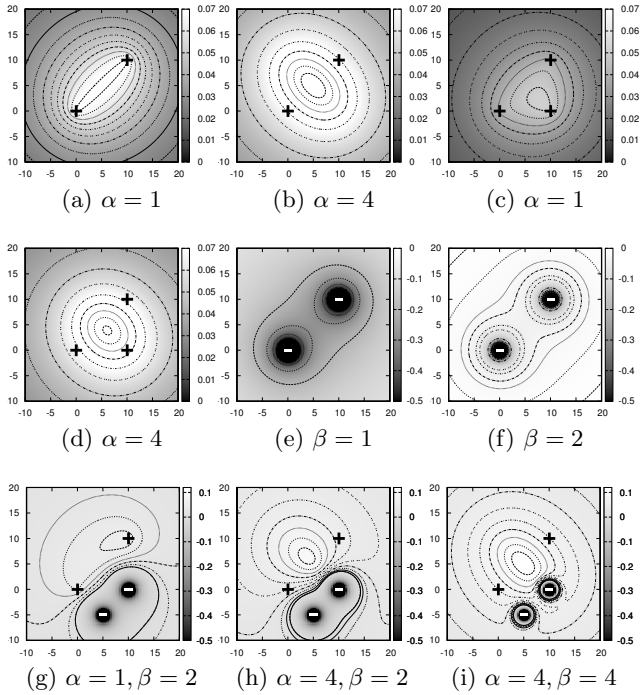
The definition of relevance (Equation 2) has some nice properties. It is monotone decreasing in the distance to each query object (with the exception of $\alpha = \infty$ when it is a function of the largest distance alone). Further, the formulation as a function of the set of distances guarantees certain simplicity as it rules out complex relevance functions that would depend on the inner structure of the set Q_P of positive query objects.

Irrelevance. In addition to positive query objects, we allow use of negative query objects to specify subjective irrelevance (uninterestingness) of objects. The negative relevance of object u with respect to a single negative query object $\bar{q}$ is measured with the given similarity or distance function, just like relevance to a single positive query object:

$$rel_N(u, \bar{q}) = s(u, \bar{q}) = 1/d(u, \bar{q}). \quad (3)$$

A negative query object's contribution is then $-rel_N(u, \bar{q})$.

Given a set $Q_N \subset V$ of negative query objects, the question next is how to define the total negative relevance with respect to this set. The situation is subtly different from positive query objects, where relevance of an object was defined to be highest when the object is relevant to *all* query points (as weighted by parameter α). The set of negative query objects are usually treated more as a disjunction: the result is irrelevant if it is close to *any* negative query object. Hence, p-norm would *not* be a good alternative here, as it would prefer objects centered between all negative query



(a) $\alpha = 1$ (b) $\alpha = 4$ (c) $\alpha = 1$

(d) $\alpha = 4$ (e) $\beta = 1$ (f) $\beta = 2$

(g) $\alpha = 1, \beta = 2$ (h) $\alpha = 4, \beta = 2$ (i) $\alpha = 4, \beta = 4$

Figure 1: Altitude profiles of (a)-(d) relevance (e)-(f) negative relevance, and (g)-(i) overall relevance for positive (pluses) and negative (minuses) query points on a plane. Lighter areas are more relevant. Thick contour lines depict overall relevances of zero.

objects. Consider again the branch-root example: Given *equation* and *plant* as negative query objects, *mathematics* and *tree* are quite irrelevant, but *languages* is not.

We base the definition of irrelevance on the sum of similarities, giving more weight to larger similarities, i.e., to more proximal negative query objects. To tune this weighting, each similarity is raised to the power of $\beta \geq 1$: the higher its value is, the more dominant are the most proximal points.

We thus define the negative relevance of object u with respect to a set $Q_N \subset V$ of negative query objects as

$$rel_N(u, Q_N) = \sum_{\bar{q} \in Q_N} d(u, \bar{q})^{-\beta} = \sum_{\bar{q} \in Q_N} s(u, \bar{q})^{\beta}. \quad (4)$$

Their contribution to overall relevance is then $-rel_n(u, Q_N)$.

For an illustration of the effects of negative query points, consider again a set V of points on a plane. Figure 1 (e)–(f) show the situation for two negative query objects and two different values of β. In both cases, the effects of negative query objects are concentrated locally around them. Increasing the value of β from 1 to 2 increases the concentration quite clearly.

The irrelevance function has desirable properties, too. It is zero if there are no negative query points, the effect of a negative query point infinitely far away is zero, and the function is monotonically decreasing in each distance.

Relevance vs. irrelevance. Given positive and negative query objects (sets Q_P and Q_N, respectively), the total relevance of object u is defined as

$$REL(u, Q_P, Q_N) = rel_p(u, Q_P) - rel_N(u, Q_N). \quad (5)$$

It favors objects that are centered between the positive query objects and that are not close to any negative one. Given a single positive and single negative query point, it simply measures which one is closer.

Figure 1 (g)–(i) illustrate the combined effect of two positive and two negative query points on a plane. In Panels (g)–(h), α grows from 1 to 4, resulting in an increasing emphasis on points more equally distant to both positive query points. In Panels (h)–(i) β grows from 2 to 4. With smaller values of β, the most relevant area is not anymore exactly between the positive query points, but is pushed away by the negative query points. When β is increased, the effect decreases.

Non-redundancy. The most relevant objects, as defined above, can be close neighbors. While we want to retrieve a list of relevant objects, we also want them to be mutually non-redundant or complementary.

This desirable effect is similar to negative query objects: two objects close to each other are mutually redundant, just like an object close to a negative query object is irrelevant. Consequently, we define redundancy in a similar way that we defined negative relevance.

The *redundancy* of a set $R \subseteq V$ of objects is defined by

$$red(R) = \sum_{\substack{u,v \in R \\ u \neq v}} d(u, v)^{-\beta} = \sum_{\substack{u,v \in R \\ u \neq v}} s(u, v)^{\beta}, \quad (6)$$

where $\beta \geq 1$. Redundancy will also contribute negatively to the overall relevance of a set, i.e., by $-red(R)$.

A Relevant and Non-redundant Set of Objects. The overall goal is to find a diverse set of relevant objects according to the user's query. Using the definitions above, we define the overall relevance and non-redundancy of a set of (retrieved) objects $R \subseteq V$ as

$$REL(R, Q_P, Q_N) = \sum_{u \in R} rel_P(u, Q_P) - rel_N(u, Q_N) - red(R). \quad (7)$$

We can now present the problem formally. In addition to the positive and negative query objects, assume the user also specifies the number k of objects in the output, and (for practical convenience) a set $V' \subseteq V$ of objects among which to select the output.

Problem definition. Given a set V of objects, a distance function $d(u, v)$ or a proximity function $s(u, v)$ for objects $u, v \in V$, a set $Q_P \subset V$ of positive query objects, a set $Q_N \subset V$ of negative query objects, a target set $V' \subseteq V$, and an integer k, the *problem of retrieving a relevant and non-redundant set of objects* is to identify a set $R \subseteq V'$ of size $|R| = k$ that maximizes $REL(R, Q_P, Q_N)$.

4. ALGORITHMS

We next present algorithms to find relevant and non-redundant objects. We first analyze the overall relevance and non-redundancy function. We then give two methods, one that greedily produces a ranked list, and one that optimizes the result for a fixed number of objects in the output.

4.1 Problem Analysis

Submodularity. Well-known approximation results exist for submodular functions. The overall relevance $REL(\cdot)$ (Equation 7) obviously is submodular since it satisfies the following diminishing returns property: the marginal gain of

adding an object to a set A of objects is at least as big as adding it to any of its supersets $B \supseteq A$.

Theorem 1. The overall relevance of Equation 7 is *submodular* for all $A \subseteq B \subseteq V'$ and $x \in V' \setminus B$.

The proof is simple. Function $REL(\cdot)$ is submodular if

$$REL(A \cup \{x\}) - REL(A) \geq REL(B \cup \{x\}) - REL(B).$$

This clearly holds since the marginal change is

$$\begin{aligned} REL(A \cup \{x\}, Q_P, Q_N) \\ -REL(A, Q_P, Q_N) \quad &= \quad rel_p(x, Q_P) \\ &\quad -rel_N(x, Q_N) - rel_N(x, A) \end{aligned} \tag{8}$$

and $rel_N(x, A) \leq rel_N(x, B)$ if $A \subseteq B$.

For a submodular function, a greedy algorithm is guaranteed to find a set which achieves at least $1/k$ of the optimal score [14]. Unfortunately, the overall relevance $REL(\cdot)$ is neither nondecreasing (the marginal change is either positive or negative) nor non-negative. If it was, tighter bounds would apply [14, 3].

Maximizing a submodular function that is neither nondecreasing nor non-negative with an approach that optimizes the result for a fixed number k incrementally can be bound if one can bound the marginal change [14]. However, the marginal change of the overall relevance is unbounded, since the negative relevance (or redundancy) approaches infinity as the distance to a negative query object (or another object in the result) approaches zero.

Negative relevance and redundancy. A greedy algorithm for the problem will iteratively choose object u to maximize the marginal change of Equation 8, where A is the set of already chosen relevant objects. As the equation suggests, negative query objects and already selected objects A can be treated uniformly, and the marginal change is then

$$\begin{aligned} REL(A \cup \{u\}, Q_P, Q_N) \\ -REL(A, Q_P, Q_N) \quad &= \quad rel_p(u, Q_P) \\ &\quad -rel_N(u, Q_N \cup A). \end{aligned} \tag{9}$$

This helps make the greedy algorithm very simple as will be seen next.

4.2 Greedy Algorithm

The greedy algorithm produces a ranked list of objects in an incremental, greedy fashion with respect to the overall relevance $REL(u, Q_P, Q_N)$. In each iteration, it finds the currently most relevant object and outputs it. Based on Equation 9, the objects can be simply added to the negative query points as they are selected. As a result, the ith object output is non-redundant with respect to first $i-1$ objects already output.

Greedy algorithm
1. Repeat until a sufficient number of representatives has been retrieved:
   1.1 Find the most relevant object r w.r.t. Q_P and Q_N
   1.2 Output r and add it to Q_N

As an example, consider points on a plane, two positive and one negative query point. Figure 2 illustrates how the greedy algorithm incrementally picks points from the plane.

The loop of the greedy algorithm is iterated $O(k) = O(|R|)$ times. The most complex task in the loop is the identification of the next most relevant object. If distances

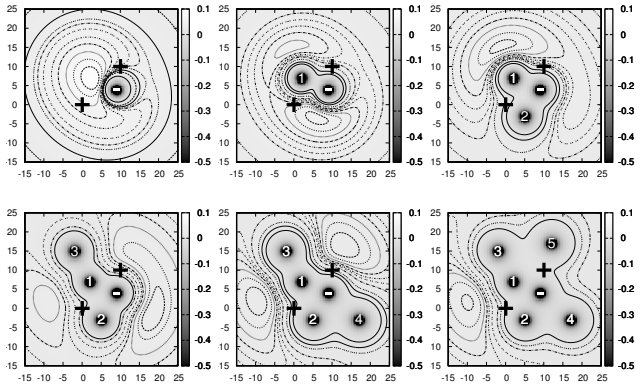


Figure 2: The greedy algorithm applied on points on the plane, with $\alpha = 4$, $\beta = 2$, two positive (plusses), and one negative (minus) query point. Initial situation and after one to five representative points (denoted by digits in their output order).

are known (or computed in constant time), then computing the relevances and finding the most relevant one(s) can be done in time $O(|V'|)$. Hence, the total time complexity is $O(k|V'|) = O(|V'|)$.

4.3 Iterative Algorithm

Even though the greedy algorithm makes the best possible choice with respect to $REL(u, Q_P, Q_N)$ in each step, the set of top k objects is not necessarily optimal for any k except $k = 1$. The iterative algorithm, in turn, produces a non-redundant set of k relevant objects, where k is given. The algorithm takes k objects as input, used as an initial solution that is then iteratively improved. In each iteration, the algorithm takes one of the k objects and replaces it by the optimal one, given the $k-1$ other current objects. When no improvements can be achieved, the algorithm stops.

Iterative algorithm
1. Get an initial solution R of k objects (e.g. random)
2. Repeat while R changes:
   2.1 Find the optimal swap of any object r in R to any object not in R
   2.2 If the swap improves the result, implement it

The iterative algorithm stops when it converges, but it may converge to a local optimum. The algorithm as such is deterministic (except when there are ties) so the initial solution R clearly could have an important effect on the quality of the result. We therefore propose the following alternatives to initializing it: (1) Run the greedy algorithm first (for k iterations at least) and then use the top k objects from it as the initial solution to the iterative algorithm. (2) Give k random objects as the initial solution. Optionally run the iterative algorithm several times with different random seeds and choose the result that maximizes $REL(R, Q_P, Q_N)$.

The algorithm is guaranteed to stop assuming that V' is finite: the number of possible configurations of k objects is finite, and since the solution is changed only if it is improved, the algorithm never returns to a previous solution. Unfortunately the number of possible solutions is exponential.

5. EXPERIMENTS

We will first illustrate the concepts and methods in a linguistic setting, analysing relations between words. We then study co-authorship relations between computer scientists, looking for researchers related to *C. Faloutsos* and *J. Han*.

5.1 Word Relations and Senses

Test Setting. In the first setting, objects are English words and their proximity is measured by co-occurrence statistics in a corpus. The goal will be to test how the proposed model manages to separate different uses (or senses or contexts) of a given word. The corpus in our test is 2008/9 Wikipedia Selection for schools[1], a collection of about 5500 wikipedia articles of 20 million words in total. The text was processed using standard techniques (see, e.g., [12]): lowercasing, removing common words (so called stop words) and punctuation, and lemmatizing (transforming words to their canonical forms). Statistics of word co-occurrences within sentences were evaluated using a multinomial model and the log-likelihood ratio test.

Results. Reliable systematic testing of how well relevant and non-redundant words represent different senses is difficult, so we present some illustrative results in Table 1.

The two most relevant non-redundant words associated to *bank*, for instance, are *reserve* (which corresponds to sense #5 of bank in the WordNet[2] dictionary: "a supply or stock held in reserve for future use"), and *river* (sense #1: "sloping land [...] beside a body of water"). The third most relevant word is *gaza*, as in Gaza Strip, which occurs in the specific context of the West Bank of the Jordan river. The fourth most relevant word is *credit* (sense #2: "a credit card processing bank"). The fifth most relevant word is *international*, which does not correspond to any WordNet sense of bank (or banking), but is highly ranked, because it occurs often in the corpus in phrases like "international banking".

Table 1: Top five words ranked as relevant and non-redundant by the greedy algorithm for $\alpha = 4$, $\beta = 2$ and different words and a word pair.

bank	star	root	branch, root
reserve	planet	plant	tree
river	trek	equation	indo
gaza	cluster	word	mathematics
credit	sirius	irrationality	line
international	movie	unity	equation

For *star*, we observe several relevant words from the astronomical context, but also one name (Star Trek) and the sense of being a celebrity or movie star. For *root*, the three first relevant words represent different WordNet senses, but the other two relate to the earlier ones.

For *branch* and *root* as the positive query terms, the three first relevant words again represent different contexts: botanics (*tree*), *mathematics*, and languages (*indo*). The other two terms relate to the context of *mathematics* as well.

The purpose of these results is not to shed new light on word sense disambiguation or related tasks. Rather, the aim is to illustrate that the generic model is able to perform a non-trivial task without being specifically tuned for it.

[1]http://schools-wikipedia.org

[2]http://wordnet.princeton.edu/

Table 2: Eight most relevant and non-redundant authors (left) or just relevant authors (right column) for $Q_P = \{$*C. Faloutsos, J. Han*$\}$ $\alpha = 4$, and $\beta = 2$.

Rel. and non-redundancy		Relevance only	
P.S. Yu	IL, USA	P.S. Yu	IL, USA
D. Srivastava	NJ, USA	R.T. Ng	Canada
H.J. Zhang	China	S. Papadimitriou	NY, USA
Y. Tao	Hong Kong	L.V.S. Lakshmanan	Canada
C. Liu	WA, USA	H.V. Jagadish	MI, USA
B. Chin Ooi	Singapore	X. Yan	CA, USA
T.K. Sellis	Greece	J. Yang	OH, USA
J. Gao	IL, USA	W. Fan	NY, USA

5.2 Co-authorship Relations

Test Setting. Next, we used co-authorships extracted from DBLP[3] (Digital Bibliography & Library Project) of Oct 6th, 2010. We extracted a network of 20 authors and 45 co-authorships connecting *C. Faloutsos* and *J. Han* and used pairwise similarities proportional to a cumulative distribution function [16] in the range $[0, 1]$. Further, the proximity of two authors, especially when not co-authors, was defined using the best path between them, taking the product of pairwise similarities along the path as the final proximity.

Results. The top eight relevant and non-redundant authors obtained with the greedy algorithm are shown in the left column of Table 2. They are all prominent researchers that are relatively closely related to Faloutsos and Han by direct or indirect co-authorship relations. The first four of the chosen authors have never published together according to DBLP, so they are likely to represent different communities or areas relevant to Faloutsos and Han. The spread of the results is also illustrated by the fact that many of the first eight authors come from different countries.

In contrast, if redundancy is ignored and the computation is based only on relevance, a redundant set of authors is obtained (Table 2, right column). The eight most relevant authors are highly connected to each other in the co-authorships graph, and come from either the US or Canada.

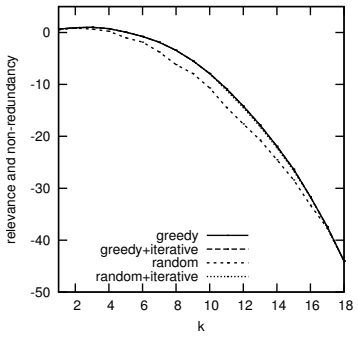


Figure 3: Overall relevance of set R_k of top k nodes obtained by different methods. (The lines for all non-trivial algorithms are indistinguishable, only random ranking differs from them.)

Comparison of algorithms. Let us next compare the algorithmic variants when a fixed number k of objects should be given as result. We compare four different approaches:

[3]http://dblp.uni-trier.de/

(1) finding relevant and non-redundant nodes with the greedy algorithm and taking the top k nodes, (2) finding them initially with the greedy algorithm and improving the results with the iterative algorithm, (3) picking k nodes randomly initially and improving the results with the iterative algorithm, and (4) simply picking k nodes randomly.

Figure 3 shows a comparison of the four algorithmic variants. The three first ones, using the greedy and iterative algorithms, are practically indistinguishable while the random results are systematically inferior. This indicates that the result of the greedy algorithm is, in addition to being a ranking of the nodes, also a good choice for any given k. Another observation is that the iterative algorithm performed equally well with random initialization as it does with initial ranking obtained by the greedy algorithm.

6. CONCLUSION

This paper is a step towards a generic approach to problems where a non-redundant set of relevant objects should be found, given positive and negative query objects and a distance measure. We based our definitions of relevance, irrelevance and non-redundancy only on object distance or proximity. We analyzed the problem and gave two algorithms: one that greedily ranks a given set of objects, and another one for finding an optimal set of objects when the size of the set is fixed. We performed experiments with real data from linguistics and co-authorship, to illustrate the setting and the behavior of the methods. Based on the results, both algorithms seem to produce a good set of objects. An interesting result is that the algorithm that produces a ranking seems also to work well in practice for any top k objects.

This work is preliminary in several aspects and at least the following aspects should be addressed in future work. (1) A deeper analysis of the problem and its properties is needed. For theoretical guarantees, it would be nice to have a nondecreasing, nonnegative relevance function. (2) The proposed algorithms are simple but efficient if the proximity measure $s(\cdot)$ is readily available. For more complex and larger cases, faster methods would be useful. (3) The current experiments are a proof of concept and show great promise, but more experimentation is needed to understand the practical behaviour of the methods and parameters. (4) It would be interesting to adapt and apply the approach to different applications, e.g., networks and probabilistic data.

7. ACKNOWLEDGMENTS

This work has been supported by the Algorithmic Data Analysis (Algodan) Centre of Excellence of the Academy of Finland and by the European Commission under the 7th Framework Programme FP7-ICT-2007-C FET-Open, contract no. BISON-211898. We would like to thank Petteri Hintsanen and Atte Hinkka for the DBLP implementation, and Oskar Gross for his help with the word relations test data and WordNet implementation.

8. REFERENCES

[1] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR '98*, pages 335–336, 1998.

[2] C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR '08*, pages 659–666, 2008.

[3] U. Feige, V. S. Mirrokni, and J. Vondrák. Maximizing non-monotone submodular functions. In *FOCS '07*, pages 461–471, 2007.

[4] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[5] F. C. Gey, A. Chen, J. He, L. Xu, and J. Meggs. Term importance. Boolean conjunct training, negative terms, and foreign language retrieval: Probabilists algorithms at TREC-5. In *TREC-5*, 1996.

[6] S. Gollapudi and A. Sharma. An axiomatic approach for result diversification. In *WWW '09*, pages 381–390, 2009.

[7] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20:422–446, 2002.

[8] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Core algorithms in the CLEVER system. *ACM Transactions on Internet Technology*, 6(2):131–152, 2006.

[9] A. Lad and Y. Yang. Learning to rank relevant and novel documents through user feedback. In *CIKM '10*, pages 469–478, 2010.

[10] L. Langohr and T. Toivonen. Finding representative nodes in probabilistic graphs. In *WEIN at ECML PKDD '09*, pages 65–76, 2009.

[11] T. Lappas, K. Liu, and E. Terzi. Finding a team of experts in social networks. In *KDD '09*, pages 467–476, 2009.

[12] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[13] C. D. Meyer. *Matrix Ananlysis and Applied Linear Algebra*. SIAM, 2000.

[14] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions — I. *Mathematical Programming*, 14:265–294, 1978.

[15] F. Pan, W. Wang, A. K. H. Tung, and J. Yang. Finding representative set from massive data. In *ICDM '05*, pages 338–345, 2005.

[16] M. Potamias, F. Bonchi, A. Gionis, and G. Kollios. k-nearest neighbors in uncertain graphs. In *VLDB '10*, 2010.

[17] F. Radlinski, R. Kleinberg, and T. Joachims. Learning diverse rankings with multi-armed bandits. In *ICML '08*, pages 784–791, 2008.

[18] J. C. Riquelme, J. S. Aguilar-Ruiz, and M. Toro. Finding representative patterns with ordered projections. *Pattern Recognition*, 36(4):1009–1018, 2003.

[19] X. Wang, H. Fang, and C. X. Zhai. A study of methods for negative relevance feedback. In *SIGIR '08*, pages 219–226, 2008.

[20] Z. Xu and R. Akella. Active relevance feedback for difficult queries. In *CIKM '08*, pages 459–468, 2008.

[21] D. Yan, L. Huang, and M. I. Jordan. Fast approximate spectral clustering. In *KDD '09*, pages 907–916, 2009.