Privacy-Preserving Profiling

Thomas Barnard, Adam Prügel-Bennett
Information: Signals, Images, Systems (ISIS),
School of Electronics and Computer Science,
University of Southampton, United Kingdom

Abstract— *With the rise of social networking, and other sites which collect vast amounts of user data, the issue of user privacy has never been more important. When creating user profiles care must be taken to avoid collecting sensitive information, while ensuring that these profiles are fit for purpose. In this paper we present a specific instance of the privacy-preserving profiling problem in an expert-finding application. We present a dataset of profiles, as well as several datasets for contaminating these profiles, and provide experiments to test data quality and privacy-preserving performance. We present a simple solution based on training an LSA model on a clean profile corpus, which maintains performance and provides a moderate level of privacy.*

Keywords: User Profiling, Information Retrieval, Privacy

1. Introduction

People spend an increasing amount of their time using social networking sites. In building and maintaining social networking profiles, users provide large amounts of information to these sites. Of course these users expect something in return; providing this information may help to find new friends or business contacts, and strengthen existing relationships, while the social networking provider gains access to profiles which it can use to provide personalized advertisements.

There have been a number of cases of privacy being compromised or potentially compromised by user profiling. Facebook have been criticized for their use of profiling in providing personalized adverts, which may allow advertisers and others to discover the sexual orientation of users[1]. Privacy concerns also led to the second Netflix recommendation prize being cancelled, and the dataset for the first prize being made unavailable for download[2].

While the user shares information about their interests and contacts, they may unwittingly disclose private information about themselves. Relying on a user to ensure their own privacy is an unacceptable solution, both because it places an additional burden on the user, and because the user may not be the best judge of what information about themselves should be made available. They may also be broadcasting their details more widely than they realise; privacy settings may be set incorrectly, and third-party applications may collect data from profiles without users' knowledge or consent.

In this paper we will introduce the problem of privacy-preserving profiling. We will look at the specific problem of generating profiles within the *Instant Knowledge* project. We will describe a series of experiments to determine the preservation of privacy, and use these experiments to evaluate our early attempts to solve this problem.

2. Instant Knowledge

The *Instant Knowledge* (IK) project aims to provide as solution to the problem of finding experts within an organization. It can be difficult to keep track of expertise within an organization, which can limit collaboration, or make it difficult to find the appropriate people to work on a new project. In academia researchers often find out too late that somebody was working on a similar problem in the same department, with each unaware of the other's work.

The IK system is a keyword-based information system utilizing a client-server architecture. Users' personal devices collect context information, and generate queries based on user activity. Keywords relating to an area of expertise are sent to the server which returns a ranked list of experts. In this paper we will focus on the generation of profiles, and ignore more complex aspects of the system such as context awareness, distributed algorithms, and query augmentation.

The IK system requires accurate, up-to-date profiles of expert interests in order to provide the best responses to user queries. The simplest method of generating these profiles would be for the experts to enter free-text describing their professional interests. This may, however, lead to profiles which are poorly maintained as the user loses interest in the task.

The next step would be for users to manually provide documents which they feel represent their interests, for example technical reports or academic publications. This approach is not without its problems, as it still requires user effort. Even if documents are added automatically, for example if they are added to a publication repository, if these documents are added infrequently, they might not fully represent a user's interests. Certain approaches to a problem may not lead to a publication but may nonetheless help enrich a user's profile.

Instead we favour a fully automatic approach, building a profile from all the documents authored or collected by a user, as well as other sources of information, such as email, web browsing activity, and social networking. By including these

additional sources of information we hope to build profiles which are more accurate and up-to-date than those produced manually.

This approach does, however, present some challenges; some of the information collected will be irrelevant or private. In the case of irrelevant data, recommendation performance may be reduced, in the case of private information disclosure may have serious negative consequences.

3. Privacy

Profiles within the IK system are assumed to be private in the sense that their exact contents is only known to the user they belong to and the system itself. In this paper we will assume that there are no third parties who can peek at the profile, or observe it in transit from the expert to the IK server. The user profile is however assumed to be accessible, either publicly or within an organization, through the profile recommendation system.

The main attack vector we consider is profile reconstruction through repeated queries. By making a series of carefully constructed queries it may be possible to infer the presence and weights of certain terms and concepts within a profile, by observing how highly a given user is ranked for these queries. The construction of such an attack will not be addressed in this paper.

While a notion of privacy in data mining and user profiling can have a number of different interpretations, from anonymity to an uncertainty in the particular values of an attribute, we consider profiles to be made up of public and private information, and it is our job to remove the private information while leaving the public information intact. This is in contrast to some applications where the whole profile is assumed to be private; the need to recommend specific, named users is incompatible with absolute privacy.

We aim to conceal two main types of private information within a profile: passwords, bank account details, usernames, and other private tokens; interests which would be embarrassing, controversial, or would cause some harm to the user should they be disclosed. We are also interested in removing irrelevant information from a profile, for example non-professional interests such as musical tastes, or hobbies.

4. Privacy-Preserving Profiling

Our goals in automatic privacy-preserving profiling are the production of an useful user profile, and the preservation of user privacy. These goals are to some extent at opposition with each other: as we remove private information we will remove useful profile which will reduce performance; as more information is retained in a profile the greater the risk of disclosing sensitive information will be.

Our task is made harder as our privacy-preserving techniques must operate without user input. It would be much simpler to train a classifier to identify public and private documents by using user labelled documents, building a model for each user. We could consider building a global model using a profile corpus and examples of private information.

The problem here is that what each user considers private may vary considerably. It could be argued that there are subjects that most users would consider private for example sexual preferences and habits, political affiliation, or health concerns. For some users, however, these controversial topics may be their main area of expertise, so we cannot filter them outright.

Determining the nature of information without help from the owner of that information requires us to rely on patterns in the data itself, and the overall properties of public and private data in general. It is difficult and may be impossible to build a privacy-preserving profile by analysing an expert's documents *in vacuo*.

5. Methodology

As our focus is on the automatic production of profiles and their privacy preserving attributes we have implemented a very simple information retrieval system.

The documents belonging to a user are converted into a bag-of-words representation, removing structure, turning them into an unordered collection of words. Commonly words with little discriminative power, called stop words, are removed. We use the list provided by Fox in [3]. Finally words are reduced to their root form using a stemming algorithm, for example 'computer' and 'computation' may be reduced to the stem 'comput'. Finally these processed words are counted to produce a term frequency representation of the original document. While this processing removes some information from the documents and may result in reduced performance, it should also remove private information.

We could produce profiles by adding together term frequency representations of their constituent documents, however this could lead to larger documents dominating the profile. Instead we normalize these document representations by their length before adding them together,

$$\mathrm{TW}_{p,i} = \sum_{j \in D_p} \frac{\mathrm{TF}_{j,i}}{N_j}, \tag{1}$$

where $\mathrm{TW}_{p,i}$ is the weight of term i in profile p, D_p is the set of documents that profile p contains, $\mathrm{TF}_{j,i}$ is the frequency of term i in document j, and N_j is the size of document j.

We then use a vector-space model (VSM)[4], treating each profile as a multidimensional vector, where each dimension corresponds to the weight of a particular term in the profile. We apply a weighting scheme to the raw frequency based weights called TF-IDF, here the term frequency weight is normalized by the profile length, and multiplied by the inverse document frequency (IDF), giving a higher weighting to terms which occur in fewer documents. The TF-IDF

weighting equations are given below,

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}},$$
$$IDF_i = log\frac{|D|}{|\{d : t_i \in d\}|},$$
$$TFIDF_{i,j} = TF_{i,j} \times IDF_i,$$

where $tf_{i,j}$ is the term frequency of term i in document j, $n_{i,j}$ is the number of times term i occurs in document j, idf_i is the inverse document frequency of term i, D is the collection of documents, and $tfidf_{i,j}$ is the TF-IDF weighting of term i in document j.

The process described so far will produce vectors which may have many thousands of dimensions. Differences in the terms used means that documents which concern similar topics may have few terms in common. In addition high-dimensional vectors require more resources to manipulate and compare. To solve these problems we apply a dimensionality reduction technique to our profile vectors.

Latent Semantic Analysis (LSA), or Latent Semantic Indexing (LSI) is a technique for taking document vectors and projecting them into a lower dimensional space[5]. As well as reducing the dimensions, LSA has the advantage of projecting the term vectors into a concept space, where concepts are represented rather than specific terms. This means that terms with similar meanings are close in this space, where in term space there would be no match.

LSA is implemented using a singular value decomposition (SVD) of the profile matrix. The details of this process are beyond the scope of this paper, but essentially the matrix is factorized into a form capturing the directions of maximal variance in the data,

$$A = USV^T, \tag{2}$$

where U and V are matrices corresponding to rows (terms) and columns (profiles) in the matrix respectively, and S contains the singular values. By retaining only the top singular values it is possible to reduce the dimensionality of the matrix. This also has the effect of removing noise in the matrix at the expense of fine detail.

To compare profiles and queries we must first project them into concept space,

$$\hat{D} = S^{-1}U^T D, \tag{3}$$

where D is the document, and $\hat{D}$ is its concept-space representation. We compare vectors by using the cosine similarity which gives a value between 0 and 1 indicating the degree to which two vectors point in the same direction,

$$similarity(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}||||\mathbf{B}||}. \tag{4}$$

To recommend a profile given a query, we simply order profiles in descending order of their similarity to the query vector.

While these simple techniques may lead to useful information being removed from profiles, such as the context of words in a bag-of-words model, or the same word being used with a different meaning being ignored by an LSA model, their simplicity makes analysis of the privacy-preserving aspects easier. Removing structural information from documents should also lead to an increase in privacy.

5.1 Privacy Preservation

The projection of data onto a lower dimensional concept space provides some blurring of information, and innocuous documents and terms will share some similarity with private documents and terms. This provides some measure of plausible deniability, at the expense of loss of fine detail. Some evidence of private terms may remain.

We propose a method of privacy-preserving profiling using a technique we have already described in this paper, *Latent Semantic Analysis*. LSA works by finding a concept space representing a collection of documents, whose dimensions represent the directions of greatest variability in the collection of documents.

Making the assumption that public information differs significantly from private information, and using the fact that the LSA projection depends on the corpus of documents that was used to create it, we propose the simple technique of using a corpus of public information to build a projection, and then projecting all information into this concept space.

Our hypothesis is that as the public concept space has been learned using public documents it will be less well suited to representing private information. In this way private information will be "projected out" of the profiles.

6. Related Work

While there has been quite a lot of research into privacy in data mining in profiling generally, there has been surprisingly little research into the problems described earlier in the paper. That is profiles are usually treated as objects which are either wholly private or public.

Reichling et al.[6] presented a similar approach to user profiling for the purpose of finding experts, using an LSA model to represent profiles. In their approach privacy is dealt with manually: the user is responsible for selecting directories which the system is allowed to search for documents.

Privacy preserving data mining (PPDM) is a growing area of research which aims to ensure that data mining activities can be conducted while safeguarding user privacy[7]. While there are some overlaps with what we are doing, most research in PPDM seems to deal with anonymity[8], hiding precise values of data[9], and cryptographic methods.

While the problem may at first appear to be superficially similar to the problem of spam filtering, except the aim is to prevent information leaking out rather than being received, there are some important differences. Firstly with spam filtering it is possible to maintain a global model of

spam which can be used to filter incoming messages for every user, this may then be tweaked by user feedback (e.g. identifying misclassified messages), but large changes to the global model seem unlikely. Secondly, instead of filtering documents out completely, we may have documents which contain a mixture of private and public information and it would be ideal to have this public information added to the profile.

7. Experiments

Bertino et al.[10] describe five criteria with which to evaluate PPDM algorithms:

- Efficiency
- Scalability
- Data Quality
- Hiding Failure
- Privacy Level

Of these criteria the most applicable to our problem are data quality and hiding failure.

Data quality describes the effect that the privacy preserving process has on the original data. They suggest that this can be tested by the change in data mining performance on the when using the processed data versus the original dataset. Hiding failure relates to the amount of private data that can be recovered from the sanitized data.

In the following sections we will describe the experiments we performed to test our techniques given these criteria.

7.1 Datasets

In order to test our hypothesis and carry out experiments in user profiling we require both a source of user profiles, and of private information with which to "poison" them. It would be difficult and time consuming to obtain samples of real user profile data, as well as real private information, so instead we have created profiles from academic publications data and obtained surrogate private information from a different source.

The RKBExplorer website[1] which is part of the ReSIST project at the University of Southampton provides a semantic web database containing information from a number of institutions where authors of academic papers have self-archived their publications in ePrints repositories. This dataset has information on authors and their publications, including titles and abstracts, but unfortunately not full document texts. We have sampled this database to create a dataset with around 750 profiles and a total of around 14,000 documents. We believe this is a good representation of a set of expert profiles.

We decided to create a dataset of "poison" documents from another source; a collection of text files obtained from BBS (Bulletin Board Systems), grouped broadly by topic. Amongst these groups were collections of files categorized as "Anarchy" and "Drugs". We processed these documents in the same way as our profile data to create datasets with around 1500 and 500 documents respectively.

[1] www.RKBExplorer.com

7.2 Data Quality

For each experiment we first split our collection of academic publications randomly in two, holding back half the data for the creation of a corpus and using the rest of the data for training and testing.

We performed two experiments, the first was to determine the appropriate number of dimensions to retain in our LSA model. For this experiment we compared the performance of the corpus derived LSA model, with one built using the documents themselves, and another model built using the documents filtered to remove terms which are not present in the corpus. For the corpus derived model we looked at a model built from individual documents, as well as one built from profiles in this withheld data. At this stage no poison is added to the documents.

We used ten fold cross-validation, using the withheld documents as queries. Relevance is binary (i.e. a document is relevant or not) and will be determined by authorship of each query. This leads to very low scores, as many documents only have a single author, and if this author is not at the top of recommendation list then performance will be less than perfect. Additionally some experts who are not authors of the query document may nonetheless be relevant to it.

We use *Mean Absolute Precision* (MAP) to measure performance, which is the *Average Precision* averaged over all queries. The *Average Precision* is simply the precision of the top-r results of a query averaged over each relevant result at rank r. The equations are given below,

$$P(r) = \frac{|\{\text{relevant retrieved documents} <= \text{rank r}\}|}{r}, \quad (5)$$

$$AP = \frac{\sum_{r=1}^{N}(P(r)relevant(r))}{|R|}, \quad (6)$$

$$MAP = \frac{\sum_{q=1}^{Q} AP(q)}{|Q|}, \quad (7)$$

where R is the set of relevant documents, r is the rank, N is the number of relevant documents retrieved, and Q is the set of queries.

The results for our first experiment are shown in Figure 1. From these results we decided to use a model rank of 500 for good performance, but note that most of the performance is retained down to a model rank of around 100. Additionally we note that a corpus derived model LSA built on profiles performs better than one built on individual documents.

The second experiment involved testing the effect of profile poisoning on performance. For these experiments we used a model size of 100 and 500. Increasing amounts of poison was added to the documents. A poison level of 1 meaning that the number of poison documents added to a profile was equal to the number of documents already in the profile.

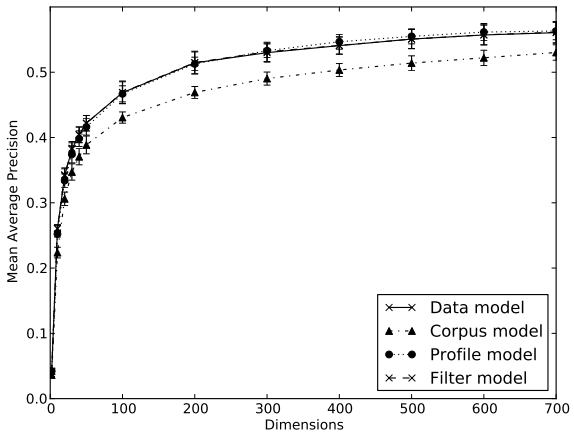

Fig. 1: Experiment one.

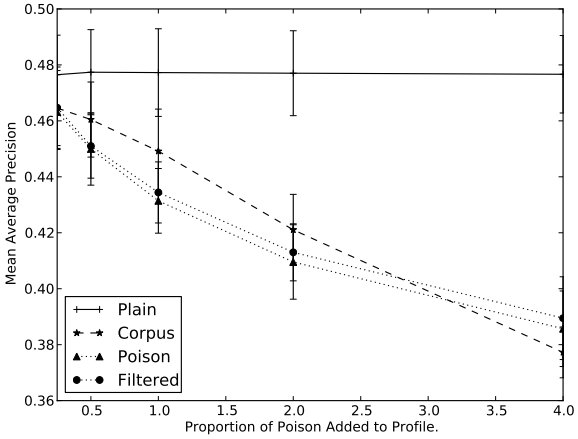

Fig. 2: Data Quality Experiment k = 100, Anarchy

7.3 Hiding Failure

Our privacy experiment is based on possible mining attacks that could be used to extract information about experts from the system. As attackers will not have direct access to profile vectors it does not seem sensible to look at the change in profile vectors with and without private projection, but instead to look at what information can be obtained through the query interface.

The scenario we consider is an attacker trying to find experts with interest in topics which may be controversial, embarrassing, or incriminating. We add poison to a certain proportion of profiles and attempt to detect these profiles by using a different set of poison documents as queries. In this experiment we add four times as many poison documents to each selected profile as the public documents that profile contains.

For these experiments we follow a similar approach to the performance experiments, except in this case success will be judged by how poorly the system performed in the experiment. Relevant profiles are all of the profiles which have had poison added to them, regardless of the specific documents used.

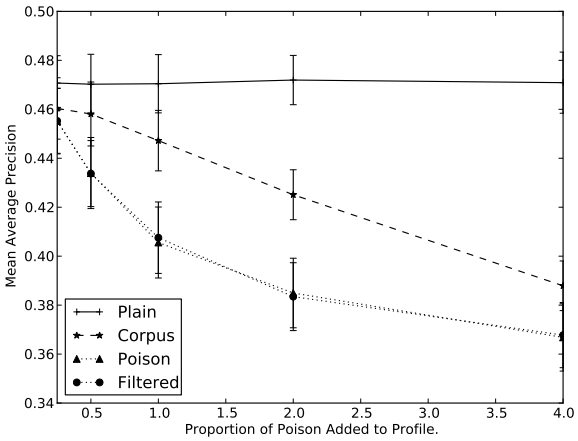

Fig. 3: Data Quality Experiment k = 100, Drugs

7.4 Results

Figure 2 shows the results for the data quality experiment using the anarchy dataset with a model size of 100, and Figure 3 shows the results of the same experiment with the drugs dataset. Figure 4 shows the results for the data quality experiment using the anarchy dataset with a model size of 100, and Figure 5 shows the results of the same experiment with the drugs dataset.

The results of the data quality experiment are roughly the same for both datasets. The quality of results degrades much more slowly when the higher rank model is used, and the corpus derived model performs the best on these tasks. It is interesting that simply removing words which are not in the corpus does not help maintain performance levels. This is probably because many of the important terms in the poison documents are present in the corpus.

Figure 6 shows the results for the hiding failure experiment using the anarchy dataset with a model size of 100, and Figure 7 shows the results of the same experiment with the drugs dataset. Figure 8 shows the results for the hiding failure experiment using the anarchy dataset with a model size of 100, and Figure 9 shows the results of the same experiment with the drugs dataset.

In each case the corpus derived model performs better than the poisoned and filtered models, which reach a MAP of almost 1 at certain points. While the corpus model does provide some level of privacy protection, it is slight, and much worse than the untainted profiles tested against the same queries. A higher level of privacy is provided using

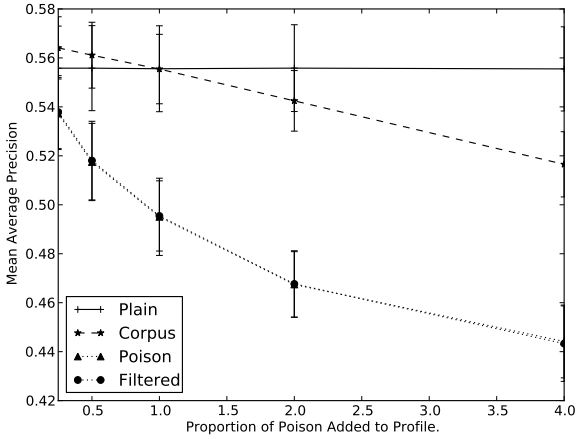

Fig. 4: Data Quality Experiment k = 500, Anarchy

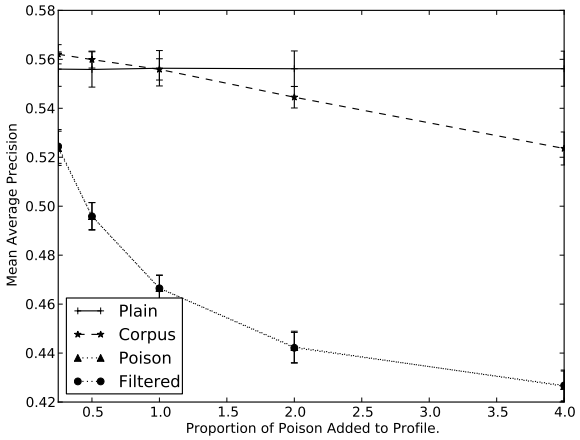

Fig. 5: Data Quality Experiment k = 500, Drugs

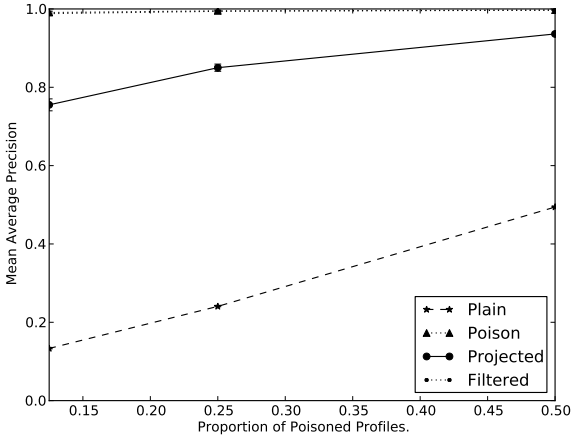

Fig. 6: Hiding Failure Experiment k = 100, Anarchy

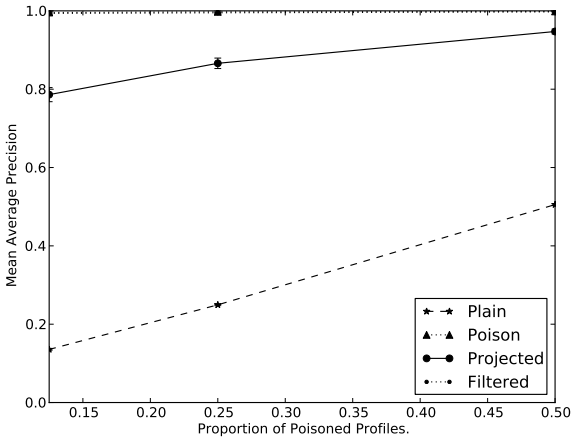

Fig. 7: Hiding Failure Experiment k = 100, Drugs

a lower-dimensional model.

8. Conclusion

In this paper we presented a specific instance of a privacy-preserving profiling problem relating to the Instant Knowledge expert recommendation system. Our main goals are the automatic generation of expert profiles, while preserving user privacy with little or no user feedback.

We presented a set of datasets and experiments which can be used to evaluate performance on this task. While our simple initial solution to the problem failed to hide private data adequately it significantly reduced the degradation of performance caused by polluting a profile with poison data.

We believe that the model failed to preserve privacy adequately as the LSA model was sufficient to represent most of the public and private information. The private information may be closer to public information than we had anticipated. While performance can be improved by reducing the rank of the profile matrix approximation, this affects the performance of the model on the data quality tasks.

8.1 Future Work

The simple privacy-preserving method we applied in this paper was largely passive. The intention was to create a model which was incapable of adequately representing the private information, which would lead to such data being filtered or reduced in magnitude.

Active filtering is more difficult without user feedback to guide the classification of documents or terms in a profile. We could, however, make better use of the profile corpus to train a filtering model. While private information may be different for each user, we should be able to make an educated guess about what makes a coherent profile.

For example we might not expect papers on sexually-transmitted infections to be present in the profile of a

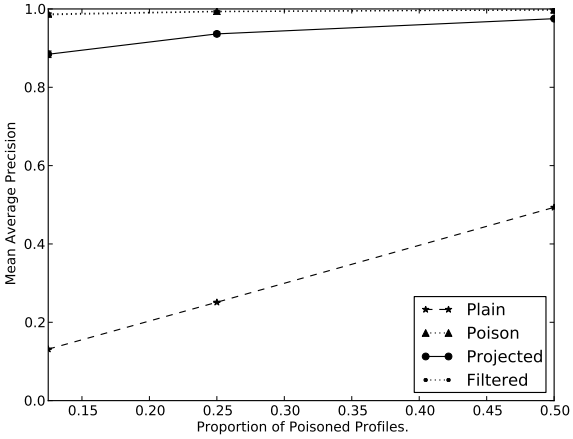

Fig. 8: Hiding Failure Experiment k = 500, Anarchy

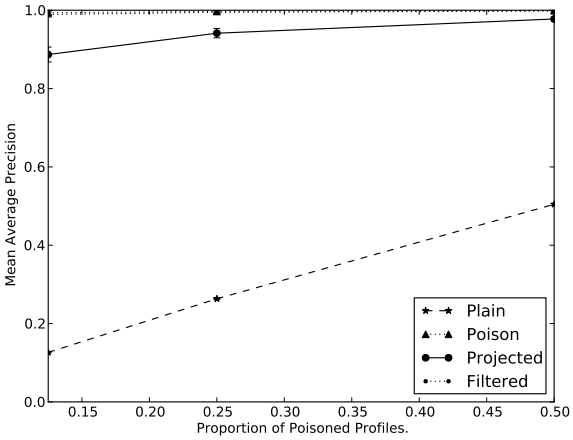

Fig. 9: Hiding Failure Experiment k = 500, Drugs

computer science researcher. While this researcher may have coauthored a paper on STIs, it is unlikely. Using the corpus we could calculate the probability of different interests co-existing in the same profile and use this information to filter out dubious interests.

The assumption of zero user input is perhaps too strong, and a wider range of techniques could be applied even if we have only a small number of labelled documents. We would also like to look at the issue of updating profiles with new documents, and how an existing profile can be used to preserve privacy.

Acknowledgment

The work reported in this paper has formed part of the Instant Knowledge Research Programme of Mobile VCE, (the Virtual Centre of Excellence in Mobile & Personal Communications), www.mobilevce.com. The programme is co-funded by the UK Technology Strategy Boards Collaborative Research and Development programme. Detailed technical reports on this research are available to all Industrial Members of Mobile VCE.

References

[1] S. Guha, B. Cheng, and P. Francis, "Challenges in measuring online advertising systems," in *Proceedings of the 10th annual conference on Internet measurement*, ser. IMC '10. New York, NY, USA: ACM, 2010, pp. 81–87. [Online]. Available: http://doi.acm.org/10.1145/1879141.1879152

[2] R. Singel, "Netflix cancels recommendation contest after privacy lawsuit," http://www.wired.com/threatlevel/2010/03/netflix-cancels-contest/, March 2010, retrieved on Wednesday 16th February 2011.

[3] C. Fox, "A stop list for general text," *SIGIR Forum*, vol. 24, pp. 19–21, September 1989. [Online]. Available: http://doi.acm.org/10.1145/378881.378888

[4] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, pp. 613–620, November 1975. [Online]. Available: http://doi.acm.org/10.1145/361219.361220

[5] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[6] T. Reichling and V. Wulf, "Expert recommender systems in practice: evaluating semi-automatic profile generation," in *Proceedings of the 27th international conference on Human factors in computing systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, pp. 59–68. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518712

[7] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis, "State-of-the-art in privacy preserving data mining," *SIGMOD Rec.*, vol. 33, pp. 50–57, March 2004. [Online]. Available: http://doi.acm.org/10.1145/974121.974131

[8] L. Sweeney, "k-anonymity: a model for protecting privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, pp. 557–570, October 2002. [Online]. Available: http://portal.acm.org/citation.cfm?id=774544.774552

[9] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ser. PODS '01. New York, NY, USA: ACM, 2001, pp. 247–255. [Online]. Available: http://doi.acm.org/10.1145/375551.375602

[10] E. Bertino, I. N. Fovino, and L. P. Provenza, "A framework for evaluating privacy preserving data mining algorithms*," *Data Min. Knowl. Discov.*, vol. 11, pp. 121–154, September 2005. [Online]. Available: http://portal.acm.org/citation.cfm?id=1095655.1095681