First Report on
Paedophile Keywords Observed in *eDonkey*

Clémence Magnien, Matthieu Latapy, Jean-Loup Guillaume, and Bénédicte Le Grand

LIP6 – CNRS and Université Pierre et Marie Curie

Firstname.Lastname@lip6.fr

Abstract

This report presents our first analysis results on paedophile keywords observed in exchanges between eDonkey clients and their server. We first describe our dataset and the messages studied in this context. General statistics on the number of queries, filenames, clients and keywords are provided, before focusing on paedophile keywords appearing in user queries and/or in filenames. Statistical and graph analysis methods have been used to characterize paedophile keywords in terms of frequency distributions and co-occurrences in queries and in filenames.

1 Introduction.

In most P2P systems, including eDonkey, users are searching files by making keyword-based queries. The system then searches for files whose name contain these keywords and sends the list of matching files to the user. The user can then choose to download the files using various information: filename, size, type of file, number of providers, etc. As a consequence, keywords play a key role in the daily use of such systems. They must reflect the content of files so that users can find them efficiently, and they also give rich insights on user interests and available contents. Our goal here is to use these keywords to gain information on paedophile activity in the eDonkey P2P system.

We first describe the dataset on which our study is based in Section 2. General statistics and a global view of the keywords encountered in queries and filenames is then given in Section 3, whereas Section 4 is dedicated to paedophile keywords. We provide a first analysis of the co-occurrences of specific paedophile keywords in queries and in filenames in Section 5, before giving perspectives on our future work in Section 6.

2 Dataset.

We use here the largest measurement currently available of an eDonkey system, which is described in detail in [2]. We give its main features here, with a focus on the information it contains regarding keywords. Notice that a public version of the dataset is available, where everything, including keywords, is fully anonymised. We use for this report a less anonymised version, internal to the project; however, user privacy is also protected in this version, as described below.

The dataset consists in all the messages managed (sent or received) by a large eDonkey server during almost 10 weeks. The number of messages exchanged during this period is almost 9 billion and these messages contain information on nearly 90 million users and more than 275 million distinct files. Notice that this dataset represent *typical* use of the system by users: it includes some paedophile activity but there is no specific focus on this type of activity.

Among the recorded messages, some contain textual data, which we call *strings*. In particular, we focus here on keyword-based queries and filenames. A query is a set a keywords aimed at describing the files sought by the user who sent it. Conversely, filenames are supposed to describe file contents, and therefore users offering files are supposed to choose filenames accordingly. This allows users to choose advisedly which files to download when they are presented a set of filenames in answer to a query.

Note that a given content may be described by different filenames and that a given filename can refer to different files. Therefore P2P system often use an unique identifier for files (file identifier or *fid* in the sequel) and keep an internal association between *fids* and filenames. In eDonkey, when a user sends a keyword-based query, the server looks for filenames matching the keywords and returns the corresponding *fids*, with other information such as filenames, file sizes, and others.

Keyword queries and filenames contain therefore very valuable information about the activity in the system, and the types of contents that are exchanged: for instance, a user entering "Madonna" in a search query expresses interest in finding content related to Madonna such as music or video. The presence of the "Madonna" keyword in a filename indicates that this file is probably related to Madonna (like a song or a concert video). The rest of the filename (including the extension of the file) gives generally additional details on the content.

However, keyword queries and filenames may also contain personal information about users. For instance, a personal video of a user that he/she distributes over the eDonkey system may contain his/her name. Similarly, somebody can enter his/her name as a query to see if the system contains files related to him/her. In general, people studying the cases in which personal information may be hidden in some seemingly anonymous data acknowledge the following: in a system with a large number of users, if the users have to enter textual information, then some users will tend to enter personal information about themselves or people close to them, such as names and phone numbers [7].

Following this assertion, due to their large size, sets of keyword queries and filenames in the eDonkey system most probably contain personal information. Since we do not want to retain such information, both for conforming to legal constraints and for ethical reasons, we used an anonymisation procedure to suppress such personal information. We will describe this procedure now.

Anonymisation procedure

We created two versions of the dataset: a public one, and one available only to members of the project. In the public dataset, all strings (including queries and filenames) are fully anonymised by replacing each word by an integer, without possibility to reverse the process. However, each word is always replaced by the same integer, which makes it possible to compare two queries or filenames by comparing the integers they contain. This is a very strong protection, because none of the words present in the original queries or filenames are retained in this version. This means however that it is impossible to study the types of contents provided or searched for in the system with this version of the dataset.

The internal, restricted version of the dataset is also anonymised, but in a less complete way. For this version, we chose to anonymize personal information, we therefore had to distinguish between personal/sensitive and general/non-sensitive information. We chose the following approach, inspired by [1]: non-sensitive data appear *frequently*, in different forms, in the data, while personal data are *rare*. The idea is therefore to retain frequent words, while anonymizing infrequent ones by replacing them by integers. For instance, the filename: `michael jackson vs lionel richie wanne be all night long white label remix mp3` does not represent personal information about Michael Jackson or Lionel Richie, and we do not want to anonymize it.

More precisely, we did the following: we isolated all strings present in the dataset (keyword queries and filenames), and kept only one copy of each (each string may appear several times in the system: for instance if a user enters the same query several times, or if two distinct files (*fids*) have the same name). We then broke down these unique strings into words (words are sequences of letters and/or numbers only, no space nor punctuation characters). We then normalized the obtained words by converting all letters to lowercase.

Table 1 illustrates this normalization: it presents two filenames, with *fids* 1 and 2. These files appear with different unique filenames in the system. The number of distinct filenames is 5. Among these names some are almost identical (e.g., *hello* and *Hello*), some are semantically close (e.g., *hello* and *hi*) whereas some are completely different (e.g., *hello* and *business*). The latter case is particularly interesting as the corresponding file (2) might be a fake[1]. The normalization then brings all words to lowercase, and the number of distinct filenames after normalization is 3: `hello`, `hi` and `business` (we however keep the information that these filenames were originally different).

After this normalization, each word then appears in a certain number of strings, and words appearing in a very small number of strings have a very high chance of representing personal information. We set a threshold of 100 to distinguish between rare and common words: all words appearing in less than 100 different strings are replaced by an integer, while others are kept in clear in the dataset. Note that during the normalization process we only considered distinct strings, therefore if a client enters 100 times the same query, the corresponding keywords will count only for 1.

[1]A fake is a file whose name does not correspond to its content. In particular, if a file has some completely different filenames, it may be considered as a fake.

fid	original filename	normalized filename
1	hello	hello
1	hi	hi
1	Hello	hello
2	hello	hello
2	business	business
2	Business	business

Table 1: Example of the normalization of filenames for two *fids*.

Notice that an infrequent word may reveal personal information in two different ways: it may be the name or telephone number of a user; it can also represent a real interest from a user towards a very specific type of content. However, in this case, if this word is rare, it means that this type of content is also very rare, and it means that it might be possible to trace the user through his/her rare interest.

The above example of a filename: `michael jackson vs lionel richie wanne be all night long white label remix mp3` is in fact obtained after this anonymisation procedure. All words in this filename appear frequently, and therefore appear clearly in the dataset. The filename: `-3056538 -112669 -3086639`, on the other hand, is fully anonymized. It contains three words and, since all three are infrequent, they are all replaced by an integer (we placed a dash '-' before such anonymized words to distinguish them from words that are integers, such as '2008' or '101'). Finally, the filename `broken flowers fr -296471 avi` contains both frequent words and one infrequent word. The frequent words are kept in clear, while the infrequent ones are replaced by an integer.

Finally, the following example of a filename shows clearly how personal information is anonymized, while valuable information about the content of the file is preserved: `by karl photos zoophilie serpent gratuit amateur sylvie -219121 toulouse tel -184378 jeune salope 20 -1630843 wmv`[2]. The original name of the file contained the last name and telephone number of a girl whose first name is Sylvie who lives in Toulouse. The information retained in the anonymized version of the filename, though it indicates its content very clearly, does not give personal information about this person anymore (in the sense that it is not possible to know *who* this person is).

3 Global view.

Before turning to the specific study of paedophile keywords, we study the general characteristics of our dataset. We first present separate statistics about keywords appearing in filenames and queries, then we compare the use of keywords in queries and in filenames.

[2]The english translation of the words reads: `by karl photos zoophilia snake free amateur sylvie -219121 toulouse tel -184378 young slut 20 -1630843 wmv`.

3.1 Filenames

There are 18 953 264 files (identified by their *fid*) that have names: 19 424 369 distinct filenames before normalization, corresponding to 16 334 911 filenames after normalization.

Several *fids* may share a same name, for instance a file named `madonna mp3` can refer to many music files from Madonna. Conversely each *fid* may have several names, for instance a specific song from Madonna can have different names: `madonna vogue mp3`, `madonna vogue high quality mp3` or `madonna vogue 128k mp3`. The number of distinct (*fid*, filename) pairs is 24 666 569.

3.2 Queries

Our dataset contains 127 320 728 keyword queries (including duplicate queries). This corresponds to 52 905 135 distinct queries, independently of the user who made them (this means that if different users formulate queries with the same keywords in the same order, these queries are considered as identical). The number of peers who sent at least one query is 28 395 512. Finally, this corresponds to 115 932 041 distinct (user, query) pairs.

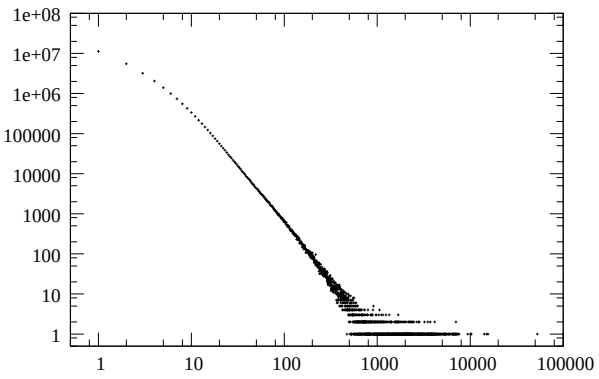

Figure 1: Distribution of the number of distinct queries per user.

We present in Figure 1 the distribution of the number of distinct queries per user. This plot reads as follows: each point has a value on the x axis which corresponds to a number of distinct queries, and the value on the y axis corresponds to the number of users who made exactly this number of queries (we consider the number of *distinct* queries, therefore if a user made the same query one thousand times, it will count as only one query). Note that this Figure is in double logarithmic scale. This distribution is highly heterogeneous: most users made only a few queries (more than 10 million users entered a single query during the 10 weeks measurement), while a small number of them entered a large number of different queries (more than 10 000 in some rare cases).

This heterogeneity indicates a high diversity of user behaviors which is not specific to this measurement: the majority of users sends a small number of distinct queries into the system, whereas a few users behave very differently and send a very large number of distinct queries. All intermediate behaviors between these two extremes can be observed.

3.3 Keywords

We now turn to the study of keywords appearing in queries and filenames. We observe 6 663 013 different keywords in total. Among these words, 1 222 937 are not anonymized (*i.e.*, appear in more than 100 different strings, see Section 2).

The number of distinct keywords appearing in filenames is 2 797 058, among which 1 222 654 are non anonymized. Concerning queries, 4 822 288 distinct words are observed, 119 793 of which are not anonymized. This indicates a difference between filenames and queries: words used in queries are in general much rarer than words appearing in filenames, meaning that users do not follow the same rules when they enter queries than when they name files: files must be named wisely so that an user can find them, in particular it should contain general words and more specific ones. On the contrary queries must be as specific as possible so that the user can find the file he/she is looking for.

filenames

rank	keyword	nb occurrences
1	mp3	12 121 052
2	avi	2 860 225
3	the	2 657 349
4	rar	1 610 669
5	de	1 607 634
6	jpg	1 296 610
7	la	1 236 001
8	of	1 082 521
9	a	1 039 469
10	mpg	993 077

queries

rank	keyword	nb occurrences
1	the	4 147 197
2	de	3 382 473
3	la	2 337 404
4	a	1 761 179
5	of	1 751 848
6	2	1 398 154
7	i	1 153 601
8	ita	1 101 964
9	2006	1 075 982
10	el	1 025 315

Table 2: Top 10 words by frequency. Left: in filenames. Right: in queries.

Table 2 presents the 10 most frequent keywords in filenames and queries. The most frequent keywords found in queries are mostly articles (e.g. *the*); this is much less the case in filenames where half of the 10 most frequent words are file extension. This is not surprising: most filenames have an extension, such as *mp3* or *avi*, indicating the type of the file which is very useful for users looking for a specific type of content.

The ten most frequent words do not however give valuable information about the contents provided or searched for in the system. We therefore present in Table 3 the most frequent *meaningful* words appearing in filenames and queries. We observe here a high similarity between filenames and queries: both lists consist of almost exactly the same words. Notice that words like *xxx* and *sex*, though they are present in both lists, appear with rather low ranks (between 67 and 199) both in filenames and in queries, which may be counter-intuitive.

Words belonging to keyword queries have been typed in by users. It is therefore interesting to study how many words are entered by a given user. Figure 2 (left) presents

filenames

rank	keyword	nb occurrences
21	you	549 050
33	love	406 261
35	dvdrip	402 954
39	live	385 676
42	remix	375 013
43	dj	373 216
48	feat	344 111
77	xxx	241 176
108	sex	152 605

queries

rank	keyword	nb occurrences
15	you	860 508
21	love	693 408
37	dj	491 165
46	live	447 906
49	pc	396 270
55	black	344 210
67	sex	294 282
199	xxx	128 404

Table 3: Top *meaningful* words. Left: in filenames. Right: in queries.

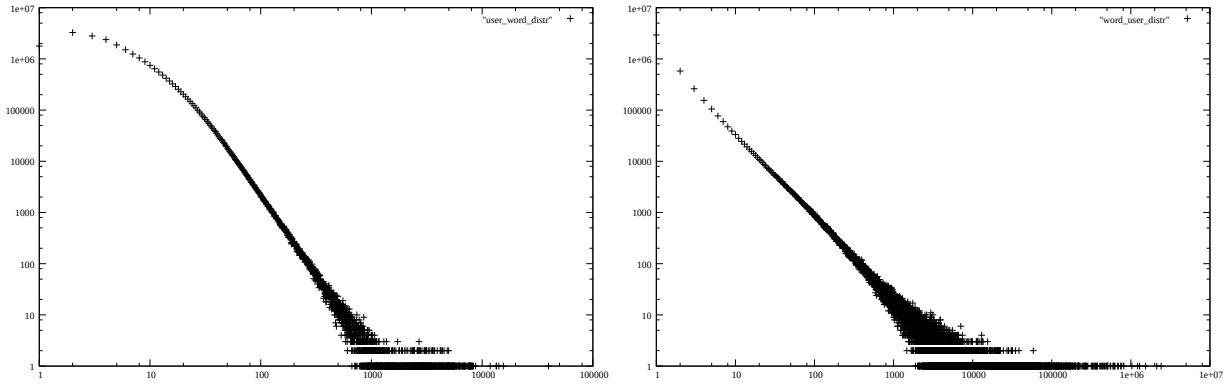

Figure 2: Left: Distribution of the number of words per user. Right: Distribution of the number of users per word.

the distribution of the number of distinct words per user. Again, we observe very different behaviors among users: while the vast majority of users uses a small number of different words in queries, a small number of users use a very large number of words during their use of the system (up to 50 000 in one extreme case[3]). This corroborates what was observed with the number of distinct queries made by users, see Figure 1. Notice however that, though most users use a small number of keywords, more users use 2, 3 or 4 words than just a single word. This is quite intuitive when we think about the way we perform queries ourselves – using more than one keyword usually provides more relevant and accurate results.

Conversely, Figure 2 (right) presents the distribution of the number of users using a given word in their queries. Again, this distribution is highly heterogeneous, most words being used by only a small number of users, while some popular words are used by up to more than 20 million users.

[3]This corresponds to one distinct word entered every 2 minutes during 10 weeks, which probably indicates a non-human user.

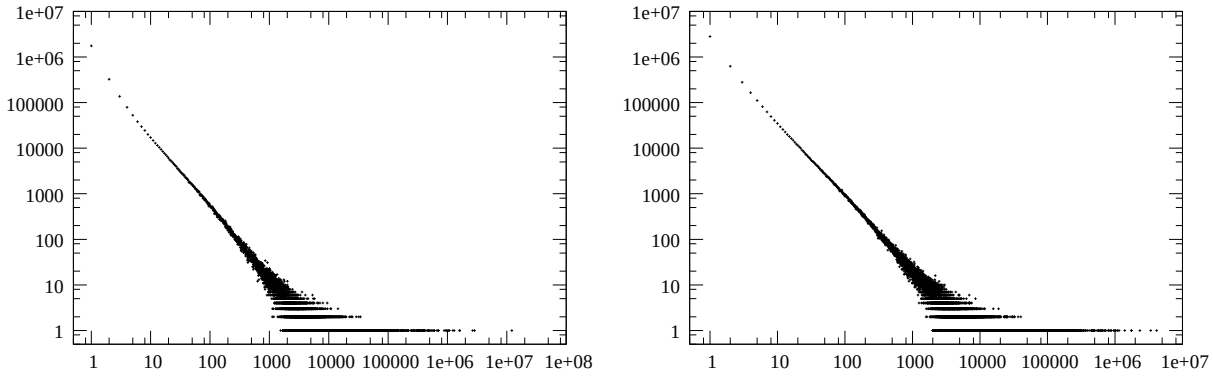

Figure 3: Distribution of word frequencies. Left: in filenames. Right: in queries.

Figure 3 presents the distribution of the frequency of words in filenames and queries, *i.e.*, the number of filenames (resp. queries) to which a given word belongs. The rightmost dots for each plot correspond to the words in Table 2. Both distributions are similar, and both heterogeneous. This means that most words appear in a small number of filenames (resp. queries): 2 484 092 (resp. 4 264 048) words appear in at most 10 filenames (resp. queries). Conversely, a small number of words appear in a very large number of filenames (resp. queries). Tables 2 and 3 show that, though there are some similarities, the words with very high frequency are not the same in filenames and queries.

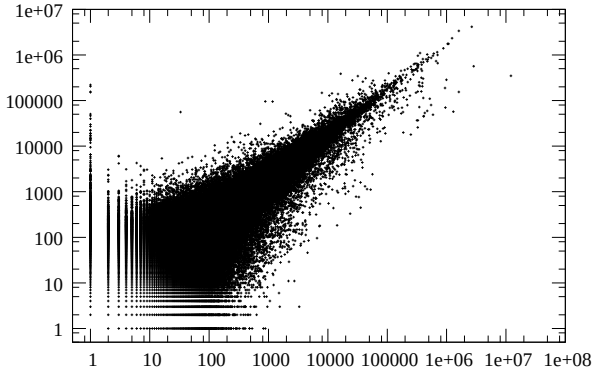

Figure 4: Correlations between number of occurrences in filenames (horizontal axis) and queries (vertical axis): for each word we print a point at coordinates (x, y) if it appears in x filenames and y queries.

Figure 4 confirms this. In this plot, each point corresponds to a word that can be found both in filenames and in queries. The x-coordinate of a point is its number of occurrences in filenames, and its y-coordinate is its number of occurrences in queries. Therefore a point that is high on the y-axis and low on the x-axis (*i.e.*, in the top left of the plot) represents a word that appears in many queries but few filenames, and conversely, a point in the bottom right of the plot represents a word appearing in many filenames but few queries.

We can see that many points are close to the diagonal: they represent words that

have the same popularity in filenames and queries (words close to the bottom left have a low popularity, while words in the top right are very popular). However, a significant number of words appear with a high frequency in one case and a low frequency in the other, confirming that filenames and queries are composed following different rules [4].

keyword	nb occurrences	keyword	nb occurrences
girl	220 498	cowboy	15 647
girls	205 927	kind	14 710
boy	154 398	bomb	14 441
boys	153 150	filles	14 235
child	50 414	kinder	13 390
children	46 484	girlfriend	11 839
playboy	39 996	bomba	9 443
pedo	29 908	pedofilia	6 558
attack	27 056	cowboys	6 519
fille	22 436	kiddy	6 519
enfants	22 222	boyfriend	5 841
boyz	19 127	fatboy	5 466
enfant	18 943	incesto	4 776
incest	17 203	underage	4 381
preteen	16 769	ladyboy	4 014

Table 4: Top 30 words appearing in queries but not in filenames.

This difference is even more striking when considering words which appear only in filenames or only in queries. Though the study of words which appear in filenames but not in queries does not reveal anything specially interesting, the study of the most frequent words used in queries but never in filenames (presented in Table 4) shows a striking observation: most of these words have a paedophile connotation. This shows a huge difference between the type of contents that users search (queries), and the type of content available (filenames).

In this case, this shows that there is a very high demand for paedophile content, but that few such content is available [5]. We study in more details paedophile keywords in the next section.

4 Paedophile keywords.

We now turn to the study of paedophile keywords, which gives good information about uses of the eDonkey system to exchange paedophile contents: peadophile keywords appearing

[4] Figure 4 uses log-log scale, therefore a small shift from the diagonal may result in a very large difference.

[5] We do not have yet a conclusive explanation for this phenomena. One possibility is that the administrator of the server configured it to remove files containing these keywords in their names.

in keyword queries indicate an interest from the user for this type of content; conversely, such keywords appearing in filenames probably indicate paedophile content.

This gives valuable information about the uses of the system, though in practice things are not that simple: paedophiles tend to avoid detection by using secret keywords, or some files with a paedophile names may not have paedophile content, while some files with innocent-sounding names may be paedophile.

	occurrences in			
keyword	filenames	queries	quer./f.names	nb users
lolita	15 890	27 053	1.7	20 807
ptsc	1 622	5 129	3.16	3 816
hussyfan	1 317	6 883	5.23	5 345
r.ygold	580	9 996	17.23	7 602
babyj	413	1 761	4.26	1 462
babyshivid	187	1 709	9.14	1 405
kidzilla	52	840	16.15	754
pthc	32	55 844	1 745.13	29 589
*n*yo	9 143	50 326	5.5	10 452
*n*yr	1 976	9 270	4.69	1 741
madonna	37 954	67 283	1.77	45 030
sex	355 114	294 282	0.83	214 961
xxx	234 225	128 404	0.55	84 380
porn	201 492	61 335	0.3	46 740
rape	16 423	27 644	1.68	19 186
torture	8 806	9 551	1.08	7 486

Table 5: Number of occurrences of classical paedophile keywords in filenames and queries. For comparison, we also provide the number of occurrences of a more general keyword (madonna), sex related keywords (porn, sex) and violent keywords (rape, torture). The ratio of the number of queries vs. the number of filenames, as well as the number of users having typed the keyword, are also given in the table.

Table 5 presents the number of occurrences of some *classical* paedophile keywords in filenames and in queries. These keywords are widely used for indicating paedophile content, and can easily be found by looking in the data. The paedophile nature of these keywords is confirmed with the help of *Urban Dictionary*[6], a slang dictionary. The keywords *yr* or *yo* mean *years* or *years old*, and they are widely used to indicate the age of a protagonist in pornographic and/or paedophile content. We also present for comparison the number of occurrences of a more general keyword (madonna), sex related keywords (porn, sex) as well as violent keywords (rape, torture).

Notice that the fraction of all filenames containing a clear paedophile keyword is over one for one thousand. The fraction for queries is similar, and one may notice the importance of

[6] `http://www.urbandictionary.com/`

age indications in this context (we will enter in more details regarding this in Section 4.1). Notice however that paedophile keywords are less common than other harmful keywords like *torture* or *rape*.

Another interesting observation, confirming what we observed at the end of the previous section, is that there are much more queries containing paedophile keywords than filenames actually containing these keywords (column 4 of Table 5 gives the ratio): all paedophile words have a ratio queries/filenames over 3 (except the word *lolita* which is often used in pornographic content to design young, but older than 18, girls). A possible explanation is that there is more demand for this type of content than supply for them, or that paedophile filenames use less common keywords, to avoid detection for instance. On the contrary, the words *sex*, *xxx* and *porn* appear in more filenames than queries.

The number of users who typed these keywords in their queries is also indicated (column 5). Not surprisingly, this number seems to be more or less proportional to the number of queries containing those words.

filenames

rank	keyword	nb occurrences
3	avi	1 207
4	mpg	824
5	rar	809
6	new	642
7	jpg	575
8	lolita	560
9	model	479
12	lolitaguy	372
13	mylola	304
14	info	288

queries

rank	keyword	nb occurrences
6	new	2 534
7	pedo	2 226
9	mpg	1 885
10	girl	1 839
12	boy	1 412
13	cum	1 202
14	webcam	1 041
15	vicky	980
16	lolita	974
17	mom	945

Table 6: Top ten non-trivial keywords in paedophile filenames and queries.

Table 6 presents the most frequent meaningful words in paedophile filenames and queries. We defined a string (whether it is a filename or a query) as paedophile if it contained one of the paedophile keywords of Table 5 (except for the word *lolita*, for the reason explained above). We can see that these words are different from the most frequent words among all filenames and queries, see Tables 2 and 3 for comparison. This means that these filenames and queries belong to a more specific context. We indeed note that these words tend to belong to a pornographic context. We can notice that keywords used in queries are more explicit than the ones used in filenames.

4.1 Age indication

Some strings may contain an age, mainly in the form of a number followed by *yo*[7]. In a filename, such ages indicate the age of the person represented in the corresponding (pornographic or paedophile) picture or video. However, this may not be a valid information in all cases, since some file providers may place false ages in filenames to make these files more attractive. More interestingly, an age in a query represents an interest from a user towards this type of content. We have seen in Table 5 that a significant number of filenames and queries contain an age indication. Figure 5 presents the repartition of these ages, both in filenames and queries. We can see that the vast majority consists of ages below 18 (92% (resp. 98%) of filename (resp. queries) indications concern ages strictly below 18), and that there are a very large number of young, and even very young, ages: about half the queries and 40 percent of the filenames refer to ages of 10 years old or less, and approximately 15% of queries and 7% of filenames refer to ages of 5 years old or less.

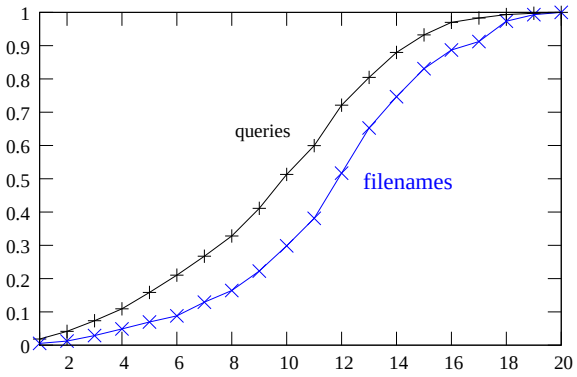

Figure 5: Repartition of ages claimed in filenames and asked for in queries. For each n from 1 to 20, we selected all filenames and queries containing the string $n$yo (for n years old), and we plotted for each x the fraction of these strings with $n \leq x$.

One striking observation is that queries focus on *younger* ages than filenames: for all ages up to 11, the proportion of queries for this age is larger than the proportion of filenames containing this age. Above 11 the tendency is inverted. This is to consider together with the fact that there seems to be more demand than supply for paedophile content: this is even more pronounced for paedophile content with very young children.

4.2 Unknown keywords

Finally, a very interesting question is the detection of unknown paedophile keywords. Indeed, users interested in paedophile content tend to avoid detection by law-enforcement authorities by using hidden keywords, known by a small number of persons. Detecting such keywords is therefore of prime interest, both for an in-depth study of paedophile activity, and for law-enforcement authorities.

[7]Less frequently, the age is indicated by *yr* or simply *y*.

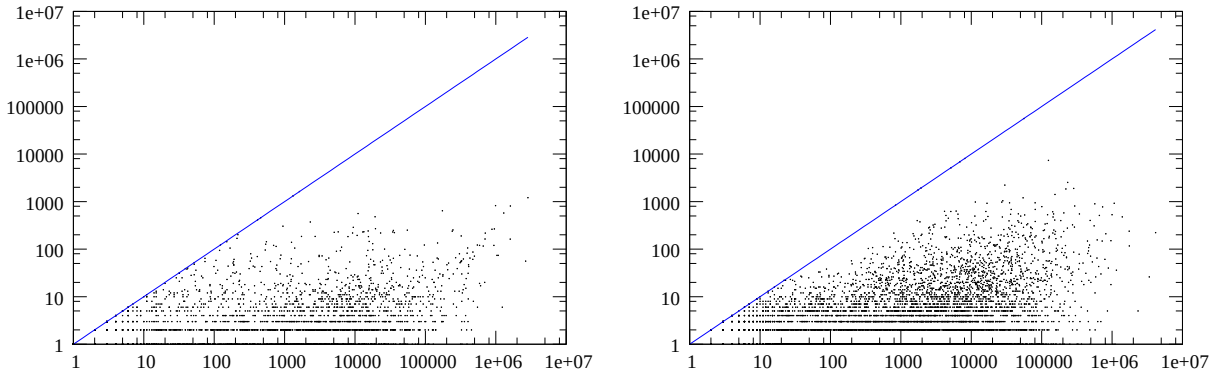

Figure 6: Correlations between number of occurrences in paedophile filenames (resp. queries) and all filenames (resp. queries).

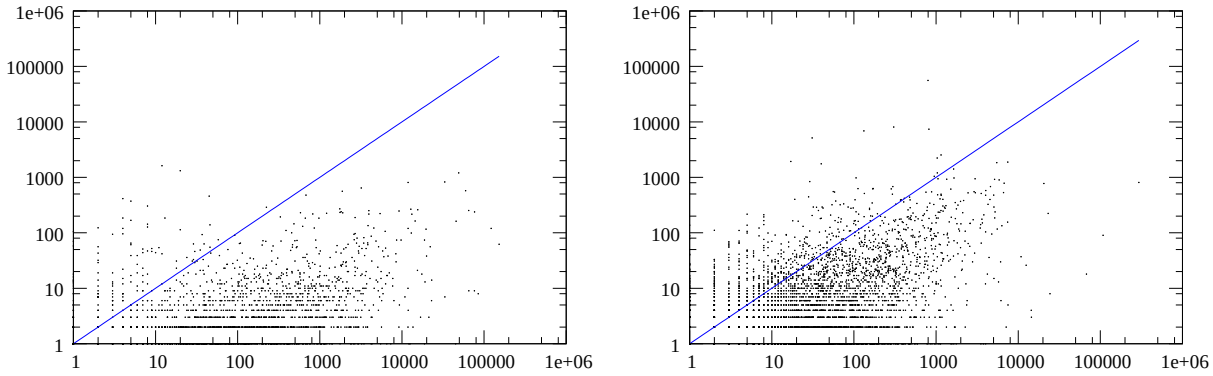

Figure 7: Correlations between number of occurrences in paedophile filenames (resp. queries) and the ones containing sex.

We have already seen at the end of Section 3 that comparing the frequency of occurrences of words in different contexts can give valuable information: studying the words appearing in queries but not in filenames yielded a list of paedophile keywords. Figure 6 uses the same idea: it presents the correlations between the number of occurrences of words in paedophile filenames (resp. queries) to their number of occurrences in all filenames (resp. queries). In this plot, points to the bottom right represent words with a high frequency in general, but a low frequency in paedophile filenames (or queries). These words most probably do not have a paedophile focus. Words close to the diagonal, however, have almost the same frequency in general than in paedophile filenames or queries: this means that these words appear only (or almost only) in paedophile context. These words therefore most probably have a strong peadophile focus[8].

Though this approach seems promising, it did not yield very interesting results. Indeed, there is too much difference between the paedophile context and the general context

[8]The words we used for determining whether a string is paedophile naturally occur *only* in paedophile strings, and are therefore exactly on the diagonal.

composed of *all* filenames or queries. For instance, pornographic words with no paedophile focus naturally tend to have a higher frequency in paedophile filenames or queries than in general.

To counter this, we study in Figure 7 the correlations between the number of occurrences of words in paedophile filenames (resp. queries) to their number of occurrences in filenames (resp. queries) containing the word *sex*. In this plot, words on the diagonal are words occurring equally frequently in paedophile strings and in strings containing *sex*: these words are generic pornographic words. Words on the bottom right appear more frequently in strings containing *sex* than in peadophile strings, and do not have a peadophile focus. Finally, words in the top left appear *more* frequently in paedophile strings, and therefore are words with a strong paedophile focus.

There are several ways to isolate words with a strong paedophile focus using this idea. A first one is to choose words that are the furthest away from the diagonal. Another one consists in choosing words that have a highest ratio of appearances in paedophile strings vs strings containg sex. We present in Tables 7 and 8 the list of words chosen according to these two techniques in both filenames and queries.

This approach is very promising. First, we can see that it succeeds in isolating words with a strong paedophile focus, which cannot be done by simply looking at the list of words appearing in paedophile strings, see Table 6. More interestingly, we can see that this method succeeds in isolating peadophile keywords that are more or less hidden. For instance, the word *qqaazz* is a paedophile keyword, known by law-enforcement authorities, that is not known by a large audience. It appears as the second keyword with the highest ratio in queries, see Table 8 (right).

Many words on the obtained lists are unknown to us, and we suspect that some of them are hidden paedophile keywords. We will discuss this with law-enforcement authorities. Finally, we obtain four keyword lists (two slightly different methods, each applied to filenames and queries). Though there are strong similarities between these lists, we can also observe noticeable differences. We will in the future investigate this to understand more precisely the advantages and drawbacks of each method, and try to refine them.

5 Co-occurrence graphs.

In previous sections, we have presented statistics describing general and paedophile keywords. We have seen that relations between keywords (co-occurrence in the same filename or query, in particular) may be used to derive meaningful information. This may be pushed much further using graph analysis, as we will do in the rest of the project. In this section, we illustrate this approach with a first very basic step which already demonstrates its strength: we observe relationships among some paedophile keywords by drawing their co-occurence graph.

This graph is built as follows. We first defined a set P of well known paedophile keywords: $P = \{$*babyj, hussyfan, kidzilla, pthc, ptsc, raygold, ygold*$\}$. We then selected the set S of all filenames in our dataset that contain (at least) one word in P. All the

filenames, differences					filenames, ratio				
rank	word	occ: sex	paedo,	diff.	rank	word	occ: sex	paedo,	ratio
1	ptsc	12	1624	1612	1	ptsc	12	1624	135.33
2	hussyfan	20	1319	1299	2	nn	1	105	105.00
3	raygold	45	457	412	3	babyj	4	414	103.50
4	babyj	4	414	410	4	lolitaguy	5	372	74.40
5	lolitaguy	5	372	367	5	amateurz	1	70	70.00
6	mylola	8	304	296	6	hussyfan	20	1319	65.95
7	tanta	4	185	181	7	ygold	2	123	61.50
8	voglia	13	185	172	8	kacy	1	58	58.00
9	eurololita	5	168	163	9	tanta	4	185	46.25
10	349	8	143	135	10	mylola	8	304	38.00
11	ygold	2	123	121	11	eurololita	5	168	33.60
12	9yo	8	127	119	12	lolalover	3	94	31.33
13	10yo	4	120	116	13	10yo	4	120	30.00
14	nn	1	105	104	14	8yo	2	56	28.00
15	11yo	20	114	94	15	arina	4	96	24.00
16	12yo	46	139	93	16	newstar	2	43	21.50
17	arina	4	96	92	17	playtoy	1	18	18.00
18	lolalover	3	94	91	18	349	8	143	17.88
19	amateurz	1	70	69	19	imouto	1	16	16.00
20	info	224	288	64	20	4yo	2	32	16.00
21	12y	5	65	60	21	9yo	8	127	15.88
22	kacy	1	58	57	22	lourinha	2	29	14.50
23	cs	37	93	56	23	voglia	13	185	14.23
24	10y	5	61	56	24	stasia	1	14	14.00
25	8yo	2	56	54	25	photobook	1	14	14.00
26	gostosinha	7	51	44	26	galia	1	13	13.00
27	kidzilla	9	52	43	27	-313544	1	13	13.00
28	company	11	54	43	28	12y	5	65	13.00
29	newstar	2	43	41	29	10y	5	61	12.20
30	5yo	8	47	39	30	shiori	1	12	12.00

Table 7: Top 30 words appearing more frequently in paedophile filenames than in filenames containing 'sex'. Left: sorted by difference between number of occurrences. Right: sorted by ratio between the number occurrences.

words appearing in these filenames are the nodes of the co-occurrence graph. Two of these nodes are linked together if they appear in a same filename in S. Notice that most of these keywords are not related to paedophile content, but some certainly are.

The obtained graph has 1807 nodes (the 7 original paedophile keywords and the other words appearing with them in filenames) and 2686 links. Figure 8 shows a drawing of this graph, in which the paedophile keywords in P are drawn in red, and the green nodes

queries, differences

rank	word	occ: sex	paedo,	diff.
1	pthc	798	55844	55046
2	ygold	305	8088	7783
3	hussyfan	132	6883	6751
4	r	817	7360	6543
5	ptsc	31	5129	5098
6	raygold	17	1929	1912
7	babyj	40	1761	1721
8	new	1151	2534	1383
9	pedo	1034	2226	1192
10	vicky	106	980	874
11	kidzilla	29	840	811
12	9yo	112	625	513
13	open	56	554	498
14	moscow	58	544	486
15	12yo	171	631	460
16	10yo	170	619	449
17	lsm	28	445	417
18	sandra	297	683	386
19	babyshivid	19	400	381
20	11yo	134	512	378
21	dad	285	621	336
22	childlover	60	374	314
23	linda	103	395	292
24	7yo	54	321	267
25	tori	32	291	259
26	8yo	75	333	258
27	petersburg	35	289	254
28	ls	68	321	253
29	kingpass	41	291	250
30	5yo	42	280	238

queries, ratio

rank	word	occ: sex	paedo,	diff.
1	ptsc	31	5129	165.45
2	qqaazz	1	137	137.00
3	raygold	17	1929	113.47
4	kinderficker	1	88	88.00
5	lso	1	78	78.00
6	pthc	798	55844	69.98
7	nablot	2	111	55.50
8	hussyfan	132	6883	52.14
9	cbaby	1	50	50.00
10	babyj	40	1761	44.02
11	kdquality	5	217	43.40
12	rika	1	41	41.00
13	izzy	1	40	40.00
14	kidzilla	29	840	28.97
15	chiharu	1	28	28.00
16	tvg	4	110	27.50
17	kimmy	5	137	27.40
18	tuesday	1	27	27.00
19	shiori	1	27	27.00
20	ygold	305	8088	26.52
21	liluplanet	8	212	26.50
22	-149121	1	26	26.00
23	mylola	8	194	24.25
24	lada	3	69	23.00
25	marga	1	22	22.00
26	kaj	1	22	22.00
27	arina	3	65	21.67
28	rca	5	106	21.20
29	babyshivid	19	400	21.05
30	cjb	1	21	21.00

Table 8: Top 30 words appearing more frequently in paedophile queries than in queries containing 'sex'. Left: sorted by difference between number of occurrences. Right: sorted by ratio between the number occurrences.

indicate words with an age description (of the form $n$yo, $n$yr or ny).

It appears clearly in this drawing that many words appear together with only one word in P. Instead, some appear with two words in P or more, and some words even appear together with many words in P. Many words of the form $n$yo are in this case. This shows that indicating age in paedophile filenames is not specific to another keyword in P.

The other words which co-appear with all words in P are also of interest: this is the

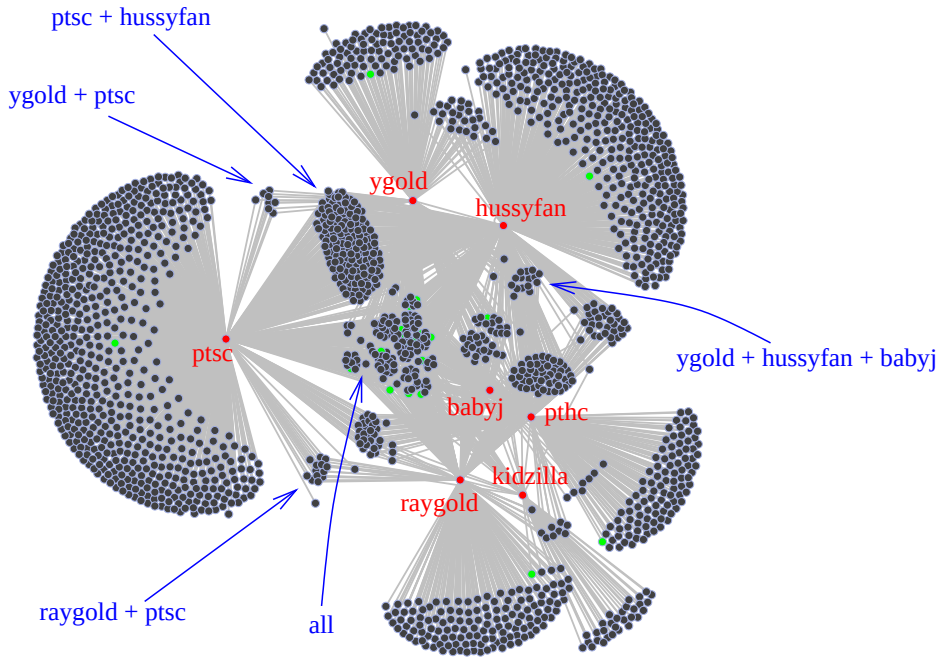

Figure 8: Representation of the occurrence of words in conjonction with paedophile words. Initial paedophile keywords are in red, age indication keyworks are in green.

case for instance of *webcam, vicky, lolita, mylola, young, sweet, kid, lolitaguy,* etc.

Such graphs may be constructed using the set of *all* filenames, and then community detection techniques [4, 6] may be used to identify interesting clusters, as well as relations between clusters. Going further, many graph analysis methods may be used to analyze co-occurrence and other relations between keywords. We will use such approaches in the rest of the project to identify clusters of paedophile keywords, and among them maybe more specific clusters.

6 Conclusion and future work.

In this report, we presented a first set of analysis of the keywords captured in the data we collected on an eDonkey server. We described the main features of these keywords from a statistical point of view, and derived results on peadophile activity from them. We also designed simple methods to identify keywords susceptible to refer to paedophile content, which is useful for instance in content rating [5], for measurement directed towards peadophile content and more generally for studying and monitoring paedophile activity on the internet and outside.

Many other directions remain to explore. In particular, richer information focused on

paedophile activity may be obtained with other kinds of measurements, based on honeypots and/or clients sending queries to the system [3]. We are currently conducting such measurements, and will present results soon.

The available data itself may be used to derive richer results. For instance, one may observe the queries entered by users who send paedophile queries: are these paedophile queries too? if not, which other kinds of content do paedophile search? is there an evolution of these queries during time? are there some queries which may indicate that the user will probably be interested in peadophile content later? etc. All these questions are extremely important for our understanding of paedophile activity, and we are currently addressing them.

Another key issue for law enforcement institution is to identify users who introduce new paedophile content in the system, or play a key role for their dissemination (by converting them into other file formats, for instance, or by changing their names, thus creating fakes). Our data contain much information on this: one may for instance observe which users provide a given content first (as seen in our limited measurement); one may study how files spread among users, in particular paedophile ones; etc.

Regarding paedophile keywords, in addition to the identification of unknown such keywords and the study of their use (in particular using community detection, see Section 5), one may study their time evolution. In particular, the emergence of *new* paedophile keywords is a poorly understood phenomenon with important implications. The dataset we collected allows the investigation of this question, and the identification of newly appearing keywords. This is an important application which we will develop in the near future.

Acknowledgements. ...

References

[1] Eytan Adar. User 4xxxxx9: Anonymizing query logs. In *Query Logs Workshop, WWW'07*, 2007.

[2] Frederic Aidouni, Matthieu Latapy, and Clemence Magnien. Ten weeks in the life of an edonkey server. Submitted, 2008.

[3] Oussama Allali, Matthieu Latapy, and Clémence Magnien. Measurement of edonkey activity with honeypots. Submitted, 2008.

[4] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. accepted in JSTAT, 2008.

[5] Matthieu Latapy, Clémence Magnien, and Guillaume Valadon. First report on database specification and access including content rating and fake detection system. `http://antipaedo.lip6.fr/`.

[6] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications (JGAA)*, 10(2):191–218, 2006.

[7] Bruce Schneier. Why 'anonymous' data sometimes isn't. `http://www.wired.com/politics/security/commentary/securitymatters/2007/12/securitymatters_1213`.