A learning-based approach for the identification of sexual predators in chat logs

Notebook for PAN at CLEF 2012

Javier Parapar[1], David E. Losada[2], and Alvaro Barreiro[1]

[1] Information Retrieval Lab, Dept. of Computer Science, University of A Coruña
[2] Centro de Investigación en Tecnoloxías da Información (CITIUS), Universidade de Santiago de Compostela
javierparapar@udc.es,david.losada@usc.es,barreiro@udc.es

Abstract The existence of sexual predators that enter into chat rooms or forums and try to convince children to provide some sexual favour is a socially worrying issue. Manually monitoring these interactions is a way to attack this problem. However, this manual approach simply cannot keep pace because of the high number of conversations and the huge number of chatrooms or forums where these conversations daily take place. We need tools that automatically process massive amounts of conversations and alert about possible offenses. The sexual predator identification challenge within PAN 2012 is a valuable way to promote research in this area. Our team faced this task as a Machine Learning problem and we designed several innovative sets of features that guide the construction of classifiers for identifying sexual predation. Our methods are driven by psycholinguistic, chat-based, and tf/idf features and yield to very effective classifiers.

1 Introduction

Many underaged children who are regular Internet users are the targets of unwanted sexual solicitation. This is a socially worrying issue of paramount importance. Health care professionals, educators, parents, police organizations, and the society as a whole should be prepared to respond to this increasingly prevalent problem.

In 2012, a sexual predator identification task was proposed within the Author Identification Task of the Unconvering Plagiarism, Authorship, and Social Software Misuse Lab (PAN 2012). This innovative and important challenge was divided into two parts: sexual identification subtask and line identification subtask. The second subtask was more exploratory and it did not have labeled data to train the systems. Therefore, we devoted our main efforts to the first subtask, which consists of identifying sexual predators from a large set of chat logs.

We tested different Machine Learning strategies and thoroughly tuned the classifiers to identify sexual predation. Our main contribution is the proposal of an innovative set of features to drive the classification of chat participants (two class problem: predator/non-predator). We utilized standard term-based features (tf/idf) but also other content-based features based on psycholinguistics. In the literature of psycholinguistics, there is strong evidence [10] that links the use of natural language to personality, social and situational fluctuations, and psychological interventions. Of particular interest are findings that point to the psychological value of studying word usage for identifying deception [6,2,8]. Since sexual predation in the Internet is an inherently deceptive activity, we felt that incorporating features based on psycholinguistics could help us to search for predatory behaviour within the chats.

We also defined other global features based on the activity of the users in the chatrooms, the type of conversations in which they tend to engage in, and other contextual factors. This is a ground-breaking set of features that substantially helps to find sexual predators.

2 Learning strategy

We approached the PAN 2012 sexual predation task as a supervised learning problem. Given the training collection, we focused on selecting the most effective classification strategy and the most effective sets of features to guide the classification. The training collection contains conversations from 97689 different subjects.

2.1 Representation of the subjects

Every participant often takes part in several chat conversations and interacts with different subjects in different ways. It is therefore quite challenging to understand how to properly represent the chatroom users from their interactions. Furthermore, the process of sexual predation is known to happen in phases [5]: gaining access, deceptive trust development, grooming, isolation, and approach. Therefore, every conversation could be classified in accordance to this categorization and, additionally, every user-to-user interaction could be monitorized to estimate what stages of predation have actually occurred. This leads to very intriguing issues related to how to extract relevant patterns of Internet sexual predation from massive amounts of chat conversations.

We are aware that these user's representation challenges are important to advance in sexual predation identification and we plan to face them in the near future. Anyway, we opted for approaching this year's task in a much simpler way. For every individual, we concatenated together all the lines written by him/her in any conversation in which he/she participated. The resulting text was our document-based representation for this chat participant (i.e. one document per subject). This means that we lose track of individual conversations and we simply record on a file all lines written by this chatter. This textual representation is recognizably simplistic but we expect that it still contains the basic clues to identify predation.

These document-based representations were used as an input to extract the content-based features (tf/idf and LWIC) described below. However, observe that we also include in our experiments a set of chat-based features that are not based on the text written by the chat participant but are based on the global behaviour of the subject in the chatrooms. This acts as a complementary representation for the chatters.

2.2 Features

We studied different strategies to extract a feature-based representation for the chat participants:

- *tf/idf* features. This is a baseline representation consisting of a standard unigram representation of the texts. Given the characteristics of the chat conversations, we decided to not apply stemming. We simply pruned the vocabulary by removing those terms appearing in 10 or less documents (i.e. terms used by 10 or less subjects were removed). This pruning eliminates those words that are used by a very limited number of people and it has the advantage of reducing significantly the number of features. This has important implications in the training time taken to build the classifiers. Terms whose character size was greater than 20 were also removed. Each term was weighted with a standard tf/idf weighting scheme [4]:

  $$tf/idf_{t,d} = (1 + \log(tf_{t,d})) \times \log(\frac{N}{df_t}) \tag{1}$$

  where $tf_{t,d}$ in the term frequency of the term t in the document d, N is the number of documents in the collection and df_t is the number of documents in the collection that contain t.
  We also considered bigrams and trigrams in our study. We excluded bigrams and trigrams occurring in three or less documents. This substantially limits the number of these n-grams features and maintains only those expressions whose pattern of usage is not marginal. The n-grams having a character size equal to or greater than 40 were also removed. We tested all the combinations of the tf/idf features, namely: unigrams only, bigrams only, trigrams only, unigrams+bigrams, unigrams+trigrams, bigrams+trigrams, and all n-grams. Anyway, for the sake of clarity, we will only report and discuss those combinations with reasonably good performance.
- *LWIC* features. We felt that predation could be discovered using psycholinguistic features. In the area of psychology [10], it has been shown that the words people use in their daily lives can reveal important aspects of their social and psychological worlds. Since we wanted to explore psychological aspects of natural language use, we decided to use Linguistic Inquiry and Word Count (LIWC) [9], which is a text analysis software program that calculates the degree to which people use different categories of words. The ways that individuals talk and write provide windows into their emotional and cognitive worlds and can be used to analyze aspects such as deception, honesty, etc. LWIC processes textual inputs and produces output variables such as standard linguistic dimensions, word categories tapping psychological constructs (e.g. affect, cognition), personal concern categories (e.g. work, home, leisure), and some other dimensions (paralinguistic dimensions, punctuation categories, and general descriptor categories). Overall, there are 80 different LWIC dimensions and we processed every document in our collection (as originally written, with no modifications or preprocessing) to obtain 80 LWIC features associated to every individual. The complete set of LWIC features is shown in Table 1. The first two features, wc and wps, are the total count of the number of words and the average number of words per sentence, respectively. The rest of the features are

percentages of occurrence of words from different linguistic categories (e.g. % of words in the text that are pronouns). The table includes the LWIC category, the abbreviation and some examples for each LWIC dimension. We used the complete LWIC 2007 English Dictionary with no modification. Therefore, the list of words in the table is just an illustrative example of the words associated to every category.

Category	**Abbrev**	**Examples**
Linguistic Processes		
Word count	wc	
words/sentence	wps	
Dictionary words	dic	
Words>6 letters	sixltr	
Function words	funct	
Pronouns	pronoun	I, them, itself
Personal pronouns	ppron	I, them, her
1st pers singular	i	I, me, mine
1st pers plural	we	We, us, our
2nd person	you	You, your, thou
3rd pers singular	shehe	She, her, him
3rd pers plural	they	They, their
Impersonal pronouns	ipron	It, it's, those
Articles	article	A, an, the
Common verb	verb	Walk, went, see
Auxiliary verbs	auxverb	Am, will, have
Past tense	past	Went, ran, had
Present tense	present	Is, does, hear
Future tense	future	Will, gonna
Adverbs	adverb	Very, really, quickly
Prepositions	prep	To, with, above
Conjunctions	conj	And, but,whereas
Negations	negate	No, not, never
Quantifiers	quant	Few, many, much
Numbers	number	Second, thousand
Swear words	swear	Damn, piss, fuck
Psychological Processes		
Social processes	social	Mate, talk,they, child
Family	family	Daughter,husband, aunt
Friends	friend	Buddy, friend, neighbor
Humans	human	Adult, baby, boy
Affective processes	affect	Happy, cried, abandon
Positive emotion	posemo	Love, nice, sweet
Negative emotion	negemo	Hurt, ugly,nasty
Anxiety	anx	Worried,fearful, nervous
Anger	anger	Hate, kill,annoyed
Sadness	sad	Crying, grief, sad
Cognitive processes	cogmech	cause, know, ought
Insight	insight	think, know,consider
Causation	cause	because, effect, hence
Discrepancy	discrep	should, would, could
Tentative	tentat	maybe, perhaps, guess
Certainty	certain	always, never
Inhibition	inhib	block,constrain, stop
Inclusive	incl	And, with, include
Exclusive	excl	But, without, exclude
Perceptual processes	percept	Observing, heard, feeling
See	see	View, saw, seen
Hear	hear	Listen, hearing
Feel	feel	Feels, touch
Biological processes	bio	Eat, blood, pain
Body	body	Cheek, hands, spit
Health	health	Clinic, flu, pill
Sexual	sexual	Horny, love, incest
Ingestion	ingest	Dish, eat, pizza
Relativity	relativ	Area, bend, exit, stop
Motion	motion	Arrive, car, go

Continued on next page

Category	Abbrev	Examples
Space	space	Down, in, thin
Time	time	End, until, season
Personal Concerns		
Work	work	Job, majors, xerox
Achievement	achieve	Earn, hero, win
Leisure	leisure	Cook, chat, movie
Home	home	Apartment, kitchen, family
Money	money	Audit, cash, owe
Religion	relig	Altar, church, mosque
Death	death	Bury, coffin, kill
Spoken categories		
Assent	assent	Agree, OK, yes
Nonfluencies	nonflu	Er, hm, umm
Fillers	filler	Blah, Imean, youknow

Table 1: LWIC dimensions

- *chat-based* features. Finally, we defined 11 additional features that capture some global aspects related to the activity of the individuals in the chatrooms. This included features such as the number of subjects contacted by a given individual, the percentage of conversations initiated by a given individual, the percentage of lines written by a given individual (computed across all the conversations in which he/she participated), the average time of day when he/she used to chat, the average number of users participating in the conversations in which he/she participated (e.g. does he/she always participate in 1-to-1 conversations?), etc. Somehow, we expected that this innovative set of features would be indicative of how active, anxious and intense each user is, and indicative of the type of conversations in which he/she usually engages (1-to-1 conversations, night/evening conversations, etc). We felt that these features could reveal some trends related to predation. The chat-based features are reported in Table 2.

2.3 Training

The PAN 2012 training collection has a large number of examples (97689 chatters) and our approach handles a large number of features for each example (e.g. there are more than 10k unigram features). Given these statistics, we decided to use LibLinear [1] for learning the classifiers. LibLinear is a highly effective library for large-scale linear classification. This library handles Support Vector Machines (SVMs) classification and Logistic Regression classification with different regularization and loss functions. We extensively tested against the training collection all the classifiers supported. We finally chose SVMs as our classifier for all our submitted runs and, therefore, in this article we will only report and discuss results for this learning model. More specifically, we utilized the L2-regularized L2-loss SVM primal solver[3].

This is a highly unbalanced two-class classification problem: 142 out of the 97689 subjects are labeled as predators in the training collection. When dealing with unbalanced problems, discriminative algorithms such as SVMs, which maximize classification accuracy, result in trivial classifiers that completely ignore the minority class [7].

[3] This is option `-s 2` when running the liblinear training script (`train`).

Feature Name	**Feature Description**
avgLineLengthChars	Average size (in characters) of the user's message lines in the collection.
avgTimeOfDayOfMessages	Average time of day when every message line was sent by the user. Time of day is measured in minutes from/to midnight (the smallest amount applies).
noOfMessageLines	Number of message lines written by the user in the collection
noOfCharacters	Character count of all the message lines written by the user in the collection
noOfDifferentUsers-Approached	Number of different users approached by the user in the collection
percentOfConversations-Started	Percentage of the conversations started by the user in the collection
avgNoOfUsersInvolved-InParticipedConversations	Average number of users participating in the conversations with the user
percentOfCharacters-InConversations	Percentage of the characters written by the user (computed across the conversations in which he/she participates)
percentOfLines-InConversations	Percentage of lines written by the user (computed across the conversations in which he/she participates)
avgTimeBetween-MessageLines	Average time, in minutes, between two consecutive message lines of the user
avgConversation-TimeLength	Average conversation length, in minutes, for the user (computed across the conversations in which he/she participates)

Table 2: Chat-level features associated to a given chat participant

Some of the typical methods to deal with this problem include oversampling the minority class (by repeating minority examples), undersampling the majority class (by removing some examples from the majority class), or adjusting the misclassification costs. Oversampling the minority class results in considerable computational costs during training because it significantly increases the size of the training collection. Undersampling the majority class is not an option for our sexual predation problem because we have a tiny number of positive examples (142) and we would need to remove most of the negative examples in order to have a sets of positive examples and negative examples that are comparable in size. This massive removal of negative examples would miss much information. We therefore opted for adjusting the misclassification costs to penalize the error of classifying a positive example as negative (i.e. a sexual predator classified as a non-predator).

In our experiments we applied 4-fold cross-validation and focused on optimizing F1 computed with respect to the positive class (being a predator):

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \tag{2}$$

where $P = TP/(TP + FP)$ and $R = TP/(TP + FN)$.

In our initial tests, we observed that performance was relatively insensitive to the SVM cost parameter (C) but very sensitive to the weights that adjust the relative cost of misclassifying positive and negative examples. We therefore focused on fine tuning this weighting. By default, LibLinear assigns a weight equal to 1 to every class label (i.e. $w_1 = 1, w_{-1} = 1$). These weights are multiplied by C and the resulting values are used by the SVM's optimization process to penalize wrongly classified examples. Since we need to penalize the misclassification of positive examples, we opted for fixing w_{-1} to its default value and iteratively optimizing w_1. The SVM cost parameter (C) was fixed to its default value ($C = 1$).

Given the feature sets described in subsection 2.2, we did not apply any feature selection strategy but simply configured a complete set of experiments combining the three sets of features. Essentially, we tested all the 1-set, 2-set and 3-set combinations of the feature sets.

Although our model selection criterion was based on F1, we also report precision and recall in all the tables. For each feature set, the results reported correspond with the highest F1 run (average 4-fold cross-validation F1) obtained after tuning the w_1 weight. Anyway, for the sake of clarity, we do not include the optimal w_1 in every table. We will analyze the optimal w_1 values after selecting our top runs.

Table 3 depicts the performance results obtained with a single set of features. The results clearly show that the content-based features perform poorly (tf/idf and LWIC both yield to F1 performance lower than 10%). The performance of the chat-based features is substantially higher but it is still rather modest (e.g. precision below 50% and $F1 = 56.73\%$). The tf/idf results were obtained with unigrams alone, tf/idf(1g). Anyway, we also tested the incorporation of bigrams and/or trigrams into the tf/idf features but they did not give much added value. The main conclusion that we extracted from these initial experiments is that taking features from a single set (tf/idf/LWIC/chat-based) is not enough to have reasonably good effectiveness.

Feature Set	**P**	**R**	**F1**
tf/idf(1g)	2.85	51.35	5.39
LWIC	4.79	70.95	8.97
chat-based	49.25	66.89	56.73

Table 3: Performance results (in percentage) with a single set of features

Next, we tested the combination of different sets of features, including different types of n-grams for the tf/idf features. This involved extensive exterimentation and validation against the training collection. Anyway, we only report in Table 4 the most representative runs. We finally decided to select five of them (those whose label is in italics) as our contribution to PAN 2012. We selected the runs not only based on the average 4-fold F1 performance but also based on how sensitive they were with respect to the w_1 setting.

Feature Set	**P**	**R**	**F1**
tf/idf(1g)+chat-based	89.15	80.99	84.87
tf/idf(1g,2g)+chat-based	91.74	78.17	84.41
tf/idf(1g,3g)+chat-based	89.68	79.58	84.33
tf/idf(1g,2g,3g)+chat-based	92.44	77.46	84.29
tf/idf(1g)+LWIC	78.99	76.76	77.86
chat-based+LWIC	45.58	66.22	53.99
tf/idf(1g)+chat-based+LWIC	87.69	80.28	83.82
tf/idf(1g,3g)+chat-based+LWIC	78.36	73.94	76.09

Table 4: Performance results (in percentage) with feature sets combining tf/idf, LWIC and chat-based. The runs that we submitted to PAN 2012 have their label in italics.

Another technique that we took into account is scaling. Scaling before applying SVM is known to be very important [3]. The main advantage of scaling is to avoid features in greater numeric range dominating those in smaller numeric ranges. Scaling also avoids numerical difficulties during the calculation. We therefore planned a thorough set of experiments with scaled features (in the interval [0,1]), either using `svm_scale` from LibLinear or applying other normalization methods (e.g. cosine normalization for the tf/idf features). In Table 5 we present the scaled version of the five runs that we selected for PAN 2012. The results with scaling were rather unsatisfactory. We never obtained any substantial gain from scaling and the performance was usually lower than the performance obtained with no scaling. Still, we decided to submit ten runs: the five selected runs in italics in Table 4 and their corresponding scaled versions (Table 5).

Overall, we observed that the unigram tf/idf features combined with the chat-based features were the most effective and robust features. Therefore, we nominated the run tf/idf(1g)+chat-based as our official run[4].

[4] The task organizers asked the participants to nominate only one run as the official run.

Feature Set	P	R	F1
tf/idf(1g)+chat-based (scaled)	80.29	77.46	78.85
tf/idf(1g,3g)+chat-based (scaled)	75.19	70.42	72.73
tf/idf(1g)+LWIC (scaled)	69.44	70.42	69.93
tf/idf(1g)+chat-based+LWIC (scaled)	84.09	78.17	81.02
tf/idf(1g,3g)+chat-based+LWIC (scaled)	71.52	76.06	73.72

Table 5: Performance results (in percentage) of the five selected feature sets when the features were scaled in [0,1].

2.4 The w_1 weight

Recall that w_1 controls the penalty given to positive examples that are misclassified. Our strategy to set w_1 was as follows. As argued above, w_{-1} was fixed to 1 (default value) and we only experimented with varying w_1 values. As recommended in [3], we tried out a grid search approach with exponentially growing sequences of w_1. More specifically, we tested $w_1 = 2^{-5}, 2^{-4}, ..., 2^{10}$. Once the best w_1 in this sequence was found we conducted a finer grid search on that better region (e.g. after finding out that $w_1 = 8$ was optimal in the exponentially growing sequence we tested $w_1 = 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15$). The weight w_1 was finally set to the value yielding the highest F1 across all these experiments.

Table 6 reports the optimal w_1 weights for the ten selected runs. These tuned weights are slightly lower than expected. Observe that the ratio of predators in the collection is 142/97689. Therefore, we would expect optimal w_1's greater than 100. Instead, we got optimal w_1's substantially smaller than 100. We will carefully analyze this issue in the future.

To further analyze the sensitivity of performance to w_1, we took our ten runs and selected from them the three runs that perform the best in terms of F1. These three high performing runs are tf/idf(1g)+chat-based, tf/idf(1g)+chat-based+LWIC, and tf/idf(1g, 3g)+chat-based. Given these runs, figure 1 depicts how F1 performance changes with varying w_1. With $w_1 < 1$ performance drops substantially for the three methods. This is not surprising because w_{-1} is set to 1 and, therefore, setting w_1 lower than 1 means that

Feature Set	w_1
tf/idf(1g)+chat-based	11
tf/idf(1g,3g)+chat-based	10
tf/idf(1g)+LWIC	1
tf/idf(1g)+chat-based+LWIC	3
tf/idf(1g,3g)+chat-based+LWIC	3
tf/idf(1g)+chat-based (scaled)	4
tf/idf(1g,3g)+chat-based (scaled)	18
tf/idf(1g)+LWIC (scaled)	1
tf/idf(1g)+chat-based+LWIC (scaled)	4
tf/idf(1g,3g)+chat-based+LWIC (scaled)	64

Table 6: Optimal w_1 weight for the ten submitted runs.

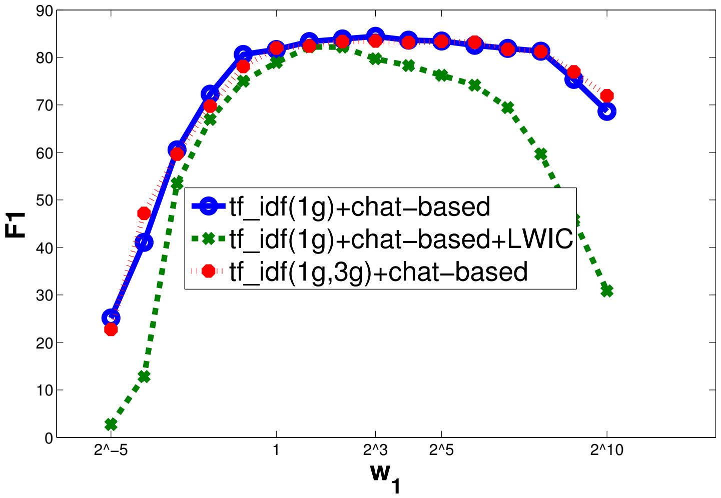

Figure 1: F1 performance with varying w_1 weights for our three best runs

we are giving more importance to the correct classification of the negative examples (non-predators). This is not a good choice for our task, which aims at finding sexual predators. With w_1 between 2^2 and 2^5, tf/idf(1g)+chat-based and tf/idf(1g,3g)+chat-based have nearly optimal performance. With $w_1 > 2^5$ performance starts to fall, showing that we are giving too much emphasis on correctly classifying the positive examples. Observe that setting w_1 to 1 yields to performance results that are not far from the optimal results and, furthermore, performance tends to nearly optimal for all w_1's in the range $[1, 2^8]$. This figure also shows that tf/idf(1g)+chat-based+LWIC is weaker than the other two methods and its performance quickly falls with $w_1 > 2^2$.

2.5 Test

The test collection contains 218702 subjects, and 254 of them are positive examples (sexual predators). The percentages of predators in the training and test collections are comparable (around 0.1%) and, therefore, both datasets are similarly unbalanced. However, it is important to observe that the number of positive examples in the training collection (142) is substantially smaller than the number of positive examples in the test collection (254). This introduces additional difficulties because the trained classifiers were built from a small set of positive examples.

The performance of our ten selected runs against the test collection is reported in Table 7. The performance of the scaled versions is always poorer than the performance of their non-scaling counterparts. This already happened in the training collection and shows that scaling our features does not work for this learning problem. Again, the tf/idf(1g)+chat-based run is the best performing run. Selecting it as our official run was indeed a good decision. In terms of F1, tf/idf(1g,3g)+chat-based, tf/idf(1g,3g)+chat-

Feature Set	P	R	F1
tf/idf(1g)+chat-based	93.92	66.93	78.16
tf/idf(1g,3g)+chat-based	94.74	63.78	76.24
tf/idf(1g)+LWIC	78.05	62.99	69.72
tf/idf(1g)+chat-based+LWIC	90.11	64.57	75.23
tf/idf(1g,3g)+chat-based+LWIC	93.06	63.39	75.41
tf/idf(1g)+chat-based (scaled)	85.80	57.09	68.56
tf/idf(1g,3g)+chat-based (scaled)	76.24	60.63	67.54
tf/idf(1g)+LWIC (scaled)	64.00	50.39	56.39
tf/idf(1g)+chat-based+LWIC (scaled)	86.29	59.45	70.40
tf/idf(1g,3g)+chat-based+LWIC (scaled)	72.20	63.39	67.51

Table 7: Performance results (in percentage) of the ten selected runs against the test collection.

based+LWIC, and tf/idf(1g)+chat-based+LWIC are not far from the performance obtained by the best run. A similar relative ordering of the runs was found with the training collection.

Our best run ranked #3, out of 16 international teams participating in PAN 2012. We believe that this is a pretty decent outcome for our very first contribution to the area of sexual predator identification. Furthermore, some of our modeling decisions (e.g. the representation of the subjects taking all their conversations) are simplistic and, in the future, we might get further gains in performance from more evolved representations of the chatters.

It seems obvious that recall was our main weakness. Comparing our training results (Table 4) against the test results (Table 7) we can clearly see that we even got higher precision in the test collection. Instead, recall fell substantially at test time. In the near future, we will carefully look into this issue. This might have something to do with the existence of many predators in the test collection, and some of them might have distinctive characteristics that do not match with the trends found for the 142 predators in the training collection.

3 Line identification task

The line identification subtask was particularly difficult because there was not labeled data. We did not have examples of predatory lines and, therefore, the participation of the teams in this subtask was somehow blind. Observe also that some of the features that we used for the subject identification subtask cannot be used at line level. For instance, the chat-based features are global characteristics of the activity of a chat participant and, therefore, it does not make sense to compute them at line level. The LWIC features could be applied at line level because they are essentially word count features. Still, we felt that the main advantage of LWIC features is the ability to extract relevant psycholinguistic patterns from the complete discourse of a chatter, rather than from a single line of text. We therefore decided to also avoid LWIC features for the line identification subtask.

We took the estimated predators from each of our ten sexual predator identification runs, and processed all their lines with a tf/idf classifier. Since there were not labeled lines, we had to apply a tf/idf classifier tuned for the predator identification task. This limitation and the poor performance of the tf/idf classifier (Table 3) made that our expectations for this task were rather low. The official results for this subtask confirmed our expectations. Our submitted run performed very poorly (2% in terms of F1) and was ranked 9th among the participating teams.

4 Conclusions and Future Work

This was our first incursion into a cybercrime detection problem. We believe that we have successfully shown that a learning-based approach is a feasible way to approach this problem. We have proposed innovative sets of features to drive the classification of chat participants as predators or non-predators. Our experiments demonstrated that the set of features utilized and the relative weighting of the misclassification costs in the SVMs are the two main factors that should be taken into account to optimize performance.

In the near future we want to carefully analyze the relative importance of the individual features in each feature set. This will help to understand psycholinguistic, contextual and behavioural characteristics of sexual predators in the Internet. Moving to more evolved representations of the Internet subjects and taking into account the sequential process of predation will be also top priorities in our future research.

References

1. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. J. Mach. Learn. Res. 9, 1871–1874 (Jun 2008)
2. Hancock, J., Curry, L., Goorha, S., Woodworth, M.: On lying and being lied to: A linguistic analysis of deception in computer-mediated communication. Discourse Processes 45(1), 1–23 (2007)
3. Hsu, C.W., Chang, C.C., Lin, C.J.: A practical guide to support vector classification. Tech. rep., Department of Computer Science, National Taiwan University (2003), http://www.csie.ntu.edu.tw/ cjlin/papers.html
4. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 28, 11–21 (1972)
5. McGhee, I., Bayzick, J., Kontostathis, A., Edwards, L., McBride, A., Jakubowski, E.: Learning to identify internet sexual predation. International Journal of Electronic Commerce 15(3) (2011)
6. Mihalcea, R., Strapparava, C.: The lie detector: Explorations in the automatic recognition of deceptive language. In: Proc. ACL-IJCNLP 2009 Conference. pp. 309–312 (2009)
7. Nallapati, R.: Discriminative models for information retrieval. In: Proc. ACM SIGIR conference on Research and development in information retrieval. pp. 64–71 (2004)
8. Newman, M., Pennebaker, J., Berry, D., Richards, J.: Lying words: Predicting deception from linguistic styles. Personality and Social Psychology Bulletin 29(5), 665–675 (2003)
9. Pennebaker, J.W., Chung, C.K., Ireland, M., Gonzales, A., Booth, R.J.: The development and psychometric properties of liwc2007 @ONLINE (Jun 2012), http://www.liwc.net/LIWC2007LanguageManual.pdf
10. Pennebaker, J., Mehl, M., Niederhoffer, K.: Psychological aspects of natural language use: Our words, our selves. Annual review of psychology 54(1), 547–577 (2003)