

# PEEP - Privacy Enforcement in Email Project

Narjès Boufaden, William Elazmeh, Stan Matwin<sup>†</sup>, Nathalie Japckowicz

School of Information Technology and Engineering University of Ottawa, Ottawa, Canada.

Email: {boufaden,welazmeh,stan,nat}@site.uottawa.ca

<sup>†</sup> is also affiliated with the Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland.

**Abstract**—Breaching information privacy is a critical problem where legal remedies intervene only after the fact rather than prevent it. This paper presents an organizational privacy compliance engine that monitors outgoing emails to detect breaches of a privacy policy in an organization. The PEEP system employs email content analysis techniques to extract information and their ownership. Access to the extracted information is verified by privacy rules assisted by an ontology-based model to represent information disclosure privileges. This paper addresses the issues of, first, the information extraction techniques from email, and second, the implementation of an ontology-based model of multi-level disclosure privileges to represent privacy rules. We experiment with the PEEP system on real life emails in an academic environment to detect breaches of privacy in emails. Our results report an F-score of 71.7% of privacy violations detection.

## I. INTRODUCTION

Information privacy is a major concern surrounding Information Technology. In recent years, many countries have introduced information privacy legislations such as; the HIPAA act in the US, Bill 31 in Ontario, and the PIPEDA (Privacy Information Protection in Electronic Documents Act) in Canada. However, legal remedies intervene only after privacy has been breached and remain unable to prevent it. Technical solutions are necessary to prevent the disclosure of private information before it occurs. Since email is a tool of communication for many organizations, it becomes an instrument to violate information privacy. Such violations may occur due to human error, for instance, wrongful email addresses in the CC field can disclose a private email message to unauthorized recipients. Although privacy violation is a significant problem, proposed solutions fail to extend beyond the lexical level of detecting and matching data against encoded privacy rules. Yet, it is obvious that detecting privacy violations requires inference of rules relating knowledge of individuals to information being communicated. Therefore, it is essential to introduce a knowledge-based representations to control information disclosure privileges, first, by information extraction (IE) techniques to analyze email contents, and second, by logical inference to verify compliance with privacy rules.

Existing work in privacy compliance in emails, generally, uses keyword-based approaches to detect potential privacy breaches. Vericept<sup>1</sup> detects potential information leaks of SIN and credit card numbers using a list of keywords. The *Policy Patrol Enterprise* software by Red Earth<sup>2</sup> provides an environment for editing privacy rules using a list of keywords with

word scores to trigger privacy rules. While such an approach imposes little demand on text processing, it fails to provide a complete description of private information contained in an email. They ignore information of ownership and levels of disclosure privileges. Such added knowledge can improve the accuracy of detecting privacy breaches resulting in a more generalized framework for privacy compliance with rules of multiple levels.

In this paper, we address the two issues of analyzing the content of emails and the verification of privacy compliance. We introduce the use of ontologies to model hierarchical information of disclosure privileges to express privacy rules. Our Information Extraction (IE) system provides contextual information by identifying ownership of information bits, while the ontology provides a formal description of these information bits and their levels of disclosure privileges. We present the PEEP privacy compliance system which operates in an organizational environment (e.g. a hospital, or a university) governed by a set of privacy rules indicating the legal responsibility of any breaches of privacy. Our prototype experiments with privacy violations in an academic setting while protecting private information in the form of student id's and names associated with their grades for a particular course. Information disclosure privileges are defined by the guidelines put forward by the Council of Ontario Universities on the freedom of information and privacy protection<sup>3</sup> along with the University of Guelph information privacy policy<sup>4</sup>.

## II. PEEP SYSTEM ARCHITECTURE

The PEEP system consists of two main components. The first is the email content analysis component which relies on information extraction (IE) techniques to extract information from the email and represent them as facts associated with their owners (individuals involved in the communication). The second component is the privacy policy compliance verification in which the PEEP system determines a successful match between the extracted facts (representing email contents and disclosure privileges of sender and recipients) against encoded privacy rules and outputs the resulting match. The architecture of the PEEP system is illustrated in figure 1. In this section, we briefly review each component.

<sup>1</sup><http://www.vericept.com>

<sup>2</sup><http://www.redearthsoftware.com/>

<sup>3</sup>[http://www.cou.on.ca/\\_bin/publications/onlinePublications.cfm](http://www.cou.on.ca/_bin/publications/onlinePublications.cfm)

<sup>4</sup><http://www.uoguelph.ca/info/privacyguidelines/>

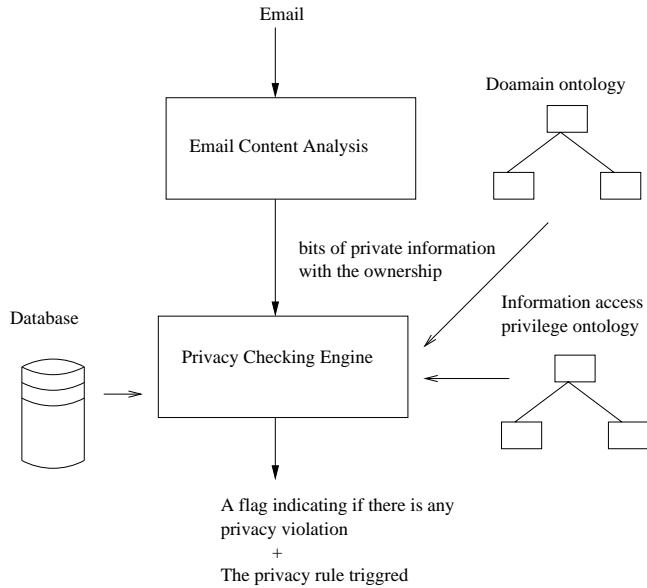


Fig. 1. A generalized architecture of the PEEP system. Assisted by two ontologies and a database, the information extraction (IE) performs email content analysis to produce facts and owners of information. These facts are then matched against privacy rules to determine a successful match. The output shows the result of the matching process.

### A. Knowledge of the domain

Information in the university domain is available in many forms. The university database contains information describing individuals (teachers and students), objects (courses and grades), and their relations (teachings and registrations). We design two additional ontologies. The first is the domain ontology which describes objects and attributes defined in the database. The second is the information disclosure privilege ontology that describes privileges and conditions required to access these objects defined by the privacy policy. In this section, we review the database and describe both ontologies.

1) *The university database*: contains information describing individuals (teachers and students), objects (courses and grades), and their relations (teachings and registrations). Each is implemented as a database table. Personal data of individuals associated with the university lists attributes such as; name, id, SIN, address, etc. Student data describes the current status of students in the university by attributes like; id, type (graduate or undergraduate), starting term of registration, etc. Finally, Staff data describes employee status in the university with attributes such as; id, type (academic or administrative), rank, office, etc. Furthermore, we implement additional tables to describe objects such as programs, courses, degrees, departments, faculties, etc. Individuals and objects are involved in activities modeled by relations. We model three such activities, they are: students registering in courses, staff members teaching (or students assisting in teaching) courses, and programs (or degrees) requiring courses. These relations have a significant impact on privileges granted for individuals to access information. For simplicity of data access, the database is implemented in Prolog, however, commercial databases such as *mySQL*, can easily be interfaced with Prolog.

2) *The domain ontology*: is a hierarchical organization of database objects into classes mapped to the database tables (e.g. the class *Person* is mapped to the *Person* table). The database provides information about relations between objects (e.g. students registered in courses) while the domain ontology provides added knowledge of individuals, their roles, and categories (e.g. a student is a person, a course is a concept which may contains private information such as grades). The PEEP system consults with the domain ontology to obtain knowledge during two stages. The first is during email content analysis stage where the domain ontology assists in the semantic tagging and in the learning of extraction patterns from email text. The second is during the privacy compliance verification stage. The domain ontology models the organizational structure of roles and ranks of actors. Each person class is a potential actor in the disclosure of private information scenarios associated with specified levels of disclosure privileges (described by the second ontology). The domain ontology is a tree divided into two abstract classes; the Physical and the Conceptual Entity classes. The subclasses of the Physical Entity class are the classes of objects found in the database such as; person, student and departments, whereas the conceptual entity class subsumes classes such as course and program.

3) *The information disclosure privilege ontology*: is a hierarchical organization of database attributes with respect to roles and types of individuals releasing the information. The motivation is to model levels of disclosure privileges to represent contextual information of the organization. Existing privacy compliance systems employ a flat representation of “information privacy disclosure” represented as a list of keywords to implement privacy rules regardless of the individuals who either release or receive information. The PEEP system consults with information disclosure privilege ontology to obtain knowledge during the privacy compliance verification stage. The ontology models the organizational structure of roles and ranks of actors. Each person class is a potential actor in the disclosure of private information scenarios associated with specified levels of disclosure privileges (described by the second ontology).

Hence, privacy rules are designed to abstracts specific disclosure privileges that can be granted for different individuals of different ranks and roles in the organization. The organization of the disclosure privilege ontology is based on types of disclosure privileges which specify “who has the right to access what”. Types of persons are determined by the database to comply with the desired privacy policy. We distinguish between three classes of information disclosure privileges; public (accessible by everyone), private (accessible by owners), and person-type dependent (specified person-types) disclosure privileges. The latter takes into account the role and rank of the person-type releasing the information.

### B. Email content analysis

Aided by the database, the email content analysis process begins with a preprocessing stage to extract email header information identifying the sender and the recipients of the email. The preprocessing procedure converts information describing

| Words               | Recall | Precision | F-score |
|---------------------|--------|-----------|---------|
| Person names        | 72.6%  | 85.6%     | 78.6%   |
| Student identifiers | 96.7%  | 97.8%     | 97.2%   |
| Numbers             | 69.6%  | 96.7%     | 80.9%   |
| All                 | 79.6%  | 93.4%     | 85.6%   |

TABLE I  
RESULTS OF THE SEMANTIC TAGGER.

each individual involved in the email (names, types, ids) into a Prolog predicate form, then, proceeds to translate abbreviations (e.g. TA is *teaching assistant*). The subsequent step is the information extraction (IE) task.

Information extraction is about finding structures of relevant information in text for a particular domain. It provides a shallow understanding of the text by highlighting target information and relevant relations among them. In our academic context, relevant information can be; student ids, names, addresses, course codes and marks. Furthermore, we target relations between student names and assignment marks of the form “Y has mark X for assignment Z”. Our IE system takes the email body as input, and outputs private information with their ownership as Prolog predicates. This IE engine performs two main tasks:

1) *Shallow semantic tagging*: In this stage, we annotated named entities such as persons and numbers, student identifiers, but also verbs to characterize the context of relevant information which are in this case attributes of the relation “the assignment mark X of student Y”. In addition, by annotating other semantic classes such as relevant verbs or student identifiers we reduced the data sparseness to help learning the extraction patterns.

Student identifiers are recognized by their format, while person names are recognized by the parser when detecting nouns starting with an uppercase letter. However, it is often the case that in emails person names do not begin with uppercase letter. To get around this problem, we generated automatically a dictionary of person names from the database to check lowercase names. The semantic tagger was tested on 1640 words and expressions and the precision and recall scores for more major semantic classes are given in table I.

Some errors occurred with nouns written with lower case. Since we are systematically matching untagged noun phrase against the list of names in the dictionary, we found that some nouns such as mark (the score) are listed as person names. This mislead the semantic tagger which annotated some common nouns as persons. Another error was the word accuracy, where some irrelevant words were associated with a particular keyword and tagged as part of the expression.

The word accuracy is an issue for the privacy checking, because noisy IE output mislead the privacy checking engine. For instance, consider the following output generated by the IE system: `id(['mark', 'change', '1597904'])` which is the result of a problem during the parsing. In this example, the privacy checking engine is expected to have a fact more like `id(['1597904'])` that it will use to access the database records in the file *Student*. In this particular, the access would fail because it will not have the right information.

2) *Pattern matching*: This stage is divided into two parts. The first part learns extraction patterns from semantically annotated emails. It uses Markov models [1] to learn relevant sequences of semantic tags along with their semantic roles. This stage allows the detection of the target relation “the assignment mark X of student Y”. For this purpose we trained a first order Markov model with states representing the targeted semantic roles which are :

- `student` referring to the argument “Y” of the relation “the assignment mark X of student Y”.
- `mark` refers to the mark assignment “X” of a student.
- `verb_score` which groups a list of verbs introducing scores such as `received` and `scored`.

We evaluate the Markov model with 10-fold cross validation with 65% of the corpus for the training and 35% for the test and obtained an average F-score of 78.25% .

The second part is the extraction of individual facts, (the arguments “X” and “Y”), by matching the patterns learned. We evaluate the IE system by choosing the Markov model with the F-score closest to our average F-score to avoid overfitting. The evaluation was made for 83 relations identified manually from the 205 emails and we obtained an F-score of 77.3%.

The analysis of the output shows that classification errors occur when there was no informative context, for example missing keywords such as the verbs `received` and `scored`. Most common error was with numbers being tagged as irrelevant where actually they were referring to marks. In addition, the small size of our corpus puts limits in the interpretation of our results.

### C. Privacy Compliance Verification Engine

The second main component of the PEEP system is the privacy compliance verification engine. This engine takes as input facts and relations (extracted by both the pre-processing and by the IE components), then, searches for a successful match between these information predicates and the encoded privacy rules. On the one hand, the existence of a successful match implies that information access has been granted for a particular individual. Unsuccessful matches indicate potential breaches of privacy. On the other hand, the inexistence of a match suggests that the system is faced with an unknown access scenario. For instance, a privacy rule is a Prolog predicate that links a particular domain ontology class to a particular information disclosure privilege class designed to represent a valid disclosure privilege granted for an individual (of that domain class) to access a specified set of database attributes. Therefore, for each individual involved in the email exchange, the process of verification searches for a successful rule to grant their access to each information predicate. The database provides data describing the individual. The domain knowledge ontology produces the class type and role of that individual. The information disclosure privilege ontology defines access privileges for that individual. Then finally, a matching and successful privacy rule is triggered granting the access. The rule may specify additional access conditions, such as the individual may be required to be an owner of the information

```
Date: Thu, 17 Apr 2003 14:29:24 -0400 (EDT)
From: SFName SLName <SLName@university.ca>
To: RFName RLName <RLName@university.ca>
Subject: Re: Marks?
```

```
Hi RFName,
I looked at FName1's Test1, his score is 40/40.
Cheers,
SFName
```

TABLE II

A TA EMAILS THE PROFESSOR A STUDENT'S MARK.

predicated being exchanged. For example, an individual of a general class *Person*-type can only access their own *Private*-class information. Accordingly, the privacy compliance is a process of verification that every individual involved in the email can be granted access to every information predicated found in the email. If any such accesses fail to be granted, a potential breach of privacy is flagged to trigger the appropriate desired action (reporting the breach).

### III. EXPERIMENTS AND RESULTS

Our data is composed of 205 emails (see figure II for an example of the data) involving students, professors, and teaching assistants. such texts fall into the category of unstructured text [2] similar to manually transcribed spontaneous speech with ungrammatical sentences (10.5% in 667 utterances of our data). Emails may contain repetitions, omissions (absence of a word from an utterance), misspellings (0.03% and 1.13% respectively in 12134 words of our data), acronyms, and lack explicit punctuations. In order to evaluate the performance of the privacy compliance engine independently, we modify the type entries in the database of selected recipients to cause breaches in student information disclosure (e.g. marks). We label the emails to contain at least one privacy violation or none. The results of running the PEEP system on these emails show the recall of 61.5%, the precision of 85.9% and the F-score of 71.7%. The analysis of the errors suggests that most were due to failures to making decisions. These failures are caused by; missing data from the database, missing email header (possibly a reply to an email), or by the inability of the IE system to extract relevant relations from the email. While the first and second type of errors can be addressed easily, the third error is a complex issue in the field of information extraction. Learning extraction patterns capable of extracting relevant information is faced with two challenges. The first is the challenge of data sparseness resulting from linguistic variations and various grammatical structures of conveying relevant information. The second challenge is co-reference resolution which links bits of related information (the object and its antecedent) to enhance the description of the private information.

### IV. CONCLUSIONS AND FUTURE WORK

In this paper we presented a novel approach to the problem of privacy compliance in emails. We introduced the use of an

IE-based email content analysis and an ontology-based representation of domain information and disclosure privileges. The IE system extracts private information along with their ownership to enhance the description of information released in an email. The domain and information disclosure privilege ontologies define a general framework to represent hierarchical relations among individuals and their disclosure privileges in an organization. This design facilitates the implementation of multi-layer privacy rules of different access privileges. Information extraction from emails is a relatively new problem in comparison to IE from newspaper articles. For instance, Soderland [2] uses semantic classes to learn regular expressions from on-line rental ads extracting individual facts with an F-score of 94%. However, ads are shorter texts with a restricted format more than emails. The closest work to ours was developed for the CALO (A Cognitive Assistant that Learns and Organizes) project, which aims to extract information about people and other entities such as person names, job titles and addresses. They use a conditional random field model [3] to learn Markov models to extract these informations. On emails from the Enron corpus [4], they achieved an average F-score of 80.8%.

While the results achieved by the overall privacy compliance verification engine are encouraging, there is room for improvement. In particular, during both the pre-processing stage and during the semantic tagging stage. We are also aware of the small size of our corpus and we plan to work on larger corpus. In future work, we plan to tackle two issues. The first one is the integration of the EPAL language (developed by IBM) in our design by translating the information disclosure privileges ontology into an EPAL description, so it would be expressed in a standard way, allowing its use for other privacy applications. The second issue is applying our system to the health care domain. We are collaborating with The Ottawa General Hospital (TOH) on this application of research.

### V. ACKNOWLEDGMENT

We thank Yimin Ma, Nourredine El-Kadri and Quintin Armour for their help and discussions in this project. We acknowledge the support of the Communications and Information technology Ontario (CITO) and the Natural Sciences and Engineering Research Council of Canada (NSERC) for this research.

### REFERENCES

- [1] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, vol. 77, no. 2, 1989, pp. 257–286.
- [2] S. Soderland, "Learning information extraction rules for semi-structured and free text," *Machine Learning*, vol. 44, no. 1-3, pp. 233–272, 1998.
- [3] A. Culotta, R. Bekkerman, and A. McCallum, "Extracting social networks and contact information from email and the web," in *the First Conference on Email and Anti-Spam (CEAS)*, 2004.
- [4] B. Klimt and Y. Yang, "The Enron Corpus: A New Dataset for Email Classification Research," in *In proceedings of the European Conference on Machine Learning*, Pisa, Italy, 2004.