# Detection of Child Sexual Abuse Media on P2P Networks: Normalization and Classification of Associated Filenames

**Alexander Panchenko, Richard Beaufort, Cédrick Fairon**
Centre for Natural Language Processing (CENTAL) – Université catholique de Louvain
Place Blaise Pascal 1 – 1348 Louvain-la-Neuve – Belgium
E-mail: {alexander.panchenko, richard.beaufort, cedrick.fairon}@uclouvain.be

### Abstract

The goal of the iCOP project is to build a system detecting the originators of pedophile content on P2P networks such as BitTorrent, eDonkey, or Kad. This paper outlines the key functions of the language processing in the iCOP system. Next, we describe the architecture of the language analysis module and its key components – filename classifier, term extractor, and filename normalizer. The language resources used in each component are discussed. The paper is also presenting the first experiments with the module on the standard porn data (used in the preliminary tests as a substitute of child pornography data). Our results show that the module is able to separate titles of the pornographic galleries and videos from the titles of encyclopaedia articles with accuracy up to 97%. Finally, we discuss the directions for the future research and developments of the iCOP language analysis module.

**Keywords:** text classification, text normalization, P2P networks, CSA, porn filters

## 1. Introduction

The goal of the iCOP project[1] is to develop a novel forensics software toolkit to help law enforcement agencies across the EU identify new or previously unknown child abuse media and its originators on peer-to-peer (P2P) networks. Until now, the only way to identify such media is through manual analysis by law enforcement personnel. However, such a manual approach is difficult or impossible given the large number of files that need to be reviewed individually. The limited resources that law enforcement agencies possess make it impractical for them to examine the thousands of new files that may appear on P2P networks every day.

The key output of the project – the iCOP software toolkit – will be used by law enforcement to help detect, filter, and prioritize new instances of child abuse media on P2P networks. iCOP system is operates alongside existing P2P monitoring tools like PeerPrecision. Using trace data from these monitors as input, iCOP identifies candidate suspect media that contain Child Sexual Abuse (CSA) based on a combination of advanced pattern recognition techniques. The language analysis tool employs potential CSA media in P2P networks based on a combination of several sources of evidence: the modelling of offender file sharing behaviour; the analysis of file sharing patterns and query patterns; language processing of file names and queries. Candidate media discovered by language analysis are fed to a content-based media analysis tool, which prioritizes and filters material further.

The core components of the iCOP system are targeted at an automatic identification of new CSA content and its distributors, using evidence in the form of textual queries and filenames in P2P networks, of user's file sharing behaviour, and of the image and video content itself. It is of particular importance to search for “new” material as these previously unknown files are usually introduced on the network by people who one believed to be close to the victim or the abuse. These files have also a greater chance to be related to on-going situation of abuse.

Our contribution to the iCOP is focused on the language technology. In particular, we are working on machine learning techniques that recognize the pedophile queries or filenames in the data of a P2P system. The development of these components follows an iterative optimization process in which the underlying statistical models and data representations are modified and system performance is validated on a test dataset. In iCOP, such a data-driven improvement is not straightforward, as number of target texts (filenames and queries) is very limited and may even be illegal to access in some cases. Furthermore, crawling additional pedophile texts directly from the network is illegal. Therefore, the construction of appropriate training (language) resources require special attention. Evaluation and optimization strategies need to be adapted to work around the problem of accessing the data.

In the following section, we outline the architecture of the language processing module of the iCOP system. We also discuss the assessment strategies with respect to the file classification, term extraction, and filename normalization components of the module. Next, we present our first experiments with the module. Results of the classification of the “standard” porn texts are described. We conclude with directions for the future research and developments.

[1] http://scc-sentinel.lancs.ac.uk/icop/

## 2. Architecture of the System

Figure 1 presents the architecture of the language analysis module and outlines how it will be integrated with the iCOP toolkit[2]. The core goal of the this module is to complement the media and behaviour analysis tools of iCOP with an analysis of filenames and other text content associated to a candidate child sexual abuse media (i.e. metadata enclosed with videos and pictures). Other modules of iCOP will identify potential CSA images and videos. These files are used as input for the language analysis module. First, the module checks if the media is a known CSA file. If not, features are extracted from the input file name and its text description. They are used in a statistical model which decides if the file contains CSA materials. Secondly, the module extracts key terms describing the CSA media. These terms are used to analyse and track the evolution of the offender vocabulary over time.

The iCOP toolkit is going to use the language analysis module to analyse both filenames and queries. Our module processes these two kinds of texts exactly in the same way. In the following, we will refer to language analysis as filename processing. In the next sections, we will outline methods used to build the file classification, the term extraction and filename normalization components of the language analysis module.

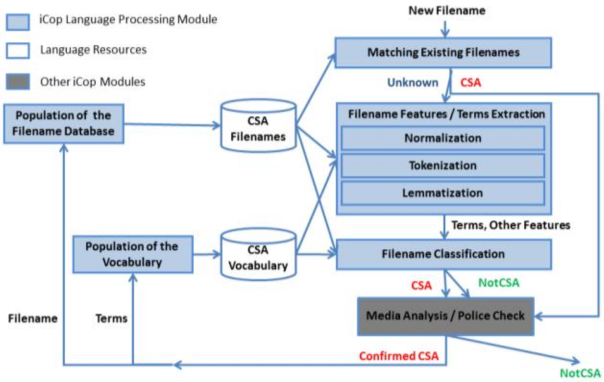

Figure 1: Architecture of the language analysis module of the iCOP forensics toolkit.

### 2.1 File Classification by Textual Description

The goal of the file classification component is to decide if an input file contains some CSA material using solely the textual descriptions of the file. This kind of classification is challenging for several reasons. First, filenames may be meaningless, such as “0012664321.mpg”. Second, file text descriptions and metadata may be short or absent. Finally, text descriptions often use highly non-standard language (abridgments, abbreviations, spelling errors, technical terms, etc). It is also challenging to separate CSA from regular porn as the two categories use a lot of common terms.

In order to deal with these and other issues we have developed a text classification system. The purpose of this system – given a file name and its metadata as input – is to check if the file is an already known CSA media stored in a database of filenames. If the file is unknown, features are extracted from the text describing the file. To do so, the text is segmented. Next, text normalization (Beufort et al., 2010) is used to represent content of a file from scarce, noisy, and misspelled text descriptions. Then a statistical machine learning classifier (a Support Vector Machine or a Regularized Logistic Regression) is used to separate regular files from those containing some abuse content. We rely on the LIBSVM (Chang and Lin, 2011) and LIBLINEAR (Fan et al., 2008) classification software for two reasons. First, because of their commercial-friendly open source licenses. Second, because the libraries efficiently implement the state-of-the-art classification algorithms.

The classifier is trained on a database of file names containing CSA material. The file classification module is used to recognize newly published files in a P2P network which contain abusing content. The list of these candidate files is transferred to the Media Analysis component, where the content of each file is further analyzed (Ulges and Stahl, 2011).

The file classifier is trained and tested on three sets of filenames: regular filenames, regular pornographic filenames, and filenames containing CSA material. We constructed the two first datasets ourselves from the files openly and legally available on the Internet. The third dataset should be provided by our collaborators from law enforcement. Each dataset is a list of entries composed of tuples <Class, Filename>. Here, the Filename is the original name of the file (with the tags if any); Class is either “positive” (porn) or “negative” (non-porn). Following the standard machine learning methodology, these datasets (divided into disjoint training and validation sets) are used to assess performance of the classifiers. Cross-validation is used to estimate performance of the filename classification. In particular, the following metrics are calculated: accuracy, balanced classification rate, mean squared error (MSE), root-mean-square error (RMSE). We present the first results of filename classification in Section 3.

### 2.2 Term Extraction

For each detected child abuse media, our module returns a set of normalized terms describing it. These terms let monitors seamlessly analyze and track topics on P2P networks. Basically, the extracted terms, are normalized lexical units obtained at the feature extraction stage. Words which were never used before for describing CSA content are also recognized. Detection of the new terms aims to help identify personal names, and locations related to recent victims and abuse cases. In order to detect new terms, we rely on a database of CSA terms. A term is considered new if and only if it appears in a CSA filename (alongside known CSA terms) but is not present in the CSA vocabulary.

The database of CSA language is constructed during the training phase. It is based on specialized hand-crafted terminologies provided by law enforcement agencies. This initial lexicon is enriched with terms extracted from

[2] http://cental.fltr.ucl.ac.be/projects/icop/

the filenames containing child abuse media. To do so, we run the segmentation and normalization tools used on the feature extraction stage on these files and add the most frequent terms to the CSA vocabulary. The CSA database is updated once new files containing illegal material are provided.

The evaluation of the term extraction relies on the users' feedback. Once the module is integrated into the iCOP toolkit, the users can interact with it. Thus, this evaluation will be conducted at the final stages of the project. Generally, satisfaction of users will be an evaluation measure for this module.

### 2.3 Filename Normalization

The text normalization component module transforms input filenames into normalized versions. Text normalization plays two important roles in the system. As we have already said, filenames and their text descriptions contain highly non-standard language patterns, such as abridgments, abbreviations, spelling errors, and so on. These language phenomena hamper standard text classifiers, which stumble against big number of Out-Of-Vocabulary words. Therefore, the first goal of the text normalization is to improve the feature extraction procedure. For instance, we would prefer to treat all variations of the word “porn” including its abbreviations and forms with spelling errors as a single feature. Secondly, the text normalization improves readability of the output by the term extraction module.

The text normalization is performed by an algorithm which learns rewriting rules from parallel aligned corpora. This normalization is loosely inspired by previous works on SMS normalization (Beaufort et al., 2010). This method shares similarities with both spell checking and machine translation approaches. In our system, all lexicons, language models and sets of rules are compiled into finite-state machines and combined with the input filename by composition, a special operation defined on (weighted) transducers and on (weighted) automata. We use our own finite-state tools: a finite-state machine library and its associated compiler (Beaufort, 2008). In conformance with the format of the library, the compiler builds finite-state machines from weighted rewrite rules, weighted regular expressions and n-gram models. We first tested this approach on the normalization of text messages. The algorithm and its models are described in (Beaufort et al., 2010). In order to learn a normalization model we needed a sequence alignment at the character-level. The best sequence alignment was obtained by applying the algorithm described in (Cougnon and Beaufort, 2009). This algorithm gradually learns the best way of aligning strings.

The filename normalization component is evaluated similarly to the filename classification component. The evaluation is performed on a corpus of aligned filenames by 10-fold cross-validation. The system is trained 10 times, each time leaving out one of the subsets from the training corpus, but using only the omitted subset as test corpus. The normalization system is evaluated in terms of BLEU score, Word Error Rate (WER), and Sentence Error Rate (SER).

## 3. The First Results

The first experiments with the language analysis module described above were conducted on a dataset of regular pornographic files for three reasons. First, the CSA data is a special case of pornographic content. Therefore, the “standard porn” is useful to develop the system, as well as for initial selection of parameter configurations. In these initial experiments the names of regular pornographic files serve as a substitute for child pornography filenames. Pornography can be expected to share important characteristics with CSA material (like general sex-related vocabulary, types of file extensions, text indicating technical parameters of the media file, etc.). Second, at the time of writing this article, the CSA data were not yet provided by our law enforcement partners due to various administrative and legal issues.

In the experiment described in this article, the system was trained to separate a title of a porn gallery from a title of an encyclopedia. Our training dataset consisted of regular pornographic data crawled from the four specialized porn sites: PicHunter[3], PornoHub[4], RedTube[5], and Xvideos[6]. In particular, each positive training example was composed of a title of a porn video/gallery and tags associated with it. We collected 51350 pornographic titles. We used as the negative training examples 55000 randomly selected titles of the English Wikipedia, each composed of at least 4 words. The full dataset was composed of 106.350 titles. We kept 10% of the data (10635 texts) for the validation. We selected from rest 90% of the data 93.629 titles which are represented with at least two features after all the preprocessing steps. These 93.629 titles were used to train a binary classifier. We did not perform any feature selection – all 39.127 lemmas extracted from the training fold were used as features.

It is important to mention that since the text normalization component was not yet fully integrated in the language analysis module, we used a simplified text normalization procedure in the experiment described here. First, the titles were cleaned up from the numbers and the special symbols. Second, they were POS tagged and lemmatized with TreeTagger (Schmid, 1994). All the standard stopwords (except the “sex-related” ones such as *him*, *her*, *woman*, *man*, etc.) were removed from the titles. Examples of two negative and two positive training examples are provided below:

```
<text class='negative'>
<original>Contractors and General Workers
Trade Union</original>
<lemmas>contractors#NNS#contractor
and#CC#and general#JJ#general
workers#NNS#worker trade#NN#trade
```

[3] http://www.pichunter.com/
[4] http://www.porno-hub.com/
[5] http://www.redtube.com/
[6] http://www.xvideos.com/

```
union#NN#union</lemmas>
</text>

<text class='negative'>
<original>1957-58 American Hockey League
season</original>
<lemmas>1957-58#CD#1957-58
american#JJ#American hockey#NN#hockey
league#NN#league season#NN#season</lemmas>
</text>

<text class='positive'>
<original>Husband catches his wife fucking
with his brother .</original>
<lemmas>husband#NN#husband
catches#VVZ#catch his#PP$#his wife#NN#wife
fucking#VVG#fuck with#IN#with his#PP$#his
brother#NN#brother .#SENT#.</lemmas>
</text>

<text class='positive'>
<original>Slim Can Bearly Take The
Dick .</original>
<lemmas>slim#JJ#slim can#MD#can
bearly#RB#bearly take#VV#take the#DT#the
dick#NN#dick .#SENT#.</lemmas>
</text>
```

Results of our preliminary experiments are presented in Table 1. Our results show that a Support Vector Machine (SVM) or a Regularized Logistic Regression (LR) can distinguish a Wikipedia title from a pornographic video title with accuracy of 96-97%. In particular, the best results were obtained with C-SVM with liner kernel (96.97%). We tested also other kernels of the C-SVM and nu-SVM, but the linear kernel appeared to provide the best results. The training of a model with the linear kernel is also much faster. These results suggest that the titles of encyclopedia are linearly separable from the pornographic titles in the vector space of 39.127 lemmas (therefore the complex kernels are not required). Figure 2 depicts results of the metaparameter optimization of the C-SVM with linear kernel with the grid search. As we can see, this procedure improved the accuracy by 0.37%.

Our results confirm the correctness of the chosen methodology for the filename classification. However, separating CSA from regular porn is much more challenging due to overlapping vocabularies, and because of the lower number of available CSA filenames. Therefore, we expect that the accuracy of the final classifier will be lower. The next stage of our project is training the system on the real CSA data provided by our police collaborators.

## 4. Conclusion

In this paper we described the principal functions of the language processing module of the iCOP forensics toolkit. Next we presented the architecture of the module and outlined its main components. We listed the language resources used in the filename classifier, the term extractor, and the filename normalizer. Next, this work also presented the first filename classification results with the developed language processing module. Our experiments have shown that the state-of-the-art linear models such as Support Vector Machine or Regularized Logistic Regression are able to distinguish titles of the porn galleries from the titles of encyclopedia articles with an accuracy of 97%. Finally, we discussed the directions for the future research and developments of the module.

| Classifier | Training | Accuracy |
|---|---|---|
| C-SVM, linear kernel | 8m 59s | **96.97%** |
| C-SVM, polynomial kernel | 15m 11s | 51.71% |
| C-SVM, RBF kernel | 22m 20s | 51.71% |
| C-SVM, sigmoid kernel | 14m 58s | 51.71% |
| nu-SVM, linear kernel | 12m 48s | 88.20% |
| nu-SVM, poly. kernel | 4m 39s | 79.77% |
| nu-SVM, RBF kernel | 26m 49s | 88.35% |
| nu-SVM, sigmoid kernel | 14m 5s | 87.45% |
| L2-reg.L2-loss SVM (dual) | 0.364s | 96.45% |
| L2-reg.L2-loss SVM (primal) | 0.459s | 96.52% |
| L2-reg.L1-loss SVM (dual) | 0.366s | 96.47% |
| L1-reg. L2-loss SVM | 0.162s | 96.47% |
| L2-reg.L2-loss LR (primal) | 0.548s | 96.24% |
| L1-reg. LR | 0.388s | 93.95% |
| L2-reg. LR | 1.176s | 96.27% |

Table 1: Results of the title classification with different learning algorithms (default metaparameters were used).

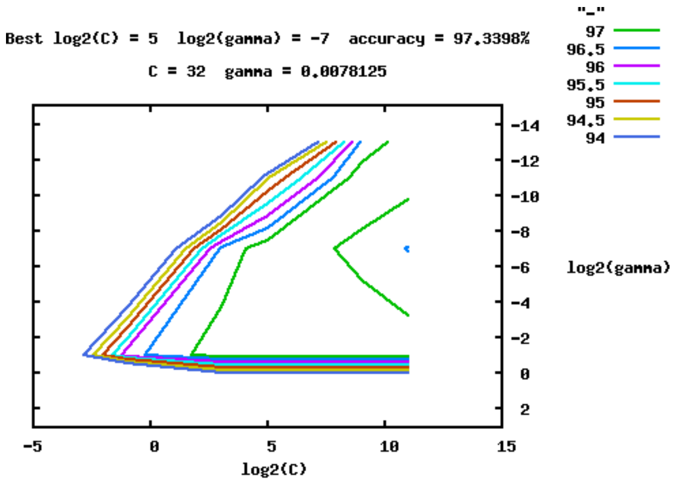

Figure 2: Optimization of the metaparameters of the title classifier (C-SVM, linear kernel) with the grid search.

## 5. Acknowledgements

This work has been funded by the EC Safer Internet project iCOP. Stéphanie Weiser from Université catholique de Louvain provided comments and remarks, which considerably improved quality of the paper.

## 6. References

Beaufort R. (2008). Application des Machines à Etats Finis en Synthèse de la Parole. Sélection d'unités non uniformes et Correction orthographique. *Ph.D. Thesis, Faculty of Computer Science, FUNDP, Belgium*.

Beaufort, R. Roekhaut, S. Cougnon, L.-A. Fairon C. (2010). A hybrid rule/model-based finite-state framework for normalizing SMS messages. In *Proceedings of ACL 2010*, pp. 770-779.

Chang C.-C. and Lin C.-J. (2011) LIBSVM: a library for support vector machines. In *ACM Transactions on Intelligent Systems and Technology*, 2:27:1--27:27.

Cougnon, L.-A. Beaufort, R. (2010). SSLD: a French SMS to Standard Language Dictionary. In *Proceedings of eLEX 2009*, pp. 33-42.

Fan, R.-E. Chang, K.-W. Hsieh, C.-J. Wang, X.-R. and Lin C.-J. LIBLINEAR: A library for large linear classification. In *Journal of Machine Learning Research 9*, 1871-1874.

Schmid H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. *In Proceedings of International Conference on New Methods in Language Processing*. 44–49.

Ugles A., and Stahl, A. (2011). Automatic Detection of Child Pornography using Color Visual Words. In *Proceedings of International Conference on Multimedia and Expo.*