A Network-Based Model for High-Dimensional Information Filtering

Nikolaos Nanas
Centre for Research and Technology Thessaly
Volos, 58300, Greece
n.nanas@cereteth.gr

Manolis Vavalis
Centre for Research and Technology Thessaly
Volos, 58300, Greece
m.vavalis@cereteth.gr

Anne De Roeck
Computing Department
The Open University
Milton Keynes, MKA 6AA, U.K.
a.deroeck@open.ac.uk

ABSTRACT

The Vector Space Model has been and to a great extent still is the de facto choice for profile representation in content-based Information Filtering. However, user profiles represented as weighted keyword vectors have inherent dimensionality problems. As the number of profile keywords increases, the vector representation becomes ambiguous, due to the exponential increase in the volume of the vector space and in the number of possible keyword combinations. We argue that the complexity and dynamics of Information Filtering require user profile representations which are resilient and resistant to this "curse of dimensionality". A user profile has to be able to incorporate many features and to adapt to a variety of interest changes. We propose an alternative, network-based profile representation that meets these challenging requirements. Experiments show that the network profile representation can more effectively capture additional information about a user's interests and thus achieve significant performance improvements over a vector-based representation comprising the same weighted keywords.

Categories and Subject Descriptors

H.3.4 [**Information Storage and Retrieval**]: Systems and Software

General Terms

Algorithms, Experimentation, Performance

Keywords

Content-based Information Filtering, User Profiling, Curse of Dimensionality

SIGIR'10, July 19–23, 2010, Geneva, Switzerland.

1. INTRODUCTION

When today on the WWW everyone can be both a consumer, but also a producer of information, a dual information overload problem arises. On one hand, it is impossible to keep track of the information that is being dynamically generated and disseminated in the context of the so called *real-time* Web, or to spot interesting sources of information out of the available glut. On the other hand, nobody can ensure the individual publisher that broadcasted information will reach the right audience. Information Filtering (IF) and the personalisation of information delivery that it achieves, can have a radical impact on the way we interact with information media. Already, Collaborative Filtering (CF) has been successfully deployed for calculating recommendations of movies, music tracks and books, but is admittedly not well suited for dynamic domains, like news publishing. Unlike CF, content-based IF has not yet produced similar success stories. After two decades of research on content-based IF, there is a surprising lack of publicly available and broadly adopted content-based IF applications.

Some of the reasons for this absence are discussed in [10], where we argued that IF is a complex and dynamic problem with its own particular characteristics and requirements, which differentiate it from Information Retrieval and Text Classification. IF is complex and dynamic because user interests and the information environment are complex and dynamic. Unlike Text Classification, the notion of a "topic" of interest is not that distinct in the case of IF. A user is typically interested in a variety of topics, which are fluid and interrelated. Over time, the level of interest in each topic may vary, new topics of interest can emerge and a previously interesting topic may wane and even become obsolete. Furthermore, there is an immense variety of topics to choose from in the information space. From general topics of interest, such as news categories (e.g., economy, technology, etc.) to a "long-tail" of more personal and specific interests. Of course, the information space itself continuously changes with the new material dealing with new combinations of concepts, even new concepts, the development of new technologies, the occurrence of temporal events, etc. Successful IF requires a user profile representation that can capture the various topics of interest and can continuously adapt to interest drifts and changes in the information space.

One significant implication of the above specification is that a user profile has to be able to incorporate a large number of features. For example, if we focus on textual information, then a larger number of keywords is required to represent multiple topics of interest than to represent a

single topic. More keywords are also required when the topics are specific rather than general. But, as we will further discuss in the next section, the Vector Space Model (VSM), the most popular choice for profile representation in IF, has inherent problems when the number of keywords, i.e., the dimensions of the vector space, increases. This "curse of dimensionality" [3] has forced vector-based approaches to IF to adopt radical dimensionality reduction techniques and to approach the modelling of user interests by requiring a separate profile for each topic. Furthermore, the VSM is typically coupled with the "bag of words" assumption and hence any information encoded in the correlated placement of words in text is not captured.

In this paper, we propose an alternative to the VSM. The user profile is no longer a weighted feature vector but a weighted network of descriptive features, which has been assigned to, or can be automatically extracted from interesting information items. Links in this network represent correlations between features appearing within the same context. Relevance evaluation of information items is not performed with trigonometric measures of similarity between keyword vectors, but with a directional spreading activation process. In the case of textual information, the user profile is a weighted network of keywords extracted from the content of interesting documents. We argue and experimentally support that the proposed network-based profile can incorporate a large number of keywords, more effectively than a vector-based profile. In doing so, the network-based profile captures additional information about a user's multiple interests, it becomes more specific and achieves significant performance improvements. We substantiate this claim in section 4. This resistance to the "curse of dimensionality" is a significant property of the proposed profile representation, offering many practical advantages and new perspectives.

In the rest of this paper we first identify and discuss the causes for the inherent dimensionality problems of vector-based approaches to IF. Then in section 3 we describe the proposed network-based representation. The experiments in section 4 have been performed with a methodology that adopts the Reuters-21578 and simulates users with multiple topics of interest. We compared a vector-based to a network-based profile comprising the same weighted keywords. The results indicate that as the number of keywords in these profiles increases, the existence of links in the network profile contributes to an increase in performance of up to 50% on average, compared to the vector-based profile. We discuss the implications of these positive results and we conclude with a summary and future research plans in section 5.

2. DIMENSIONALITY PROBLEMS IN IF

The VSM [15] is the most popular choice for profile representation and has had fundamental impact for research in IF. According to the VSM both documents and profiles are represented as, typically weighted, keyword vectors in a multidimensional space with as many dimensions as the number of keywords in the documents' vocabulary. This abstraction allows the application of trigonometric measures of similarity, like the inner product or cosine similarity, for assessing how "close" to a user profile a given document is [6]. The profile's goal is to define decision boundaries between relevant and non-relevant documents, or represent regions in the vector space which are dense with interesting documents. In multidimensional spaces however, the discriminatory power of pairwise distances is significantly affected.

In [7], the authors thoroughly analyse and discuss in the context of Artificial Immune Systems, the issues that undermine vector-based approaches in multidimensional spaces. Their argumentation is directly applicable to vector-based approaches to IF and so here we recapitulate some basic points:

- As the number of dimensions increases, the volume of the space increases exponentially faster, and distance based metrics become increasingly meaningless, because all points in such a space become essentially equidistant [1].
- In a multidimensional space small amounts of noise along many dimensions can cause significant displacement to a vector. Along the same lines, "A scalar metric cannot differentiate between two vectors that differ slightly across all dimensions (and may be the same after accounting for noise) or differ significantly on just a few dimensions (and are clearly different in some respect)" [7].
- When instance-based methods (such as kNN [2]) are deployed, then typically the number of required data points scales with the number of dimensions causing a significant increase in memory and time requirements. These problems are exaggerated in the case of continuous learning problems, such as IF, where there is no stopping criterion.

We would like to complement the above issues with an intuitive argument. Let's assume that a user is interested in two topics: "long river" and "bank holidays". If we use a keyword vector, containing these four words, then we equally represent any possible combination of these four words, including for instance the combinations, "river banks" and "long holidays", which do not necessarily represent topics of interest to the user. In general, as the number of keywords in a user profile increases, the number of possible combinations increases exponentially and the profile becomes ambiguous, because the majority of keyword combinations will be irrelevant to the user's interests. To counteract this effect one should at least avoid the common term-independence assumption, which is inherent to the VSM and its orthogonal dimensions, and move away from the "bag of words" simplification. In the opposite case, valuable information about the user's interests is lost and the profile's specificity drops. Our experimental results, clearly support this argument.

Although in this paper we concentrate on a static IF problem, we will briefly touch on dynamic aspects. As both the user's interests and the information space change over time, a fixed keyword space becomes an inadequate choice. Tackling the problem's dynamics requires a fluid keyword space where additional dimensions can be added and removed. But even if this is the case, the adoption of a common vector space where all profiles and documents are represented is still impractical. If an IF system needs to accommodate a large number of users, then it is only safe to assume that their interests will vary. To cover all possible topics of interest with a common vector space, a very large number of keywords and hence dimensions are required and the computational and memory costs would significantly increase. For instance, experiments performed in [9] demonstrate that

covering 23 of the topics in Reuters-21578 requires a common vector space comprising more than 30000 documents.

Although seldom clearly stated, the above dimensionality problems are evident in current vector-based IF practices. They heavily depend on dimensionality reduction techniques, such as stop word removal, stemming, term weighting [23], Latent Semantic Indexing (LSI) [5] and more. This pre-processing of documents takes place in advance and typically results in a *fixed* vector space with manageable dimensions. Furthermore, they tend to break up the problem into distinct pre-defined topics and built a separate single-topic profile for each individual topic [21, 22]. This practice has been inherited from Text Classification and is reflected by evaluation standards for IF [14]. In reality though, a user's topics of interest cannot be easily predefined and they are definitely not fixed. Even within a general topic (e.g., "economy") different users will develop specific interests in various subtopics (e.g., "credit card fraud", "equity markets" etc.). Furthermore, given the plenitude of documents (e.g., news stories, blog posts, etc.) being published daily on a topic, a user is only interested in and has the time to read a very small percentage. So specificity is essential for successful IF, but it requires profiles with the ability to capture any available information about a user's interests. Although outside the scope of the current work, we would also like to note that many machine learning algorithms, such as Rocchio's linear learning algorithm, which have been a popular solution for profile adaptation [17, 24], lack an inherent mechanism for adding or removing keywords to a user profile. This means that they assume a fixed vector space that predefines the possible repertoire of profile keywords. The algorithm can only modify the weights of keywords in the profile's vector. It has also been argued, that learning algorithms cannot easily cope with radical interest shifts [20]. A new profile is typically generated whenever a new topic of interest emerges and a profile that corresponds to a no longer interesting topic is destroyed [21]. This is of course only a partial solution to the problem that unnecessarily complicates the task with additional system parameters and in any case, no profile will be able to represent a topic that is not already covered by the keywords in the predefined vector space.

3. A NETWORK-BASED PROFILE

To alleviate the above issues, we propose an alternative to the VSM. We need a model for IF that satisfies the following basic requirements:

- It does not require a fixed (and common) vector space that a priori defines the available features for representing user profiles and documents.
- The new model should not ignore the additional information encoded in correlations between features that appear in the same context. In the case of textual information, this means capturing dependencies between terms in text.
- The user profile should be dynamic, capable of continuously adapting its structure to the changing user interests and the evolving information space. A user profile that can not maintain a satisfactory level of performance will eventually dissatisfy the user and will be abandoned.

In the proposed model, the user profile is a weighted network with nodes representing features extracted from interesting information items and links representing their correlations. Since we will concentrate on textual information, nodes will represent terms (i.e., single words) extracted from the content of documents and links will represent correlations between terms in text. The discussion however can be easily extended to any type of descriptive feature that can be automatically extracted from the content of information items, or have already been assigned to them (e.g., tags). Every node is assigned a weight that measures the importance of the corresponding term given the user's interests. Every link between two nodes is assigned a weight[1] measuring the degree of correlation between the respective terms. The weights are of course important, but the way they are calculated is not constrained by the model itself. Any keyword weighting method can be used to calculate the weights of nodes in the profile's network. To calculate the weights of links, co-occurence statistics between terms appearing in the same context are required. Various appropriate methods appear in the literature and have been used to construct similar network structures for capturing term correlations in Text Retrieval and even Information Filtering. For example, "collocation maps" [12] and "dependence trees" [19] have been proposed for query expansion and "concept hierarchies" [16] for navigating document collection and search results. In IF, correlations between terms have been generally neglected. One notable exception is the adoption of an associative graph for capturing syntactic correlations between terms that appear next to each other and of a spreading activation process for document evaluation in [18].

The essential contribution of the proposed model is not the network itself, but a new way of using the weighted network to evaluate the relevance of documents. We treat content-based IF as the general problem of assigning a relevance score to each document based on its content, rather than making a binary classification between relevant and non-relevant documents. Document evaluation is based on a *non-iterative*, *directional*, spreading activation process that takes into account not only the weight of profile nodes (terms), but also the weight of links between them. The process can be deployed to assign a relevance score to any portion of a document's text, ranging from a single sentence to the complete document. The document is not treated as a "bag of words" and it does not have to be represented as a keyword vector. Nevertheless, the terms in the text can be weighted with methods such as Term Frequency Inverse Document Frequency (TFIDF). To assign a relevance score to a portion of text T, each term in the profile's network that also appears in T is assigned an initial activation equal to the term's weight in T. The activation phase is followed by a dissemination phase, which starts with the activated node with the smaller weight in the profile's network and proceeds sequentially with the remaining activated nodes in increasing weight order, until the activated node with the largest weight is reached. Every activated node is triggered once to disseminate part of each current activation to the activated nodes with larger weights that it is linked to. The amount of activation that is disseminated between two nodes is proportional to the weight of the link between them. The relevance

[1]Here we focus on symmetric links but the model could also account for non-symmetric links.

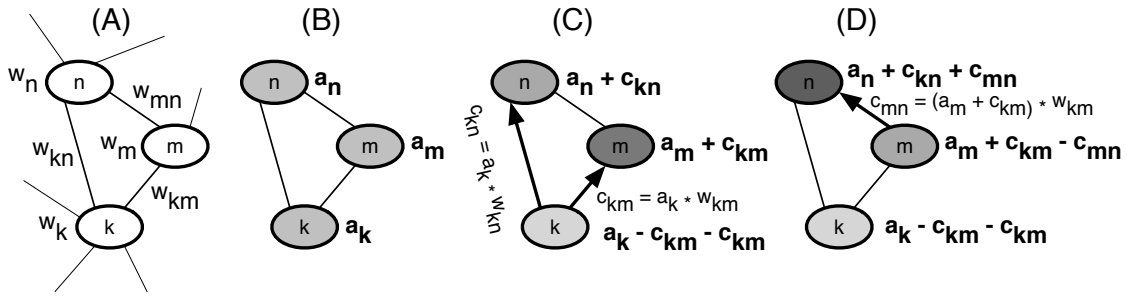

Figure 1: Directional Spreading Activation: (A) idle network, (B) activation phase, (C) node k disseminates, (D) node m disseminates.

score of T is calculated as the weighted sum of the final activation of nodes.

Figure 3 illustrates the above process. The portion of text T activates the initially idle nodes k, m, n, out of the complete profile network, which is not depicted in the figure. The weights of the three nodes are w_k, w_m, w_n respectively, with $0 < w_k < w_m < w_n$, and the weights of the links between the three nodes have weights w_{kn}, w_{km} and w_{mn} with positive values. The initial activation of the three nodes (a_k, a_m, a_n) is equal to the weight of the corresponding terms in T. The dissemination process starts with the activated node with the least weight[2]. Node k disseminates an amount of activation equal to $c_{kn} = a_k \cdot w_{kn}$ to node n and an amount equal to $c_{km} = a_k \cdot w_{km}$ to node m. To avoid the situation where a node disseminates more than its current activation, i.e., when the sum of the weights of links to activated nodes is more than one, we normalise the weight of these links so that they add up to one. Once node k has disseminated its activation, the new activation of the three nodes k, m and n, becomes $a_k - c_{kn} - c_{km}$, $a_m + c_{km}$ and $a_n + c_{kn}$ respectively. It is now the turn of the next node in the order of increasing weight to disseminate part of its current activation to activated nodes with larger weight that it is linked to. So node m disseminates the amount $c_{mn} = (a_m + c_{km}) \cdot w_{mn}$ to node n and their respective activation becomes $a_m + c_{km} - c_{mn}$ and $a_n + c_{kn} + c_{nm}$. At this point the dissemination process terminates because node n is not linked to any other activated nodes with larger weights. The relevance score R_T of T is given by the following formula:

$$\begin{aligned} R_T &= w_k \cdot (a_k - c_{kn} - c_{km}) + w_m \cdot (a_m + c_{km} - c_{mn}) \\ &\quad + w_n \cdot (a_n + c_{kn} + c_{nm}) \\ &= (w_k \cdot a_k + w_m \cdot a_m + w_n \cdot a_n) \\ &\quad + (w_m - w_k) \cdot a_k \cdot w_{km} + (w_n - w_k) \cdot a_k \cdot w_{kn} \\ &\quad + (w_n - w_m) \cdot (a_m + a_k \cdot w_{km}) \cdot w_{mn} \end{aligned} \tag{1}$$

Note that the first term in the above sum, is actually the inner product between a weighted keyword vector of the three profile terms and a weighted keyword vector of the three terms in T. In other words, the above formula specialises to the inner product if there are no links between the activated nodes ($w_{kn} = w_{km} = w_{mn} = 0$). However, when the activated nodes are linked then the relevance score of T increases by an additional amount equal to the sum of the last three terms in equation 1. It is clear that this additional amount is positive because all weights are positive and $w_k < w_m < w_n$. So every time a portion of text activates correlated nodes in the profile's network and not isolated ones, it receives an additional relevance reward. In this way the profile becomes more specific, because it concentrates on those term combinations that are relevant to the user's interests and it does not equally represent every possible combination of terms in the profile. Our experimental results clearly show that this property results in significant improvements in filtering accuracy, especially when the number of terms in the profile increases.

Overall, the proposed model, as described so far, satisfies the first two of the three requirements described earlier. The user profile is not represented as a weighted keyword vector on a common, central, predefined and fixed space. For each individual user, the profile is a separate network structure containing only those terms that are representative of the user's interests. Furthermore, the number of profile terms is neither predefined nor fixed. It is dynamically controlled during the profile's adaptation to interest changes. The profile's network captures correlations between terms in text and a directional spreading activation process takes them into account during document evaluation. Note also, that unlike existing spreading activation processes that are typically iterative and involve the complete network [18], thus increasing the computational cost, the proposed approach involves only the subset of activated profile nodes and its directionality ensures that each activated node is visited only once and thereafter, each link between activated nodes is also traversed only once. So in the worst case of a fully connected and fully activated network, its complexity is $O(N_p^2)$, where N_p is the number of profile terms. The inclusion of links increases the complexity of the user profile, in comparison to the linear complexity $O(N_L)$ of a vector-based profile in a N_L-dimensional space, using the inner product for document evaluation. Note however, that if a common vector space is used to represent the profiles of many users with a variety of interests, then a large number of keywords would be required to cover all possible interests. In contrast, the proposed model is inherently distributed and comprises only the terms required to represent the interests of a single user. So $N_p << N_L$ and the difference in complexity between $O(N_p^2)$ and $O(N_L)$ is alleviated. If needed, the computational and memory requirements of each profile can be controlled with upper limits on the number of profile terms and links. In any case, if the user profile resides on the user's machine, scalability issues do not arise.

[2]If two terms have the same weight then they are ordered alphabetically

Table 1: Topics involved in the experiments and their corresponding size

topic	earn (1)	acq (2)	money-fx (3)	crude (4)	grain (5)	trade (6)	interest (7)	wheat (8)
size	3987	2448	801	634	628	552	513	306
topic	ship (9)	corn (10)	dlr (11)	oilseed (12)	money-supply (13)	sugar (14)	gnp (15)	coffee (16)
size	305	254	217	192	190	184	163	145
topic	veg-oil (17)	gold (18)	nat-gas (19)	soybean (20)	bop (21)	livestock (22)	cpi (23)	
size	137	135	130	120	116	114	112	

The theoretical background of the proposed model is discussed in detail in [8] and is biologically-inspired. The user profile is modelled after the network of interacting antibodies in the immune system of vertebrates and through the chains of suppression and reinforcement that the spreading activation process generates, it defines the host organism's "self", i.e., the user's interests. In the same paper, we describe an algorithm that allows the profile to continuously adapt to a variety of interest changes, through a biologically-inspired process of self-organisation. The algorithm adjust the profile's structure in response to user feedback, through variations in the weight of profile terms, recruitment of new terms that cover emerging topics of interest and removal of terms that correspond to no longer interesting topics. So the profile is not constrained by the terms in a pre-defined vector space. Experiments show that through this process the profile can adapt to both short-term variations and more long-term, radical changes in user interests, while autonomously controlling both its size and connectivity [8]. Furthermore, comparative experiments show that this algorithm outperforms the popular Rocchio's learning algorithm on a continuous learning problem [11]. So, although outside the scope of the current work, the third of the aforementioned requirements can also fulfilled.

4. COMPARATIVE EXPERIMENTS

In this section, we evaluate experimentally a specific realisation of the model and compare it with a vector-based profile. In [10], we argued that existing evaluation methodologies do not accurately reflect the particularities of content-based IF, mainly because they usually treat it as a Text Classification problem, with each user profile representing a single topic of interest and trained with a large number of relevant documents. Furthermore, since the removal of the filtering track from the Text Retrieval Conference (TREC) in 2001, there is no established evaluation standard for content-based IF. Here we adopt a stripped down version of the methodology in [11], which simulates users with multiple interests, but ignores interest changes.

The methodology uses the Reuters-21578 document collection, but could be applied for any pre-classified collection of documents. The evaluation concentrates on the 23 topics in Reuters-21578 with more than 100 relevant documents (table 1). Each topic is assigned a serial number (in parenthesis) to identify it when presenting the experimental results. For each topic we use the first fifty relevant documents in the collection for training and the complete collection as a test set. This is a significant departure from current practices for the evaluation of Text Classification systems, such as the ModApte split, that uses three quarters of the documents for training and the remaining quarter for testing. By using the same small number of training documents per topic and given that the number of relevant documents range from 112 to 3987 (table 1), a variable percentage of the relevant documents is used to build the profile. So the user profile has to be specific for topics with a small number of relevant documents and exhaustive for topics with a large number of relevant documents.

The most significant contribution of the proposed methodology, is the simulation of users with multiple interests. In particular, we simulated users with parallel interest in one, two, three, four and five topics. For instance, to simulate a user interested in two topics we train a single profile for each combination of two consequent topics (e.g., earn and acq (1:2), acq and money-fx (2:3), money-fx and crude (3:4) and so on). We combine consequent topics with similar sizes to avoid biases towards topics with a larger number of relevant documents. For each topic combination we use the first 50 relevant documents per topic to train a single profile, which is then used to evaluate the complete collection. The 21578 documents are then ranked according to their relevance score and the Average Uninterpolated Precision (AUP) measure is calculated for each individual topic and also for their combination, i.e., an aggregate topic that includes all documents relevant to the constituent topics. A topic's AUP is defined as the sum of the precision – i.e., the percentage of documents relevant to that topic – at each point in the ordered list where a relevant document appears, divided by the topic's size. We use an evaluation list comprising all documents in the collection and not just the best 1000 scoring documents (as in TREC's routing subtask). This way, we obtain more accurate and unbiased measurements, since a list of the best 1000 scoring documents can be easily populated when a topic of interest has a far larger number of relevant documents. Furthermore, such a list can be biased towards one of the topics in a combination, because it can be dominated by the topic's relevant documents at the expense of documents relevant to the rest of the topics. For similar reasons, we preferred Reuters-21578 and not the more recent RCV1, because the latter causes evaluation problems due to the very large number of relevant documents per topic in the collection [14].

Overall, the methodology defines a challenging IF task that more accurately reflects the problem's complexity and proposes an alternative to existing practices. As the number of topics of interest increases from one to five the necessary number of profile terms also increases. Our aim is to experimentally support our argument regarding the effectiveness of the proposed model in comparison to the VSM when this happens.

4.1 Experimental Setting

The documents in the collection are first pre-processed using stop word removal and stemming with Porter's algorithm. We then used Information Gain (IG) [4] to weight the remaining terms in the training documents. Terms with positive weight were extracted to build a user profile for each topic or topic combination. Two different types of profile were constructed. The baseline profile is a vector-based

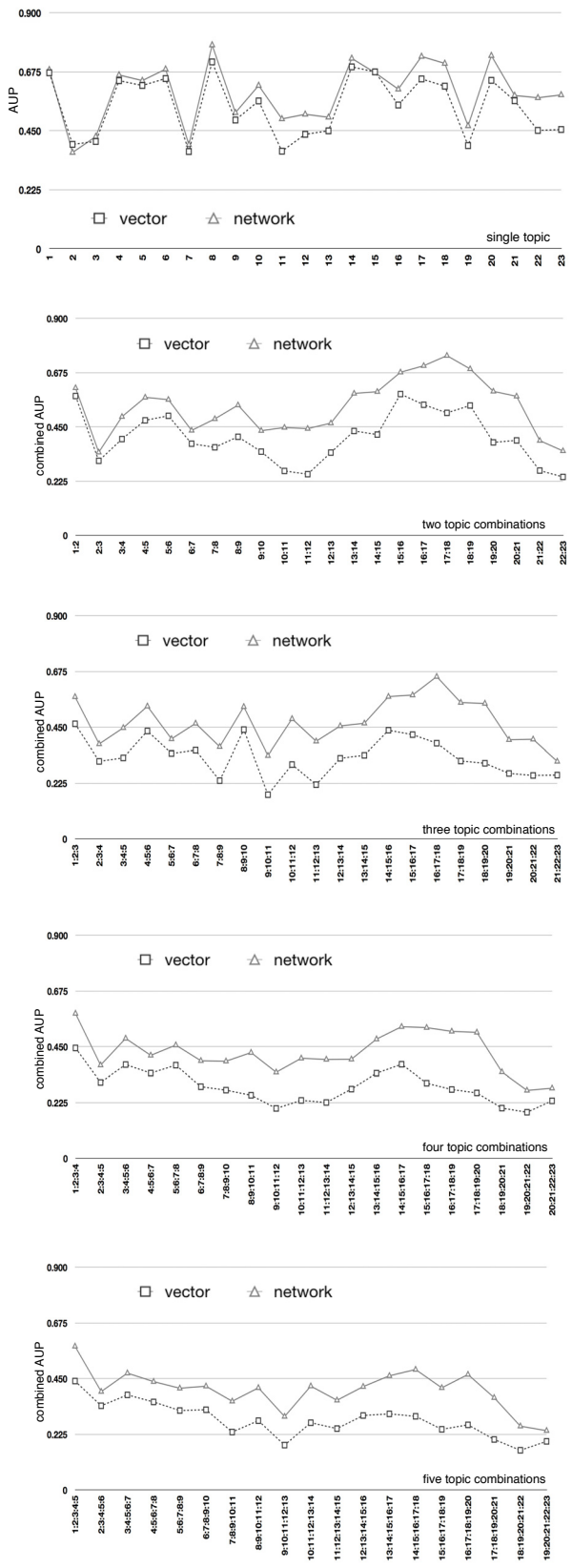

Figure 2: AUP scores for the five experiments: one-topic to five-topic.

Table 2: Summary of Results

topics:	one	two	three	four	five
per. increase (%)	10.47	33.9	45.68	50.24	46.39
st. deviation	9.83	17.92	22.68	23.55	19.95
paired t-test	$\ll .001$	$\ll .001$	$\ll .001$	$\ll .001$	$\ll .001$
av. no. terms	1123.2	1820.5	2333.5	2750.1	3094.0

profile comprising the extracted weighted terms. The second type, is the proposed network-based profile comprising exactly the same weighted terms, but an additional process is deployed to generate the links between profile terms and calculate their weights. In particular, a sliding window approach is used to define the context of terms in text and identify their co-occurences. The window defines a span of 10 contiguous terms[3]. Every time two profile terms appear within the window in the training documents, a link between them is established. The weight w_{kn} of the link between two terms k and n is calculated using the following equation, which is similar to the one adopted in [12], extended with an additional factor based on the average distance, i.e., the number of terms that intervene between k and n in the sliding window.

$$w_{kn} = \frac{fr_{kn}^2}{fr_k \cdot fr_n} \cdot \frac{1}{d_{kn}} \tag{2}$$

where:

fr_{kn}	is the number of times k and n cooccur within the sliding window
fr_k and fr_n	are respectively the number of occurrences of k and n in the training documents
d_{kn}	is the average distance between k and n, within the sliding window.

To evaluate each document in the collection, the same sliding window is deployed, so that only terms found in the same context get activated. Terms in the document are not weighted. Every position of the window defines a portion of the document's text. In the case of the vector-based profile, we assign a relevance score to the window by calculating the inner product between the profile and a keyword vector comprising the terms in the window. The network-based profile assigns a relevance score to the window using the proposed spreading activation process. In both cases, the result of the document evaluation process is a relevance score for each position of the window in the document's text, which indicates the distribution of relevance through out the document. For practical purposes though, we calculate a single relevance score for each document as the sum of the individual window scores, normalised to the logarithm of the number of terms in the document. It is important to note, that since both types of profile comprise exactly the same weighted terms, any difference in their performance is due only to the additional relevance that links contribute to documents, according to equation 1.

4.2 Experimental Results

Figure 2 includes five graphs one for each of the single-topic, two-topic, three-topic, four-topic and five-topic experiments. The x-axis shows the serial numbers of topics or topic combinations (e.g., 1:2 corresponds to earn:acq) and

[3]The window's size was chosen based on systematic experiments.

Figure 3: Average AUP for different numbers of profile terms.

Table 3: 3-Topic Experiment: overall results for different threshold values

threshold ($\times 10^{-6}$)	av. no. of terms	vector	network	increase (%)	st. dev.	t-test (p-value)
0	2333.5	0.329	0.469	45.68	22.68	4.17E-10
100	2308.7	0.330	0.469	45.64	22.58	3.91E-10
300	1976.6	0.330	0.465	43.69	21.52	4.17E-10
500	1489.7	0.330	0.456	40.96	20.30	6.43E-10
1000	763.0	0.329	0.437	34.91	18.19	2.61E-09
1500	510.7	0.327	0.422	30.73	17.04	2.61E-09
2000	369.0	0.325	0.409	27.26	15.90	2.89E-08
3000	228.0	0.319	0.388	22.41	14.19	1.74E-07
4000	154.2	0.312	0.368	18.46	13.80	2.06E-06
5000	111.3	0.308	0.354	15.40	12.41	7.87E-06
6000	82.2	0.302	0.341	13.26	11.77	2.12E-05
7000	62.9	0.294	0.330	12.00	10.67	5.23E-05
8000	48.4	0.285	0.314	10.01	9.83	1.78E-04
9000	37.9	0.274	0.299	9.17	9.35	2.65E-04
10000	31.1	0.269	0.295	9.33	8.95	1.96E-04
15000	13.2	0.212	0.228	6.16	9.35	7.35E-03

the y-axis the corresponding AUP value, or combined AUP value in the case of multi-topic experiments. Table 2 summarises for each of the five experiments the average percent increase, the standard deviation, the p value of the paired, two-tailed t-test, and the average number of profile terms.

The results clearly show that as the number of topics of interest increases, causing an increase in the number of profile terms, the network-based profile achieves significant performance improvements of up to 50.2% on average, over the vector-based profile. These differences are consistent through out the 23 topics, as highlighted by the standard deviation, and are statistically significant, since all the p values of the t-test are less than 0.05. It is also evident, that the difference is more pronounced for topics with a small number of relevant documents in the collection, where specificity is more important. As expected, the achieved AUP values get smaller as the number of topics of interest increases, because the IF task becomes more difficult.

The observed differences in performance are not only substantial, but also of computational interest, because they are due only to the existence of links in the network-based profile, since both types of profile comprise exactly the same weighted terms. Furthermore, although not reported here due to space limitation, we have also performed the same experiments with a different term weighting method, called Relative Document Frequency (RelDF) [13]. The results achieved for RelDF were overall worse than those presented here for IG, but the network-based profile achieved improvements of up to 75% on average, over the vector-based profile[4]. It is also evident, that the observed differences in AUP scores relate to the increase in the number of profile terms required to represent multiple topics of interest.

To further investigate this effect we repeated the experiments for different numbers of profile terms. In particular, we present here results for the three-topic experiments, where we progressively increase a threshold and we only extract from the training documents terms with weight larger than this threshold. Table 3, summarises the average AUP values of the vector-based and network-based profiles for different numbers of profile terms, the percent increase, the standard deviation and the p value of the paired, two-tailed t-test. Figure 3 presents a plot of the average AUP (columns 3 and 4 in table 3) on the y-axis, as the average number of profile terms (column 2 in table 3) increases on the x-axis. When the number of profile terms is small the AUP scores of the two types of profile are also small, because there is a limited scope for capturing three topics using a small number of terms. As the number of profile terms increases the AUP scores of both profile types increases, but in the case of the vector-based profile the curve quickly flattens. Approximately, after extracting the first 400 terms with the highest weights the remaining 2000 terms do not essentially contribute to the profile's accuracy. On the contrary, the AUP score of the network-based profile increases more rapidly with the increase in the number of terms and keeps on increasing until all terms with positive weight have been incorporated. The p-values show that as the number of profile terms increases the confidence in the comparison also increases. Although not reported here due to space limitations, these differences are more pronounced for more topics of interest.

These findings are revealing, because they demonstrate that the network-based profile can effectively incorporate a large number of terms (or features in general). Unlike the vector-based profile that does not distinguish between relevant and non-relevant term combinations, the term network can exploit the additional information about the user's interests, that a large number of terms and their correlations encode. We are confident that these findings generalise beyond textual information, to any type of information that can be described or associated with correlated features. A user profile that is "resistant" to dimensionality problems can not only represent multiple topics of interest more accurately than a vector-based profile, but in principle, it can incorporate a greater diversity of features, including context dependent ones. A large number of features is no longer a problem, but an advantage that can be exploited to incorporate, in the profile, additional features (such as tags) that have been assigned to, or can be automatically extracted from information items. In this way, the scope of IF can be easily extended beyond textual information and the specificity of a user profile can be augmented.

5. SUMMARY AND FUTURE WORK

The problem of information overload still impedes the dissemination of information on today's Web, where everyone can be both a receiver and a transmitter of information.

[4]All experimental results can be found at http://www.scribd.com/IF_SIGIR10

IF has an important role to play in achieving personalised information delivery to ensure that the right information reaches the right people. However, unlike the success stories of Collaborative Filtering that produced popular applications for recommending books, movies and music tracks, content-based IF has not yet lead to the development of widely adopted Web applications for personalised information delivery. In this paper, we argued that one possible explanation is the reliance on the VSM, which leads to inherent dimensionality problems. Instead, we proposed an alternative network-based model for profile representation that, in the case of textual information, captures correlations between terms appearing in the same context. The network profile can evaluate any portion of text with a directional spreading activation process.

We performed a series of experiments comparing the network profile to a vector-based profile containing the same weighted terms. Unlike existing evaluation practices, our experimental methodology simulates users with multiple concurrent interests. Representing multiple topics of interest requires a large number of profile terms and reveals the dimensionality problems of the VSM. The existence of links allows the network profile to capture additional information about the user's interests, become more specific and achieve significant performance improvements. We specifically investigated the effect of the number of terms on the profile's performance and found out that after just a few hundred terms the vector-based profile reaches its maximum representational capacity, while the network profile can effectively incorporate thousands of terms.

This is a significant property that extends beyond textual information and terms as features. We envision user profiles that do not only incorporate the necessary number of terms for representing a user's multiple interests, but are also hybridised with additional features, such as tags or even user ratings. In this way, the user profile will become more specific to the user's interests and will enable a variety of personalisation services. Given that the proposed model is also distributed and dynamic, it proposes a new perspective towards IF and may establish a new research paradigm. This paper is part of ongoing effort towards this direction, that involves further experiments, theoretical analysis and real world prototype implementations.

6. REFERENCES

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behaviour of distance metrics in high dimensional space. *Database Theory — ICDT 2001*, Volume 1973/2001:420–434, 2001.

[2] T. Ault and Y. Yang. knn, rocchio and metrics for information filtering at trec-10. In *TREC*, 2001.

[3] R. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[4] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth International Group, Belmont, CA, 1984.

[5] P. W. Foltz. Using latent semantic indexing for information filtering. In *Proceedings of the ACM SIGOIS and IEEE CS TC-OA conference on Office information systems*, pages 40–47, 1990.

[6] W. P. Jones and G. W. Furnas. Pictures of relevance: A geometric analysis of similarity measures. *Journal of the American Society of Information Science*, 38(6):420–442, May 1986.

[7] C. McEwan and E. Hart. Representation in the (artificial) immune system. *Journal of Mathematical Modelling and Algorithms*, 8(2):125–149, 2009.

[8] N. Nanas and A. De Roeck. Autopoiesis, the immune system and adaptive information filtering. *Natural Computing*, 8(2):387–427, 2009.

[9] N. Nanas, S. Kodovas, and M. Vavalis. Revisiting evolutionary information filtering. To appear in *Congress on Evolutionary Computation*, 2010.

[10] N. Nanas, M. Vavalis, and A. De Roeck. What happened to content based information filtering? In *Advances in Information Retrieval Theory, Second International Conference on the Theory of Information Retrieval (ICTIR 2009)*, pages 249–256, 2009.

[11] N. Nanas, M. Vavalis, and L. Kellis. Immune learning in a dynamic information environment. In *Artificial Immune Systems, 8th International Conference (ICARIS 2009)*, pages 192–205, 2009.

[12] Y. C. Park and K.-S. Choi. Automatic thesaurus construction using bayesian networks. *Information Processing and Management.*, 32(5):543–553, 1996.

[13] M. F. Porter. Implementing a probabilistic information retrieval system. *Information Technology: Research and Development*, 1:131–156, 1982.

[14] S. Robertson and I. Soboroff. The TREC 2001 filtering track report. In *The Tenth Text Retrieval Conference (TREC-10)*, pages 26–37, 2001.

[15] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Inc., 1983.

[16] M. Sanderson and B. W. Croft. Deriving concept hierarchies from text. In *22nd ACM SIGIR Conference*, pages 206–213, 1999.

[17] R. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In *21st ACM SIGIR Conference*, pages 215–223, 1998.

[18] H. Sorensen, A. O' Riordan, and C. O' Riordan. Profiling with the informer text filtering agent. *Journal of Universal Computer Science*, 3(8):988–1006, 1997.

[19] C. J. van Rijsbergen. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–199, 1977.

[20] G. I. Webb, M. J. Pazzani, and D. Billsus. Machine learning for user modeling. *User Modeling and User-Adapted Interaction*, 11:19–29, 2001.

[21] D. H. Widyantoro, T. R. Ioerger, and J. Yen. An adaptive algorithm for learning changes in user interests. In *ACM/CIKM'99 Conference*, pages 405–412, 1999.

[22] Y. Yang, A. Lad, N. Lao, A. Harpale, B. Kisiel, and M. Rogati. Utility-based information distillation over temporally sequenced documents. In *30th ACM SIGIR Conference*, pages 31–38, 2007.

[23] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *14th International Conference on Machine Learning (ICML '97)*, pages 412–420, 1997.

[24] Y. Zhang. Using bayesian priors to combine classifiers for adaptive filtering. In *27th ACM SIGIR Conference*, pages 345–352, 2004.