2017 IEEE 33rd International Conference on Data Engineering

TABOO: Detecting unstructured sensitive information using recursive neural networks

Jan Neerbek*†, Ira Assent‡ and Peter Dolog§
*Department of Computer Science, Aarhus University, Denmark
†Alexandra Institute, Denmark, Email: jan.neerbek@alexandra.dk
‡Department of Computer Science, Aarhus University, Denmark, Email: ira@cs.au.dk
§Department of Computer Science, Aalborg University, Denmark, Email: dolog@cs.aau.dk

***Abstract*—Leak of sensitive information from unstructured text documents is a costly problem both for government and for industrial institutions. Traditional approaches for data leak prevention are commonly based on the hypothesis that sensitive information is reflected in the presence of distinct sensitive words. However, for complex sensitive information, this hypothesis may not hold.**

Our TABOO system detects complex sensitive information in text documents by learning the semantic and syntactic structure of text documents. Our approach is based on natural language processing methods for paraphrase detection, and uses recursive neural networks to assign sensitivity scores to the semantic components of the sentence structure.

The demonstration of TABOO focuses on interactive detection of sensitive information with the TABOO system. Users may work with real documents, alter documents or prepare free text, and subject it to information detection. TABOO allows users to work with our TABOO engine or with traditional approaches, and to compare results. Users may verify that single words can change sensitivity according to context, thereby giving hands-on experience with complex cases of sensitive information.

Video: https://youtu.be/tqQ4BP0wqs8

I. Detecting Complex Sensitive Information

Detecting and redacting sensitive information in documents prior to publication, *Data Leak Prevention* (DLP), is increasingly important in industry and government, as more and more documents are made available publicly [1], [2].

Traditional approaches for DLP such as n-grams [2] or inference rules [1], assign sensitivity scores to words directly without considering context. These traditional approaches, including NLP inspired sentiment analysis, topic modeling (e.g. Latent Dirichlet Allocation) or Named Entity analysis [3], generally perform well for *private* information; usually entities such as location or personally identifiable information [3].

However, a core challenge in DLP is that the definition of *sensitive* information is often human specified and complex in nature. An entity, such as a company name, may be sensitive in one context and non-sensitive in another context, or sensitive information may not even be captured by a single name or term alone.

Consider the real case that we use in our demo, namely the internal and external communication of Enron, that was published when the company was prosecuted for fraud [4]. The documents contain both sensitive and non-sensitive content (manually labeled by experts with respect to complex issues such as "prepay transactions" [5]).

An example for such *complex* sensitive information are so-called "prepay transactions", where "letters of credit" may be sensitive, but not if discussed in the context of e.g. possible loan of money. We observe that sensitive information may be embedded in the semantic meaning of the text, even when the text contains only "non-sensitive" words.

Our *TABOO* system extracts compositional sub-structures of the sentences to learn sensitivity scores. It builds on successful NLP methods including paraphrasing, sentiment analysis and image-sentence ranking [6], [7]. TABOO is motivated by the observation that complex sensitive information bears some similarity to paraphrasing. Consider the (real) examples "We need letters of credit, with approved collateral in order to approve the prepay transaction" containing the sensitive keyword "prepay transaction". It is a paraphrase of "we have proposed letters of credit for the approved form of collateral pending further discussion" (shortened in the interest of space). Also this example is sensitive even in the absence of the keyword.

In TABOO, a Recursive Neural Network (RNN) processes the sentences in a structured way. The same neural network is applied recursively according to the structure given by the syntax tree of the text. At each node in the syntax tree the RNN generates a *representation* of the particular compositional sub-structure captured by that node. These representations have been shown to successfully predict whether one sentence *paraphrases* another [6].

II. The TABOO Engine

The TABOO system takes a set of documents containing sensitive information as training input. Using the trained model the system then detects sensitive content in new documents. TABOO consist of a number of steps to process the input. As a learning system TABOO has two different modes of operation; training mode and predicting mode. Training mode is used when training the RNN model for improved predictions. When TABOO is introduced to a new domain a training set must be provided and a RNN model must be trained. The user can also choose to retrain an existing model, if, say, the definition of sensitive changes over time. Once a model is obtained the system can be used in predicting mode. Here, TABOO takes a document and detects sensitive information in the document according to the model.

2375-026X/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDE.2017.195

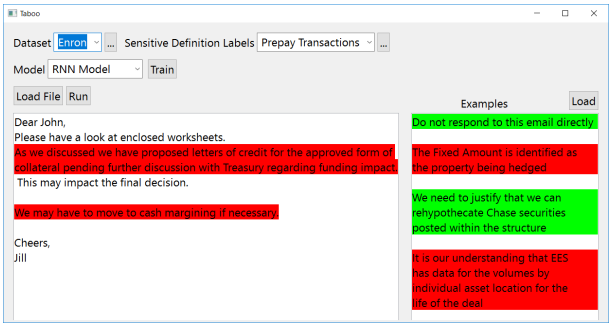

Fig. 1. Main screen of the TABOO system. The user can load a document or edit the text directly. Clicking "Run" subjects the document to sensitive information detection under the selected model and definition of sensitive information ("Dataset"). On the right, the user has access to samples of sensitive and non-sensitive content.

TABOO input documents are either loaded by scanning an input directory or through the TABOO graphical user interface where the user can load or write a document directly. Each document is then subjected to 1) Sentence splitting, 2) Syntax Parsing and 3) RNN. 1) Splits the document into sentences for further processing, using NLTK [8], the natural language toolkit for Python. 2) Extracts substructures of sentences in the form of a parse tree reflecting the structure of the sentence using the Stanford NLP parser [9]. 3) Trains a model (or detects using an existing model) on the syntax trees using Deep Recursive Neural Networks as developed by İrsoy and Cardie [10]. Training employs backpropagation-through-structure with dropout, evaluating recursively in feed-forward manner [10] and persists the model for future use. Detection outputs the highest scoring prediction per sentence under the model.

III. The TABOO System and Demo

Our interactive demo allows the user to load any document or devise any text, and subject them to a number of sensitive information detection approaches using different selectable definitions of sensitivity. The TABOO system also allows the user to load a set of documents for training a new model under a new definition of sensitivity.

For the demo, we additionally prepare real case text samples from the Enron case that users can embed in documents for testing purposes or may alter them freely to test the detection capabilities.

The TABOO demo comes with 3 different prediction engines, The RNN model (our approach), n-gram model [2] and inference rules model [1].

In Fig. 1 the TABOO system interface is shown. The left frame is the main frame that holds the document for detection which is either loaded from file, created or edited directly. The right side of the screen contains samples of real sensitive and non-sensitive text for validation purposes. These samples are not used in training to avoid any bias, and are color-coded red and green to indicate sensitive and non-sensitive text snippets, respectively. Clicking allows copying them to the document in the main frame. These samples can be used as-is (in particular when working with a new domain or new definition of sensitivity) or altered to challenge the detection capabilities of the different models.

The underlying corpus (*Dataset*), as well as the sensitive definition given through labeled training documents (*Sensitive Definition Labels*), can be selected. The system comes with pre-generated models for 8 different definitions of sensitive information. The user can also choose which particular approach to use for detecting sensitive content; "RNN model", "2-gram" or "inference rules". The latter two are traditional approaches. Clicking the *Run* button executes the model on the document in the main window and highlights sentences with sensitive content (in red).

In interacting with the demo, conference attendees will gain hands-on experience with the differences in detection power of different engines, and insights into different complexities of sensitive information and the challenges associated with them. The Enron case contains accessible sensitive information that is difficult to characterize, allowing users to develop an intuition of when keyword-based approaches suffice and when the definition of sensitive information is so complex that it requires a structure-based model such as the RNN model we propose in our TABOO engine.

TABOO is designed as an analysis tool for determining the best approach for a given application and for validating documents used for training. I.e., in interacting with the system, the analyst can verify whether particular types of sensitive information are successfully captured or whether more or different training data is required. Different detection models can be persisted and used for comparison and deployment in practice, making TABOO a tool to manage different redaction requirements prior to document publication.

Acknowledgment

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 645198 (Organicity Project)

References

[1] R. Chow, P. Golle, and J. Staddon, "Detecting privacy leaks using corpus-based association rules," in *Proc. ACM SIGKDD*, 2008.

[2] M. Hart, P. Manadhata, and R. Johnson, "Text classification for data loss prevention," in *Proc. PETS*, 2011, pp. 18–37.

[3] A. C. Islam, J. Walsh, and R. Greenstadt, "Privacy detective: Detecting private information and collective privacy behavior in a large social network," in *Proc. WPES*, 2014, pp. 35–46.

[4] B. Klimt and Y. Yang, "Introducing the Enron Corpus," in *Proc. CEAS*, 2004.

[5] S. Tomlinson, "Learning task experiments in the trec 2010 legal track," in *Proc. TREC*, 2010.

[6] R. Socher, E. H. Huang, J. Pennin, C. D. Manning, and A. Y. Ng, "Dynamic pooling and unfolding recursive autoencoders for paraphrase detection," in *Proc. NIPS*, 2011, pp. 801–809.

[7] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *Proc. NIPS*, 2015.

[8] S. Bird, "NLTK: the natural language toolkit," in *Proc. COLING/ACL*, 2006, pp. 69–72.

[9] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proc. ACL*, 2014, pp. 55–60.

[10] O. İrsoy and C. Cardie, "Deep recursive neural networks for compositionality in language," in *Proc. NIPS*, 2014, pp. 2096–2104.