# Collaborative Filtering as an Investigative Tool for Peer-to-Peer Filesharing Networks

Matthew Edwards and Awais Rashid
Security Lancaster
School of Computing and Communications
InfoLab21
Lancaster University
Lancaster, Lancashire, UK
m.edwards7@lancaster.ac.uk
marash@comp.lancs.ac.uk

## ABSTRACT

The volume of illegal material on file-sharing networks poses a challenge for investigators attempting to police such networks. We propose a novel approach that automates the resource intensive task of identifying previously unknown files of interest amongst hundreds of thousands of files shared on such networks. We also describe how this approach could be used to identify clusters of peers that might be closely related to each other, either as part of a syndicate, or as multiple personae of the same individual.

Our approach is based on the collaborative filtering techniques typically used in recommender systems. In this study we find that we can successfully make use of collaborative filtering techniques to find new media belonging to specific categories of interest to an investigation of a peer-to-peer network, without having to examine filenames or file contents. We also find evidence that distance metrics from collaborative filtering could be useful in the clustering and identification of peers on file-sharing networks. Additionally, we describe an unsuccessful attempt at using collaborative filtering to predict the future file-sharing behaviour of peers.

## I INTRODUCTION

Peer-to-Peer file-sharing networks such as Gnutella and BitTorrent pose a unique challenge to law enforcement officials. As well as facilitating the illegal distribution of copyright protected material such as games, films and music, such networks are being used for the purposes of serious organised crime, including the distribution of child abuse media and other illegal pornographic material [5]. In this paper we describe how collaborative filtering techniques, originally developed to select user-appropriate content from online and commercial datasets, can be applied to support the policing of file-sharing networks.

Law enforcers typically attempt to combat illegal activity on file-sharing networks through manual monitoring and investigation with tools such as RoundUp [7]. These tools mostly require enforcers to identify specific files of interest they wish to locate, or else connect to individual peers whose shared files they wish to browse. Given the volume of traffic generated by file-sharing networks and the significant portion of this which has been identified as illegal material [5,9], such manual investigation is unsustainable. Therefore, automatic systems to filter and prioritise targets for investigation are needed [6]. The aim here is not to provide forensic evidence for prosecution, but to assist enforcers in finding and prioritising leads for further investigation.

Such investigative aids are required in a number of scenarios. Firstly, enforcers tracking down illegal material on peer-to-peer networks need support in finding particular categories of content amongst terabytes of data. This is an extremely challenging task, as notwithstanding the scale of the networks, file-sharing content is often untagged, mis-categorised or even purposefully disguised, particularly where illegal material is concerned. Secondly, investigators might wish to model and predict the file-sharing behaviour of peers so as to detect and target the most prolific offenders or those peers who are at risk of committing an offence. Finally, there is increasing interest

ISBN: 978-1-62561-001-0

in attempting to identify relationships between peers (in order to uncover organised crime rings) and resolve multiple identities held by the same individual (in order to conflate such identities into a single targeted investigation, hence optimising resources that may otherwise be dedicated to multiple, superficially isolated, cases).

We propose a collaborative filtering approach to address the above investigative scenarios. As well as outlining how collaborative filtering may in general be applied to traces of peer-to-peer networks, we extend the underlying assumptions of collaborative filtering in designing and evaluating three possible applications which could be of use to investigators. We investigate how a collaborative filtering recommender system can be used for the discovery of new illegal media without need for processing of filenames or file content; we attempt the prediction of peers' sharing behaviour for the purposes of preventative intervention and we evaluate the suitability of a collaborative filtering distance metric for clustering and identification of peers based only on their file-sharing history. We apply our techniques to samples from two leading peer-to-peer networks and discuss how properties of these networks impact on the operation of these methods.

The novel aspects of our study are as follows:

- This is the first work to study the applicability of collaborative filtering techniques for the investigation of file-sharing networks. To date, collaborative filtering techniques have been successfully applied in this domain where users' behaviour can be readily bounded, e.g. limited to a specific and identifiable subset of file-sharing traffic such as MP3 files, or through explicit rating of items by users. This is the first work to target collaborative filtering for such unbounded multi-user settings.
- We evaluate whether our approach enables categorisation of unknown files in a large unbounded space without recourse to examining their content or relying on filenames. This would enable investigators to gather new target files from a large sample under investigation, simply requiring known examples of the same category of file.
- We evaluate whether our approach enables investigators to predict the file-sharing behaviour of peers. This would enable investigators to prioritise targets for further investigation.
- We evaluate whether our approach makes it possible to cluster related peers together and/or superimpose multiple peers on to the same identity. This would enable investigations of organised crime on file-sharing networks and contribute to building a more accurate profile of users who may be hiding behind multiple peers on the same (or multiple different) networks.

In performing this study we build upon earlier work which has identified characteristics which indicate the suitability of file-sharing networks and the distribution of certain illegal media therein for this kind of analysis [3,5]. We also aim to complement studies using collaborative filtering on file-sharing networks for the purposes of recommendation and rating [13,14,17] and approaches to media classification involving linguistic analysis of filenames [4].

In Section II we provide an overview of certain past approaches to aiding law enforcement in the monitoring of file-sharing networks, and explain the general operation of collaborative filtering algorithms. We then move on to describe how this theory can be applied to file-sharing networks in Section III, and detail how such methods are of benefit to monitoring authorities. In Section IV we evaluate three of these applications against large datasets drawn from the Gnutella and BitTorrent networks, presenting results along with their analysis. We conclude in Section V by summarising the findings and outlining areas where further research seems most appropriate.

## II BACKGROUND

There have been previous efforts to assist law enforcers in monitoring peer-to-peer file-sharing. Well known tools in this domain, such as RoundUp [7], already include useful features such as a geolocation capability to enable officers to remain within their jurisdiction [8], and law enforcers often exploit the discovery tools built into peer-to-peer protocols to track down offending material by filename or hash value. However, such tools are merely aids to manual investigation and identification of illegal material and do not provide for automated detection of leads.

The work of Chow *et al.* [2] expands on this discovery approach to provide an automated system for scraping torrent-sharing fora and investigating BitTorrent tracker information through specified conditions. This is a more promising approach, but appears limited to presenting or acting on summaries of the information directly provided by trackers.

Hughes *et al.* [4] identified the challenge of detecting child abuse media on peer-to-peer file-sharing networks when confronted with the adoption of obscure, evolving domain-specific terminology by offenders. The authors addressed this through the application of natural language processing techniques. This approach led to promising results, but is limited by a reliance on file names reflecting file content and the use of chained connections to detect new terminology. In an alternative approach presented later, we present a competitive method based on user and file association which should bypass these limitations and prove generalisable to other media types.

Although we are unaware of any other examples of collaborative filtering techniques being applied for the purpose of aiding investigation, there is supporting evidence that makes a strong case for success in this area. Research aimed at optimising the performance of file-sharing networks has revealed a strong degree of clustering around similar interests [3]. Similarly, work studying particular criminal behaviour on file-sharing networks [5] shows that most peers sharing illegal pornographic material will mostly share only such content, boding particularly well for the effectiveness of collaborative filtering techniques in this domain, as such techniques tend to perform poorly on users with eclectic preferences. Alongside this are examples noting the successful application of collaborative filtering techniques to assist file sharers searching for content [12,17] and for gathering information on music popularity [14].

## 1 COLLABORATIVE FILTERING

Collaborative filtering techniques are well established in online recommender systems. They are employed to enhance a user's experience by sifting through the vast quantities of content a site or store offers and finding items a user is likely to enjoy. This serves the dual purpose of promoting customer satisfaction and maximising the effectiveness of advertising.

The way these techniques work can vary in a number of details, but the principle of the method remains the same. Essentially, the technique relies on the assumption that *if user X has previously enjoyed items which user Y also enjoyed, user X is likely to enjoy other items user Y has enjoyed.* The concept of whether a user *enjoyed* an item of content can be explicitly stated - by asking the user to rank items - or silently inferred from those items a user is seen to access or purchase.

In a typical collaborative filtering scenario, there is a set of $n$ users $U = \{u_1, u_2, ..., u_n\}$ and a set of $m$ items $I = \{i_1, i_2, ..., i_m\}$. Each user $u_k$ has a history of enjoyed items $Iu_k$ which is a subset of the available items $I$. The task of a collaborative filtering algorithm is to decide on a subset of $I$ that a particular *active user* $u_k$ is likely to enjoy based on the content of $Iu_k$.

It accomplishes this by scanning $U$ for any users whose history of enjoyed items overlaps with that of $u_k$, finding a subset of $U$ which we could call the 'neighbours' of $u_k$. Taking the union of these neighbours' histories results in a set of items which, given the assumption above, it might be reasonable to expect the *active user* to enjoy. This can be (and almost invariably is) then improved by ranking the members of this set based on which items were most common, or more common in this group than in $I$ generally. Where user ratings are available, more highly rated items might be moved up the recommendation list. Several approaches [11,15] modify this ranking of items by considering how similar to the history of the *active user* is the history of the user recommending an item.

As mentioned, several of these details can vary and there are many subtleties to the application of this general method to specific purposes. A comprehensive review of collaborative filtering literature is provided by [15].

## III PROPOSED APPROACH AND APPLICATIONS

It is not difficult to see that certain collaborative filtering concepts map readily onto ones drawn from the operation of file-sharing protocols. Rather than users, we shall talk of a set of $n$ peers, $P = \{p_1, p_2, ..., p_n\}$, which are machines participating in file-sharing on a particular network. Rather than 'items' we shall talk of a set $F$ of $m$ files, $F = \{f_1, f_2, ..., f_m\}$, being shared on the network. Rather than an *active user* we shall talk of an *active peer* for whom we wish to make recommendations. These are only superficial alterations to the general approach outlined in Section II.

In ordinary file-sharing traffic, there is no rating of files, only querying, downloading and sharing. Therefore we must consider collaborative filtering techniques which do not rely on explicit rankings, but for which it would suffice to know whether a peer has previously shared a certain file. For each file, we shall know whether or not a peer has shared it, which is to be considered analogous to them 'enjoying' the file. A leading method for this type of binary data is the Lightweight Collaborative Filtering Method for Binary Encoded Data (LCFBED) [18]. As we will apply this method extensively, it will be instructive to outline it here.

Given an *active peer* $p_k$, who is already known to be sharing some files $Fp_k$, we find all members of $P$ who are observed to also be sharing a file in $Fp_k$. That is, for all $p_i \in P$, if $Fp_i \cap Fp_k \neq \emptyset$, then $p_i$ is a member of $Pp_k$, the group of 'neighbour peers' which share some preferences with the *active peer* $p_k$.

Next, we collect all distinct files which are known to be shared by one of the neighbour peers in $Pp_k$ and which are not in $Fp_k$. This leaves us with a set $Fq$, where

$$Fq = \bigcup_{p_j \in Pp_k} Fp_j - Fp_k$$

This is the set of files which the *active peer* might be interested in, though they are as yet unsorted. In order to sort them, the LCFBED method gauges the level of similarity between peers. To do this we must first outline the concept of a peer being known to share a file:

$$owns(p_x, f_y) = \begin{cases} 1 & \text{if peer } p_x \text{ shared file } f_y \\ 0 & \text{otherwise} \end{cases}$$

and then include this in the definition of a measure of *closeness* between any two peers $p_x$ and $p_y$.

$$closeness(p_x, p_y) = \sum_{f_j \in Fp_x \cap Fp_y} 1 + \frac{1}{\sum\limits_{p_i \in P} owns(p_i, f_j)}$$

This measure counts the number of files the two peers have in common, but it also adds a weight to each of those files which inversely reflects how common that file is in the network in general. This means that peers who have more unusual files in common are regarded as *closer* than peers who hold a relatively widespread file in common.

This *closeness* measure provides us with a general understanding of the strength of a relationship between two peers. However, where we want to compare multiple peers to an *active peer*, we find it useful to talk in terms of proportionate closeness, where each $p_j \in Pp_k$ has a relationship measure to $p_k$ gauged in relation to the *closest* of $Pp_k$. We will refer to this as the *similarity* measure between an *active peer* $p_k$ and any $p_x$.

$$sim(p_k, p_x) = \frac{closeness(p_k, p_x)}{max(closeness(P_{pk}, p_x))}$$

Using these components, we can then construct a score for each $f_j \in Fq$ which reflects how likely the *active peer* $p_k$ is to find $f_j$ enjoyable by taking into account not only the number of neighbours which share a given file, but also how *similar* each neighbour is to the *active peer*, and also the *number* of files which said neighbour is recommending.

$$score(f_j) = \sum_{p_i \in Pp_k} owns(p_i, f_j) \frac{sim(p_i, p_k)}{\sum\limits_{f_k \in Fq} owns(p_i, f_k)}$$

This scoring system assigns most weight to those files recommended by people who are recommending little else. That is, the majority of the files they are sharing are those which they have in common with the active peer. This minimises the impact of 'super sharers' or seeds in the file-sharing network, who tend to share large numbers of files. Though these peers still contribute to the recommendation of a file for $p_k$, they contribute much less than peers who share mainly the same type and number of files as $p_k$.

The method described above allows us to rank the files in $Fq$ for recommendation to the peer $p_k$. This might form the basis of a user experience improvement service for a file-sharing network, but as it stands would be of little use to law enforcement agencies. It is through re-application of this method and extension of the classic collaborative filtering assumptions that the potential utility emerges, as we shall describe in the three cases below.

## 1 MEDIA CLASSIFICATION

Hughes *et al.* [4] observed that discovering new illegal pornographic material on file-sharing networks is a challenging task due to the obscure domain-specific terminology used by offenders. They put forward an approach which uses automated linguistic analysis to identify new child abuse media. This approach did prove successful, but is limited by a reliance on filenames reflecting file content, which is especially problematic when faced with intentionally obfuscated filenames.

We propose a new method, based on the LCFBED collaborative filtering algorithm, which should allow for media classification based solely on hash value and association with peers. The key insight is that, for the purposes of the algorithm, the *active peer* is merely a means of identifying a subset $Fp_k$ of $F$. It is entirely possible to modify this step so that this subset is instead selected by some manual or automatic process to contain items of a particular category. Using the observation that file-sharing traffic tends to cluster around interest [3] and that such clustering is especially strong in the case of those sharing illegal pornographic material [5], it seems to follow that law enforcers could select a group of files representative of — for example — child abuse media, and use the collaborative filtering method to identify a list of previously unseen files which are likely to be of the same nature.

The LCFBED method is particularly appropriate for this application, as its operation allows for an inclusive list of files of a certain type without penalising effectiveness — that is to say, as it does not count a *lack* of a file in $Fpk$ against a peer in creating a measure of *closeness* to the *active peer*, as some other collaborative filtering techniques do, it is possible for arbitrarily long lists of known files to be utilised.

It is worth making clear that in this we slightly alter the classical collaborative filtering assumptions as defined in Section II. We now assume that, for a set of files $Fn$ created to reflect some category, *peers that are sharing the files in* $Fn$ *will also share files of the same type.* In essence, this method uses peer association to find previously unknown examples from a set of known examples of a type of file, with no need to examine the contents or filename of a file.

## 2 BEHAVIOURAL PREDICTION

The fundamental prediction produced by the technique previously described is as to what the *active peer* $p_k$ will 'enjoy', based on what it is seen to be sharing and what other peers who are sharing the same files are also sharing. However, it is not unreasonable to suggest that this might also be an appropriate method for predicting what $p_k$ will soon share, given that the recommendations coming from the method are formed by looking at which files are being made available for download by peers with whom $p_k$ is likely[1] to have some previous, possibly ongoing, interaction.

It is again worth making clear that here we would be extending the classical assumptions of a collaborative filtering technique, so that we instead assume that *if peer* $p_k$ *is sharing files* $Fp_k$ *and peer* $p_y$ *is also sharing files* $Fp_y$ *such that* $Fp_k \cap Fp_y \neq \emptyset$*, then peer* $p_k$ *is likely to* o*btain and* s*hare files from* $Fp_y$*, the other files peer* $p_y$ *is sharing.* This is a greater assumption than in the classical case, as we infer not only desire, but also successful attainment and distribution.

If this assumption holds up under scrutiny then it could make some contribution to understanding of how files spread through file-sharing systems. It would also allow law enforcers to make interventions ahead of time to prevent access to illegal content, perhaps at the level of targeted warnings.

## 3 DISTANCE METRIC

This application focuses on the *closeness* component of the LCFBED collaborative filtering algorithm. With it we have a measure of how closely identifiable two peers may be. This measure could by itself be of use to law enforcement officials in link analysis activity, where enforcers examine the known connections between suspects to gain an understanding of criminal networks. This technique seems particularly well-suited for the sort of organised crime which is frequently seen online [1].

An additional application lies in that when two peers are seen to have high *closeness* measures it may be taken as an indication, alongside their geographical location and other factors, that they are closely linked in another manner. This is grounded somewhat in that the *closeness* measure makes use of exact hash comparisons, so large values indicate that a number of the exact same versions of files have been shared by the two peers. To take this approach to the extreme, it seems that if two peers have a particularly high *closeness* measure, this could be a useful trait in deciding whether they are in fact the same person or physical machine - either wilfully disguising their identity, or having it masked by properties of the network. This identification property could aid law enforcers in overcoming criminal countermeasures to traffic monitoring.

[1] As their histories overlap, although the exact implications of this are protocol-dependant.

Although this application does not apply the full collaborative filtering technique, the way we are interpreting the *closeness* measure does differ from how it is used classically. Normally, it might be said that the *closeness* measure reflects how similar the file-sharing history of peers is; in our usage of the measure, the assumption is that *similarity of file-sharing habits reflects a more general similarity or connection between individuals.*

## IV EVALUATION

In this section we aim to test the correctness of the hypotheses outlined in Section III by applying them to sample data drawn from observations of real file-sharing traffic. The data is drawn from two sources: a Gnutella-based file-sharing network and a BitTorrent-based file-sharing network. The decision to use multiple networks was made so as to better test the generality of the proposed applications.

Five datasets are used in total; three collected from the BitTorrent network in the months of October, November and December of 2009, the two others collected from the Gnutella network in March and April of 2010. The BitTorrent network traffic was collected by polling popular trackers for information, with torrents being drawn from popular torrent-listing sites. Gnutella network traffic was gathered by introducing nodes into the Gnutella network which became 'super peers' and recorded the query-response messages they were tasked with transmitting.

For the sake of brevity, the datasets used in the evaluation will be referred to below as the combination of a three-letter code for their network of origin (`gnu` for Gnutella, `bit` for BitTorrent) and the month and year of their collection. For example, `gnu0310` refers to the Gnutella dataset from March, 2010.

In order to process the datasets in an origin-agnostic manner, the observations were aligned to a common format, outlined in Table 1. This format was chosen to represent the minimal information one might expect from any recording of file-sharing on a file-sharing network.

| Field | Field Type | Example |
|---|---|---|
| *IPAddress* | Text | 148.88.227.226 |
| *ObservedAt* | Timestamp | 2009-03-02 00:45:32 |
| *Filename* | Text | Soul Eater 2.mov |
| *FileHash* | Text | D3A401C565B35E5100A1D |

Table 1: The format to which the trace datasets were aligned.

Together, the *IPAddress* and *ObservedAt* fields allow us to reference any particular observation of files. The *FileHash* field allows us to determine file or torrent equality[2]. We retain the *Filename* of each observation alongside the hash so as not to sacrifice human-comprehensible identifiers.

The datasets differed in size and distribution according to their origin. The Gnutella traces consisted of 4.1 and 4.6 million examples of the form in Table 1, whereas the BitTorrent traces numbered some 3.2, 2.8 and 3.0 million examples. The resulting difference in size was some 300,000 extra examples for the BitTorrent network, though this makes up only 1/30th (3.3%) of the sample size.

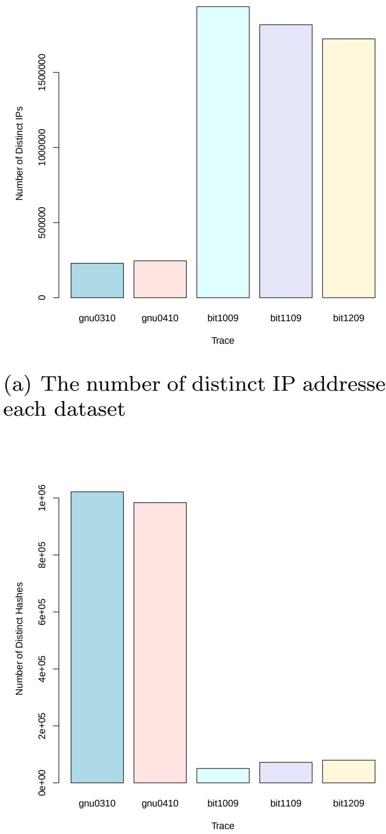
Exploratory analyses revealed that the relationship between users and files in the two networks is markedly different. As seen in Figures 1(a) and 1(b), the Gnutella traces have fewer distinct IPs (and therefore users) and many more unique files, whereas the inverse holds true for the BitTorrent samples.

These differences can be partially attributed to the different manner in which the observations were sampled, although this itself is reflective of a difference in the underlying protocols. The Gnutella network operates on a scoped 'neighbourhood' principle, and thus it is not particularly surprising that mostly the same range of IP addresses is seen, as these would be the peers in the neighbourhood of the 'super peers' collecting observations. That the Gnutella network datasets have significantly greater hash diversity than the BitTorrent network reflects that the observations of the torrents were gathered from a popular aggregation site rather than through sampling typical traffic. This is also evident in the large number of distinct IP addresses in the BitTorrent network, which demonstrates the popularity of these aggregation sites.

## 1 STUDY DESIGN

We aim to test all three of the applications proposed in Section III. As such, three separate evaluations were carried out, though all use the same datasets.

[2] Note that the Gnutella network hash is a (reported) true hash of the whole file, whereas the BitTorrent hash is the protocol's *info_hash* field. For our purposes, however, this incomplete hash will serve as a unique identifier, as the *info_hash* identifies an individual torrent. Our interest is in identifying a particular 'sharable unit' in the file-sharing network, not ensuring that units with identical content are matched.



(a) The number of distinct IP addresses in each dataset

(b) The number of distinct file hashes in each dataset

Figure 1: Counts of distinct hashes compared with counts of distinct IP addresses show an effect for the network of origin.

## 1.1 MEDIA CLASSIFICATION

The aim here is to test whether the collaborative filtering method we described can identify and recommend files of the same 'type' as an input collection. Whilst the work of Hughes *et al.* [5] suggests that this method might be highly effective in searches for child abuse media, the obscure nature of such items makes evaluation difficult without domain experts. Instead, we evaluate the performance of this media classification method in searching for three types of files: pornography, piracy software and popular music. Pornographic files were selected as a testable proxy for illegal pornographic material. Software related to the piracy industry (by which we mean to include key generators and application versions labelled as 'cracked') was chosen as an example of readily-identifiable illegal files and popular music files were chosen to investigate the effectiveness of the method on more widespread and openly shared files. Examples from each of these categories were collected from each dataset through keyword searches, though any system of selecting examples would have sufficed. These collections were then used to identify new media from the datasets using the LCFBED method. The filenames of the predicted items, along with those of an equal number of randomly selected files introduced as a control level, were then presented to human evaluators for classification. They were instructed to classify the filenames into either one of the original selected-for categories, 'Unknown' where the filename was unclear as to the content or 'Other' where it was from another category. Results for the predicted files were then compared to those for the randomly selected baseline to test for significant improvements.

## 1.2 PREDICTING BEHAVIOUR

To evaluate how well our method predicted actual behaviour, we implemented a form of cross-validation suitable for time-series data. First, we identified those peers which appear more than once in each dataset. Then we ordered the observations of their file-sharing behaviour chronologically.

For each observation, we made a number of predictions for the peer based on the network's file-sharing history as it would appear at that timestamp. We then checked whether these predicted files appear in the peer's future observations, and generated values for precision and recall. Repeating this process for each but the last observation, we can then calculate the mean precision and recall for each set of predictions, along with standard deviation of these statistics and an F1-score averaging them.

## 1.3 DISTANCE METRIC

The hypothesis under question here is that users identified as having high *closeness* values are similar in other regards which may be of interest to investigators. Such similarity, however, is hard to quantify and thus difficult to evaluate. Instead we measure the degree to which peers have high *closeness* measures to themselves across time. Closeness measures between the same peer at different points in time are taken, and compared to the *closeness* between that peer and its neighbours at each time. As a peer will undoubtedly be similar to themselves, we treat this as an initial gauge of plausibility of this method. If peers demonstrate a significantly higher *closeness* value against past observations of themselves (identified by IP address), then we demonstrate some validity in using *closeness* as a distance metric between P2P users.

## 2 THREATS TO VALIDITY

One of the main threat to the validity of this study is that the unique identification of peers by IP address in either file-sharing network is imprecise. There are numerous reasons why this might be the case. In the first and simplest case, dynamic IP address allocation schemes mean that the same machine can be assigned different addresses at different times, without any particular intent on the part of the operator. Secondly, security-conscious peers may make use of proxy services to alter their reported IP address and intentionally obscure continued network activity. Thirdly, for the BitTorrent network traffic there remains the titular issue discussed in [10], that some trackers merely accept proposed IP addresses from peers, allowing for false incrimination of innocent network devices, and also the related issue that misreporting trackers could well disguise the identity of their peers.

Within the BitTorrent and Gnutella protocols, there is a provision for unique identification of file-sharing clients. It is possible for this identifier to be altered by the client, but in practice this is rarely done by peers on either network, and it could prove a better identifier than IP address. We refrain from using such identifiers, however, as their presence in other network protocols is not guaranteed, and we aim to demonstrate the general applicability of the collaborative filtering method in file-sharing networks, which such a reliance would limit.

Instead, we inspect the difference between two *closeness* measures for a peer identified by IP address. We compare it to later observations of that IP address and also to 'neighbour' peers who have some files in common. If IP address proves to correlate with high *closeness*, we may induce that this is because the client at that address is the same client.

Another threat is posed by the question of equality across protocols. In the BitTorrent protocol, several files might be bundled together into a torrent, which is the basic sharable item, whereas on the Gnutella network individual files are considered the basic sharable item. This could lead to some confusion, so we clarify that our position is to consider the basic sharable item of a protocol as a 'file'. In support of this, we point out that a file shared on the Gnutella network might well be a zipped or archived collection of other files, and similarly many torrents on the BitTorrent network might contain only one file.

## 3 RESULTS

### 3.1 MEDIA CLASSIFICATION

For each dataset, nine examples of a file belonging to each category (of Pornographic, Pop Music and Illegal/Cracked Software) were selected through keyword searches. Using one, four and finally all nine files, the LCFBED method was used to identify ten other files which were hypothesised to be of the same type. To evaluate the accuracy of this hypothesis three human reviewers were asked to place the filenames of the resultant files, along with an equal number of 'control' files selected at random, into one of five categories (three being the original categories, the two others being 'Unknown' for unreadable filenames and 'Other' for miscellaneous other file types).

The level of inter-annotator agreement was fairly high, with an average agreement rate of 70.6% over all categorisations. Agreement on which files were pornographic in nature was 71%, the agreement for which files were illegal software was slightly lower at 67% and agreement on which files were popular music was higher, at 80%. Most disagreement came from whether a reviewer marked a file as 'Unknown' or 'Other', with agreement rates of 19% and 44% respectively.

The results across all datasets are presented in Figure 2. Accuracy is presented as divided between the three file categories, and further divided to reflect whether the classification algorithm was given one, four or nine files previously selected as being of that category in order to generate its new predictions. The files reviewers marked as 'Unknown' are included as grey bars, demonstrating a possible higher bound on accuracy, as such files could be correctly predicted but in a manner undetectable to our reviewers. These bounds were not included in tests of significance. The baseline is the frequency with which reviewers marked randomly selected files as belonging to the indicated category, giving a measure of the prevalence of that filetype in the network. The overall accuracy was 41%.

A Welch two-sample t-test (at the 5% significance level) was carried out, and a statistically significant effect (p=0.033) was found for the accuracy of the LCFBED classification method as compared to the randomly selected markup across all datasets. In-

ISBN: 978-1-62561-001-0

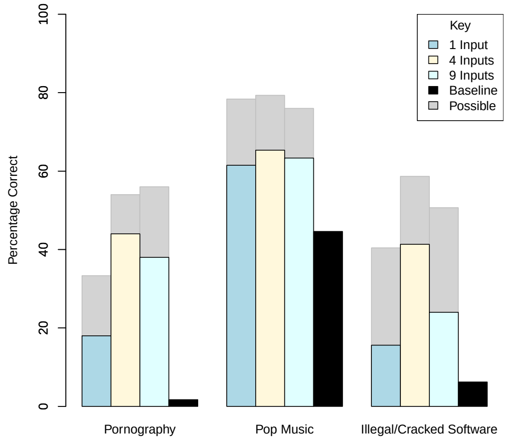
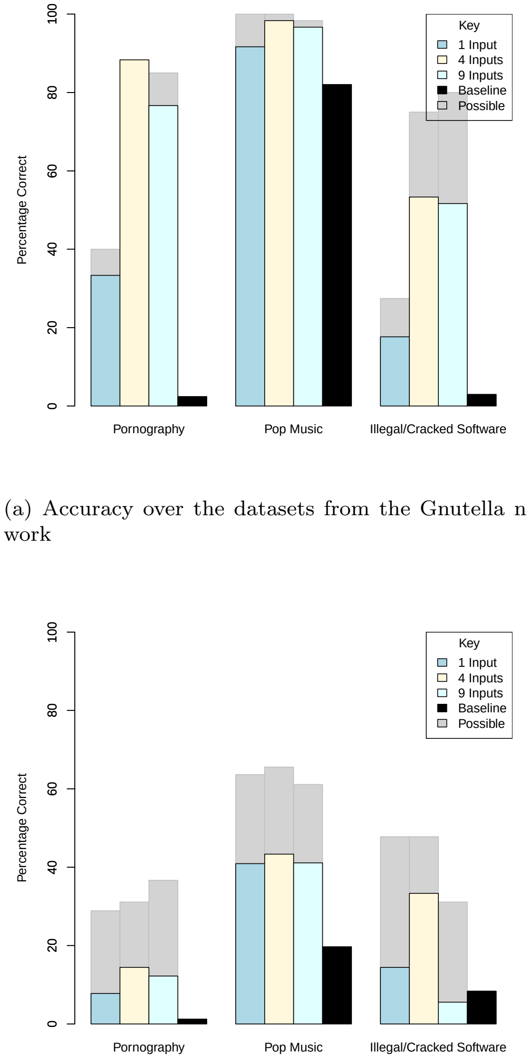


Figure 2: Accuracy over all datasets, in classification categories and number of keyword-selected input files used to generate predictions.

spection of individual dataset results revealed a likely disparity between the Gnutella and BitTorrent samples, so the t-test was repeated for these subsets, revealing a significant effect for the Gnutella datasets (p=0.036, mean accuracy of 67.5%) but not for the BitTorrent datasets (p=0.076, mean accuracy 23.68%). The disparity in the effect for the networks may be linked to the relative distribution of users and files described previously in Figure 1. Results for accuracy within each network are presented in Figures 3(a) and 3(b).

Within file categories, the effect was strongest for popular music, followed by pornographic material. The effect for illegal or cracked software was the weakest, and was not statistically significant. This may indicate that certain file-types are more amenable to classification through this method, or it may indicate that this category was too loosely defined.

Some 17% of predictions were labelled as ‘Unknown’ by reviewers, indicating that they were unable to discern the type of contents a file may have from the filename, indicating a significant motivation for the employment of this method, as such files are resistant to the methods proposed by [4]. As with other properties, there was a distinction between the percentage of ‘Unknown’ files in the Gnutella and BitTorrent datasets. The Gnutella network predictions were 9.6% unknown, whereas the BitTorrent network predictions were 22% unknown, indicating a possible variation in naming practices between the networks.



(a) Accuracy over the datasets from the Gnutella network

(b) Accuracy over the datasets from the BitTorrent network

Figure 3: Accuracy of predictions for each network of origin.

## 3.2 PREDICTING BEHAVIOUR

10 000 observations of peers which appeared more than once were drawn from each dataset. Each observation included a list of files the peer is known to be sharing at that point. For each observation, predictions of new files were made for that peer using the information visible chronologically prior to that observation, mimicking the situation the recommender would face in the real world. For some peers, there are no predictions to be made; this will be the case where there are no other peers at this point which have shared the same files and have also previously shared other files. Such observations are discarded, as they are not useful for evaluation purposes. Peers which do not gain any new files in further observations are also discarded, for the same reason. This filtering resulted in a total of roughly 1000 observations from each dataset being considered, except in the anomalous case for gnu0410, which had 5000. The intent of this filtering was to avoid the situation described by [14], where a large proportion of predictions were identified as correct simply through correctly predicting no new files.

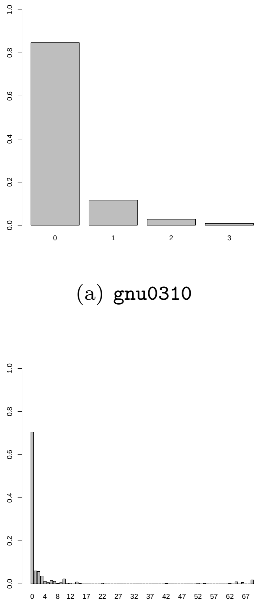
For each observation, the number of predictions validated by later observations of the same peer was recorded. This count was used to establish precision and recall values for the prediction mechanism, where precision is the number of correct predictions over the total number of predictions made and recall is the number of correct predictions over the total number of files later observed. Standard errors for these statistics were also calculated. Precision and recall were then combined to give an F1-score for the accuracy of the prediction [3]. The results are summarised in Table 2.

The low accuracy of predictions (including no correct predictions for the BitTorrent network samples) here demonstrates a strong negative result for this predictive method. Very few of the predictions made were observed to be true. For the peers from the Gnutella network, where some predictions were observed to be correct, most observations of peers resulted in no correct predictions, approximating an exponential distribution, as seen in Figure 4.

Precision measures could perhaps be increased by paring down the number of predictions made for each observation by selecting the top $n$ ranked predictions, but such an alteration would only be beneficial in the case of a high recall prediction method, which this is not. Larger samples are likely to increase the number of correct observations, but at the cost of lower overall recall. We reflect on this further in Section 4.

| Dataset | Precision | SE | Recall | SE | F-Score |
|---|---|---|---|---|---|
| gnu0310 | 0.004 | 0.0016 | 0.00295 | 0.00029 | 0.0034 |
| gnu0410 | 0.00643 | 0.00024 | 0.01667 | 0.00057 | 0.00928 |
| bit1009 | 0 | - | 0 | - | - |
| bit1109 | 0 | - | 0 | - | - |
| bit1209 | 0 | - | 0 | - | - |

Table 2: Summary of prediction results. BitTorrent datasets showed no correct predictions.



(a) gnu0310

(b) gnu0410

Figure 4: The number of correct predictions made for an observation of a peer in each Gnutella dataset.

## 3.3 DISTANCE METRIC

A selection of 5,000 observations of peers was taken from each network. For each observation, *closeness* measures were calculated between the peer (with the files it was reported to have shared up to this observation) and any previous observations of the same peer. Closeness measures were also calculated between the peer and any other peers which had a file in common. A Welch two-sample t-test was carried out to judge the significance of variance between the two samples.

The results are presented in Table 3. The results

[3] An F1-score is calculated as $F = 2 \cdot \frac{precision \cdot recall}{precision + recall}$

ISBN: 978-1-62561-001-0

demonstrate a clear distinction between the average *closeness* score for every peer in relation to previous observations of itself, compared with the average *closeness* score for other peers it has shared files with. This demonstrates that *closeness* measures are strongest within the same peer (as identified by IP address), indicating that high *closeness* will correlate with IP address and thus lending some weight to the original suggestion of using *closeness* as a means of identifying individuals.

| **Dataset** | **Self Closeness** | **Neighbour Closeness** |
|---|---|---|
| gnu0310 | 25.65 | 1.02 |
| gnu0410 | 345.63 | 1.01 |
| bit1009 | 166.42 | 1.01 |
| bit1109 | 249.36 | 1.02 |
| bit1209 | 263.56 | 1.02 |

Table 3: Summary of closeness comparison results

Interestingly, the average closeness of each peer to its neighbours was highly consistent across datasets. Examination of the standard errors for each dataset (presented in Table 4) shows that the parameter estimate of a mean closeness to a peer's neighbours of 1.02 is highly likely to be accurate across networks and datasets. The standard error for mean closeness values between a peer and past observations of that peer is higher, and Table 3 shows that inter-dataset variability in mean self closeness was quite high.

| **Dataset** | SE (Self) | SE (Neighbours) |
|---|---|---|
| gnu0310 | 0.122 | $3.69 \times 10^{-4}$ |
| gnu0310 | 0.429 | $1.30 \times 10^{-4}$ |
| bit1009 | 0.368 | $1.34 \times 10^{-5}$ |
| bit1109 | 0.396 | $2.04 \times 10^{-5}$ |
| bit1209 | 0.384 | $1.83 \times 10^{-5}$ |

Table 4: Summary of standard errors for the self and neighbour closeness means.

## 4 ANALYSIS

Considering first of all the results from Section 3.1 the indication is that collaborative filtering can be applied usefully to classify previously unknown media. Specifically, we find that the method works well on the Gnutella network, and in particular on popular music and pornography files. It would seem that our modified assumption that *peers that are sharing the files in $Fn$ will also share files of the same type as those in $Fn$* holds true for these conditions.

Interestingly, we see that the same does not apply for the BitTorrent network. While there is some increase in accuracy compared to a random selection process, it is statistically insignificant, and indeed the accuracy of the classifier is less than 50%. This difference could plausibly be linked to the relative number of unique users and files in the BitTorrent and Gnutella networks. The Gnutella network is abundant in unique files but low on users, whereas the BitTorrent network is low on unique files and high on users. Further study across a range of file-sharing networks and sampling methods would be required to identify the necessary preconditions for accurate classification under this method.

In the results from Section 3.2, we discover that the behavioural prediction method has a very poor accuracy. We may draw from this that the previously stated assumption that *if peer $p_k$ is sharing files $Fp_k$ and peer $p_y$ is also sharing files $Fp_y$ such that $Fp_k \cap Fp_y \neq \emptyset$, then peer $p_k$ is likely to obtain and share other files peer $p_y$ is sharing* is an incorrect one, and that future peer ownership of files is not directly determined by the history of similar peers. This means that the behavioural prediction application we proposed is unlikely to provide useful predictions without substantial modification.

More accurate predictions might be gleaned from a more inclusive model similar to that proposed in [16]. Alternatively, alterations to the method of determining file equality may help improve the effectiveness of the collaborative filtering method, both for prediction and in general. Recall that we test for file equality using the reported hashes; if we additionally make use of low word distance between filenames, we may correctly identify more equalities between files which are currently considered distinct, increasing the number of predictions and matches between predictions and observations.

The results from Section 3.3 show promise in support of the original assumption, that *similarity of file-sharing habits reflects a similarity or connection between individuals.* While it would be hasty to make such a strong conclusion based on such a summary test of *closeness* as a distance metric, there is at least an indication of the metric's suitability to be used in identification and connection-related tasks, which can be expanded upon in further work specifically evaluating clustering techniques for the support of link analysis.

Related work on file-based similarity metrics has been restricted to files of a specific type — song files — to attempt to overcome sparsity problems [13]. This approach has several positive aspects which could be adopted in user clustering and identification for investigative purposes, including a graph-based approach for measuring distance between non-overlapping peers.

## V CONCLUSIONS & FURTHER WORK

We have outlined three potential applications of collaborative filtering technology in criminal investigations of file-sharing networks and evaluated each application. We make three key findings:

- First, we demonstrate that collaborative filtering technology can be successfully applied in identifying new media of certain categories. We show a significant effect in the responses of human reviewers when comparing filenames selected via a collaborative filtering method with randomly selected files. We also note that a significant effect is found within observations of the Gnutella network, but not within the BitTorrent network observations, indicating that certain characteristics of file-sharing networks could be influential in determining the effectiveness of collaborative filtering techniques. Nonetheless, the large number of file-sharing applications based on the Gnutella model [19] makes this finding significant.
- Second, we find that collaborative filtering models are insufficiently accurate in predicting the file-sharing behaviour of peers. We suggest that models which consider other factors than sharing history may prove more effective, taking into account variables such a time of day and duration of file availability which may be important factors in driving downloading behaviour.
- Third, we demonstrate that high *closeness* measures correlate with IP address, lending initial support to the use of *closeness* as a distance metric in clustering and identifying users. This could be highly useful in helping investigators uniquely identify peers engaging in illegal file-sharing activity and in investigating networks of criminal file-sharers.

Although our initial intent was to demonstrate applicability of this method across file-sharing networks, we have found that the nature of the underlying network has a large impact on the effectiveness of various collaborative filtering tasks. This may be linked to observations of the disparity between user and file counts summarised in Figure 1. It would appear that the Gnutella protocol is more amenable to this form of analysis than the BitTorrent network. An interesting branch of further study would be to pursue this line of enquiry, applying collaborative filtering for the purposes of new media classification to different file-sharing networks and attempting to confirm that effectiveness is related to the ratio of distinct users and hashes. This could help optimise performance of the media classification algorithm.

Other avenues for further work include the development of a more suitable predictive model for file-sharing behaviour, and the further investigation of the utility of *closeness* as a clustering or identification distance metric; the latter could be particularly useful in conjunction with techniques from the field of social network analysis. Investigation into the effects of utilising filename edit distance equality in collaborative filtering could result in improved accuracy in classification of files and prediction of downloading behaviour.

Finally, an important area to be explored before these methods can be applied in assisting investigation is the design of a suitable system for the deployment of collaborative filtering techniques in a manner which integrates with existing network monitoring technology, so that real-time or near real-time notifications of new media can alert either human investigators or other investigative software to study potentially infringing files or potentially malicious peers.

## ACKNOWLEDGEMENTS

This work was conducted as part of the EU Safer Internet Programme's Project iCOP: Identifying and Catching Originators in Peer-to-Peer Networks, 2011-2013.

### References

[1] K.-K. Choo. Organised crime groups in cyberspace: a typology. *Trends in Organized Crime*, 11:270–295, 2008. ISSN 1084-4791. URL `http://dx.doi.org/10.1007/s12117-008-9038-9`. 10.1007/s12117-008-9038-9.

[2] K. Chow, K. Cheng, L. Man, P. Lai, L. Hui, C. Chong, K. Pun, W. Tsang, H. Chan, and S. Yiu. BTM-an automated rule-based BT monitoring system for piracy detection. In *Internet Monitoring and Protection, 2007. ICIMP 2007.*

*Second International Conference on*, page 2. IEEE, 2007.

[3] F. L. Fessant, A. m. Kermarrec, and L. MassouliÃ©. Clustering in peer-to-peer file sharing workloads. In *In 3rd International Workshop on Peer-to-Peer Systems (IPTPS)*, 2004.

[4] D. Hughes, P. Rayson, J. Walkerdine, K. Lee, P. Greenwood, A. Rashid, C. May-Chahal, and M. Brennan. Supporting law enforcement in digital communities through natural language analysis. In S. Srihari and K. Franke, editors, *Computational Forensics*, volume 5158 of *Lecture Notes in Computer Science*, pages 122–134. Springer Berlin / Heidelberg, 2008.

[5] D. Hughes, J. Walkerdine, G. Coulson, and S. Gibson. Peer-to-peer: is deviant behavior the norm on p2p file-sharing networks? *Distributed Systems Online, IEEE*, 7(2), feb. 2006.

[6] iCOP. iCOP, July 27th, 2012. URL `http://scc-sentinel.lancs.ac.uk/icop/`.

[7] M. Liberatore, R. Erdely, T. Kerle, B. N. Levine, and C. Shields. Forensic investigation of peer-to-peer file sharing networks. *Digital Investigation*, 7, Supplement(0):S95 – S103, 2010. ISSN 1742-2876. URL `http://www.sciencedirect.com/science/article/pii/S1742287610000393`.

[8] M. Liberatore, B. N. Levine, and C. Shields. Strengthening forensic investigations of child pornography on p2p networks. In *Proceedings of the 6th International Conference*, Co-NEXT '10, page 12. ACM, New York, NY, USA, 2010. ISBN 978-1-4503-0448-1. URL `http://doi.acm.org/10.1145/1921168.1921193`.

[9] C. M.S. and Steel. Child pornography in peer-to-peer networks. *Child Abuse and Neglect*, 33(8):560 – 568, 2009. ISSN 0145-2134. URL `http://www.sciencedirect.com/science/article/pii/S0145213409001604`.

[10] M. Piatek, T. Kohno, and A. Krishnamurthy. Challenges and directions for monitoring p2p file sharing networks-or: why my printer received a DMCA takedown notice. In *Proceedings of the 3rd conference on Hot topics in security*, pages 12:1–12:7. USENIX Association, Berkeley, CA, USA, 2008.

[11] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM, 2001.

[12] Y. Shavitt, E. Weinsberg, and U. Weinsberg. Building recommendation systems using peer-to-peer shared content. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1457–1460. ACM, 2010.

[13] Y. Shavitt, E. Weinsberg, and U. Weinsberg. Estimating peer similarity using distance of shared files. In *Proceedings of the 9th international conference on Peer-to-peer systems*, page 4. USENIX Association, 2010.

[14] Y. Shavitt and U. Weinsberg. Song clustering using peer-to-peer co-occurrences. In *Multimedia, 2009. ISM'09. 11th IEEE International Symposium on*, pages 471–476. IEEE, 2009.

[15] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009:4:2, January 2009. ISSN 1687-7470. URL `http://dx.doi.org/10.1155/2009/421425`.

[16] R. Thommes and M. Coates. Epidemiological modelling of peer-to-peer viruses and pollution. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1 –12, april 2006. ISSN 0743-166X.

[17] J. Wang, J. Pouwelse, R. L. Lagendijk, and M. J. T. Reinders. Distributed collaborative filtering for peer-to-peer file sharing systems. In *Proceedings of the 2006 ACM symposium on Applied computing*, SAC '06, pages 1026–1030. ACM, New York, NY, USA, 2006.

[18] S. Weiss and N. Indurkhya. Lightweight collaborative filtering method for binary-encoded data. In L. De Raedt and A. Siebes, editors, *Principles of Data Mining and Knowledge Discovery*, volume 2168 of *Lecture Notes in Computer Science*, pages 484–491. Springer Berlin / Heidelberg, 2001.

[19] ZeroPaid. Gnutella, July 2012. URL `http://www.zeropaid.com/software/file-sharing/gnutella/`.