Automatic Declassification of Textual Documents by Generalizing Sensitive Terms

Veena Vasudevan
PG Scholar, CSE
T.K.M College of Engineering
Kerala, India

Ansamma John
Associate professor, CSE
T.K.M College of Engineering
Kerala, India

ABSTRACT
With the advent of internet, large numbers of text documents are published and shared every day .Each of these documents is a collection of vast amount of information. Publically sharing of some of this information may affect the privacy of the document, if they are confidential information. So before document publishing, sanitization operations are performed on the document for preserving the privacy and inorder to retain the utility of the document. Various schemes were developed to solve this problem but most of them turned out to be domain specific and most of them didn't consider the presence of semantically correlated terms. This paper presents a generalized sanitization method that discovers the sensitive information based on the concept of information content. The proposed method removes the confidential information from the text document by first finding the independent sensitive terms. Then with the use of these sensitive terms the correlated terms that cause a disclosure threat are discovered. Again with the help of a generalization algorithm these sensitive and correlated terms with high disclosure risk are generalized.

General Terms
Text Mining, Privacy Preserving Data Publishing, Redaction, Sanitization

Keywords
Document Declassification, Generalization, Information content, Privacy, Term correlation, Unstructured Data, utility

1. INTRODUCTION
The growth of information sharing applications led to the increase in the number of documents being shared. But the risk of violating the privacy of an individual or organization also increases. Researchers are trying to find out the problems associated with sharing the private data and the remedies for it [1]. They had also looked into the importance of anonymity and/or privacy in diverse application areas: e-voting [2], Electronic Health records [3], social networking [4], electronic mail [5], etc. The information which is confidential or reveals the identity of a person or organization is considered as sensitive. So before sharing the document the sensitive information must be removed in such a way that the privacy and utility is retained.

Document sanitization is the process of removing sensitive or confidential information from a document. The main objective of sanitizing the document is to preserve privacy but at the same time the utility is retained. Various schemes are used to identify and protect sensitive data. So document sanitization is a two step process. In earlier days, the sensitive information is identified and removed manually. But it proves to be time consuming, tedious and expensive. It also does not scale well as the volume of text data increases. So semi-automatic and automatic methods are developed. The proposed system is an automatic document sanitization system which can be used to sanitize all types of documents irrespective of a particular field.

The first stage is identification of sensitive information. It again is a two step task: first independent sensitive terms are detected [6], [7] and in the second step semantically correlated terms, which posses a high disclosure risk, are identified. The terms obtained in the above two steps together form the sensitive terms, which are the input to the second stage .In the second stage of document sanitization the detected sensitive terms are sanitized. Inorder to preserve the utility of the document the sensitive terms are replaced by their generalized versions [8], [9], [10].

2. RELATED WORK
In the early stage of document sanitization, more work is done on sanitizing the structured documents such as relational databases. Later the need to sanitize the unstructured documents had come into notice. This need is revealed in initiatives from DARPA [11] or the Consortium for Healthcare Informatics Research (CHIR)[12] which aim at building new methods and tools for declassification of confidential documents. In the structured documents the structure itself provides the key to identify sensitive terms. But in an unstructured text document sensitive information identification is a difficult task.

Earlier it was done manually by trained experts, who remove the sensitive terms from the document based on standard guidelines [13] and rules. It proved to be costly, time consuming and does not scale well as the volume of data increases. Hence semi-automatic and automatic methods were proposed.

In earlier days the sanitization is performed in medical documents inorder to hide the sensitive information's related to patients. The information is treated as sensitive based on the Health Information portability and Accountability Act (HIPAA) of 1996.According to the safe harbor rules 18 entities are considered as sensitive terms such as name, geographic locations, dates, e-mail address, telephone numbers etc. Latanya Sweeney [14] proposed a system called scrub, based on this rule, to identify the identifiable information in a patient's record. It uses detection algorithms and replacement algorithms to identify the sensitive details and for replacing them respectively. Regular expression type templates and knowledge sources are used to detect sensitive terms. It detects almost 99-100% of personal information but it fails to detect the nick names, additional phone numbers,

reference to family members, etc. Douglass et.al [15] also proposed a system to identify patient details in a patient record based on HIPAA act. The main purpose is to automatically detect and remove Protected Health Information (PHI) from nursing notes. They use lexical look-up tables, simple heuristics and regular expressions to identify personal health information in free text nursing notes with an overall sensitivity of 0.92. But the look-up tables have some major limitations such as the scarceness of common abbreviations, names of drugs etc. Also this method suffers from having high false positive rates, so an additional step of manual reviewing of selected terms is essential.

Chakaravarthy et.al [16] proposed a system named Efficient RedAction for Securing Entities (ERASE) in which a database of entities is used to model public knowledge. Manual compilation of database is not required. Each entity is tagged with a set of related terms called the context of the entity. Some entities are marked as protected to provide privacy. The sanitized document obtained is least distorted, thereby increasing the utility of document. "K-safety" concept is used to set the desired privacy and utility levels of the document. K-safety demands that the maximum subset of each sensitive entity's context incorporated in a document is also included by the contexts of atleast K other entities. The use of specific purpose knowledge bases hinders the applicability of the technique when the sanitization needs differs.

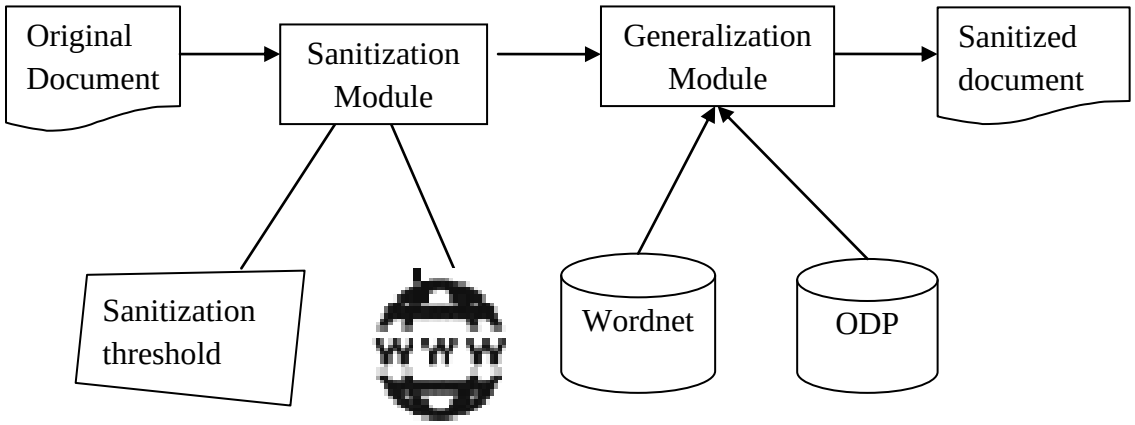

Figure 1: Overall System architecture

Daniel et.al [17] proposed a system based on the use of trained classifiers to find the sensitive terms. They used the term private entity to refer an entity which reveals information about the correspondents (organization or individual) directly or indirectly associated with the document. Private entity recognition and Private entity protection are the two tasks associated with it. Named Entity Recognition (NER) technique is used to identify the private entities in a document. Proper locations, names or organizations are the typical named entities. Generalization of entities, swapping of entities, and adding noise to entities are the various private entity protection methods. The most generic Named Entity Recognition packages limit the type of named entities. Also every sensitive term cannot be represented using named entity and similarly all named entities may not be sensitive. With reference to their specificity and the fact that they represent individuals rather than concepts, NEs are likely to reveal private information.

To overcome the limitations in [17] Sanchez et.al [7] proposed a method based on information theory and using web as corpus. The information content metric is used for identifying the sensitive terms in a document. Information Content (IC) of a term measures the amount of information provided by the given term when appearing in a context. The terms that have an information content value equal to or above a given threshold is identified as sensitive terms. The threshold value refers to the value of the feature in the document which needs to be protected. So it will be the information content value of that particular feature. In a text, the Noun Phrases (NP) usually indicates the concrete concepts or individual names. So they proposed NPs as the candidates for sensitive terms. They performed generalization of sensitive terms to preserve the utility of the document. The above techniques evaluate the sensitivity of the terms by treating them as independent variables. However the presence of other terms can also infer the already sanitized terms. This destroys the privacy of the document. The term relatedness is not a widely studied area related to text sanitization.

3. PROPOSED SYSTEM

The Proposed system performs automatic detection of sensitive terms using the information content and using web as the corpus and then generalizes the sensitive term. The overall block diagram can be represented as shown in Figure 1.

3.1 Sanitization Module

The sanitization module performs two functions. Initially, it identifies the independent sensitive terms present in a document. Secondly, it detects the correlated terms that infer the sensitive terms identified earlier from the non-sensitive terms. The concepts of information theory are used as the basis to identify the sensitive terms. The detailed block diagram of sanitization module is as shown in fig 2

3.1.1Independent Sensitive term detection

Sensitive terms are those that need to be hidden from a document based on a user's prerequisite. These terms are mostly the noun phrases (NPs) in a text document. Identification of sensitive terms can be done with the help of amount of information provided by the terms. Noun phrase detection consists of several phases such as sentence detection, tokenizing, part of speech (POS) tagging and syntactic parsing. The input text is divided into sentences using the sentence detector. Then tokenizer is used to convert

each sentence into a stream of tokens. POSTagger is used to tag each token in each sentence to corresponding word type. The chunker then splits the sentence into phrase chunks by looking at the POS tagged tokens. For each chunk, based on

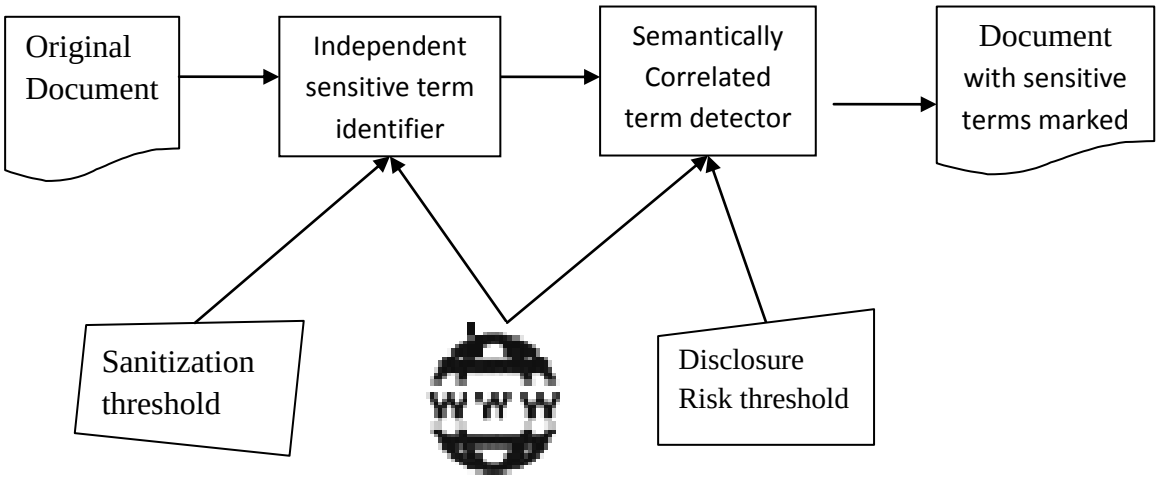

Figure 2: Overview of sanitization module

The type, only noun phrases are considered. OPENNLP can be used to perform all these tasks and finally retrieve the noun phrase. The next step after detecting NP is to measure the amount of information provided by each of them by querying them in a Web Search Engine (WSE) such as Bing. Web is used as the corpus because to calculate the information content (IC) accurately the corpus should be large, heterogeneous and up-to-date. This ensures that web provides a maximum recall and it represents the true current distribution of terms at a social scale. Formally, the IC of a term t is computed as the inverse of its probability (p) of appearance in a corpus (i.e p (t)).

$$\text{IC}(t) = -log_2 p(t)$$

When the web is used as the corpus then the probability can be calculated in terms of hit count (H) that is retrieved from the web search engine. The information content can be calculated as follows

$$IC_{WSE}(t) = -log_2 P_{WSE}(t)$$

$$-log_2 P_{WSE}(t) = -log_2 \frac{H(t)}{N}$$

Where N is the total number of web sites indexed by the specific search engine. As an intermediate output a set of NPs and corresponding IC value pairs are obtained. From the set, the pair with IC value greater than or equal to a sanitization threshold is separated and the NP corresponding to IC value is considered as sensitive.

Sanitization threshold (β) is defined based on the IC value of the concrete features. β is the sanitization criterion fixed by the user. Concrete features (Φ) are terms that the user wants to hide from the sanitized output document. Here $\Phi=\{\phi_1, \phi_2, \ldots\}$, where $\phi_1, \phi_2, \ldots$ are the different features of an entity the user want to hide. Then the β value can be computed from Φ as follows:

$$\beta = \max\left(IC_{WSE}(\phi_i)\right)$$

The independent sensitive terms (ST) are then detected as follows:

$$ST = \{NP_i | IC_{WSE}(NP_i) \geq \beta\}$$

where NP_i is the element, NP in the pair, of the set obtained as the intermediate output. The remaining NPs are considered as non sensitive terms (NST).

3.1.2 Semantically correlated term detection

The identification and protection of individual sensitive terms before publishing does not protect the document from adversaries who infer the sensitive information from already accessible information. The information that disclose the already detected and protected sensitive terms are the semantically correlated non-sensitive terms present in the document in their clear form(For example: the presence of symptoms of a particular disease in a document can infer the name of the disease even though the term is sanitized in the document). In the context of Information Theory, the amount of Mutual Information (MI) between terms, which measures the correlation between variables, can be used to extract such information. In particular, the instantiation of MI for two specific observations results in the well-known Point-wise Mutual Information (PMI), which quantifies the difference between the probability of their co-occurrence given their join distribution and their marginal distributions. In other words PMI is a measure which tells us how much one word tells us about the other. If st_j and nst_j are elements of ST and NST respectively then PMI are calculated as follows:

$$PMI(st_j, nst_j) = log_2 \frac{p(st_j, nst_j)}{p(st_j)p(nst_j)} = log_2 \frac{p(st_j|nst_j)}{p(st_j)}$$

$$= log_2 \frac{\frac{p(st_j, nst_j)}{p(nst_j)}}{p(st_j)} = log_2 \frac{(st_j, nst_j)}{p(nst_j)} - log_2 p(st_j)$$

To provide a general-purpose solution, we query WSEs to compute web-scale term probabilities obtaining the following expression:

$$PMI_{WEB}(st_j, nst_j) = IC_{WEB}(st_j) - IC_{WEB}(st_j|nst_j)$$

$$PMI_{WEB}(st_j, nst_j) = log_2 \frac{\frac{H(st_j, nst_j)}{N}}{\frac{H(st_j)}{N}\frac{H(nst_j)}{N}} = log_2 \frac{\frac{H("st_j" AND "nst_j")}{N}}{\frac{H("st_j")}{N}\frac{H("nst_j")}{N}}$$

Where H is the Hit count retrieved from the web search engine result page when querying a particular term(s) and N is the size of the particular web search engine. Double quotes are used in the queries to force the search for the exact term and the use of the AND operator to force the co-occurrence of both terms within the same document. The use of "AND" as the query operator retrieve all pages that contain the queried terms. Sometimes they may not be even related to each other. So in Bing we can use the advanced query operator "NEAR". The next step is to evaluate which term relationships may incur in disclosure risk according to their PMI values. Whenever a disclosure risk occurs, the preliminary non-sensitive term $nstj$ is proposed for sanitization.

Disclosure Risk (DR) threshold
PMI value can be used to predict the disclosure risk as follows:

$$PMI(st_j, nst_j) = \begin{cases} 0; no\ disclosure\ risk \\ IC(st_j)\ ; maximum\ disclosure\ risk \end{cases}$$

$$DR = PMI(st_j, nst_j)$$

Since most words are correlated in some degree, PMI and, hence, DR will usually be positive. From the non-sensitive terms, the one with disclosure risk value equal to or above a particular threshold is considered as risky to the already detected sensitive terms. The threshold can be said as disclosure risk threshold (t_{DR}) and computed as a function of the sanitization threshold (β). We weight this value by a user-defined parameter w that represents the relative amount (in parts per unit) of information that the non-sanitized term $nstj$ is allowed to reveal about the sensitive term stj, as follows:

$$t_{DR} = w.\beta$$

Where w is defined in the interval [0…1]

Since w weights β, and the latter measures IC, the resulting t_{DR} is also expressed in terms of IC and reflects the maximum amount of information that a non-sanitized term may disclose about a sensitive one. In this way, for any pair of terms that obtains a *PMI* value equal or above t_{DR}, we consider that the non-sanitized term reveals too much information of the sensitive one and, hence, it must be sanitized. Values of w close to 1, state that we allow revealing almost all the information provided by a sensitive term. Inversely, values of w close to 0, state that we allow revealing a very low percentage of sensitive information.

Formally, the set $R=\{rt1,\ldots,rtq\}$ of too highly related terms that may enable disclosure of elements in ST from a document $D'$is obtained as follows:
$R= \{nstj \in NST \mid \exists stj \in ST with\ PMI_{WEB}(stj, nstj) \geq t_{DR}\}$
Then the sensitive terms (ST) identified by the Independent sensitive term identifier and the related terms (R) identified by the Correlated non-sensitive term detector are passed to the generalization module for further processing.

3.2 Generalization module

Detected sensitive terms should be removed/transformed so that the amount of information they provide is null and void or, more desirably, reduced enough. Since semantics are the mean to interpret and extract conclusions from the analysis of textual data, the preservation of text semantics is important to maintain the utility of documents. Classic sanitization approaches simply remove sensitive text from the document which in turn hampers the semantics and hence utility of the document. To tackle this problem, recent methods propose replacing sensitive information by generalized versions (e.g., *"iPhone"* →*"cell phone"*). This method still retains a degree of semantics (and hence, a level of utility) while revealing less information. To enable term generalizations, a knowledge base (KB) modeling the taxonomic structure of terms to sanitize is needed. WordNet can be used as the KB for this purpose.

An ideal KB for text sanitization should have two desirable features:

1. The KB should provide a high recall, so that it covers as many sensitive terms found in the input document as possible. This is because, if a sensitive term is not found in the KB, the only option to sanitize would be to remove it [16], to replace it by a random entity [24] or to substitute it by the most general abstraction of the KB. In all cases, an excessive loss of information will occur. Some rely on ad-hoc constructed KBs offering a high recall for the sanitized documents [2], [9], [16], [22], [23], but this is neither feasible nor scalable in environments with large and heterogeneous sanitization needs.
2. The KB should offer a detailed knowledge representation, so that fine grained taxonomical trees of generalizations can be obtained for a sensitive term. In this manner, the loss of information resulting from each generalization step will be minimized.

From the data utility perspective, an optimal sanitization is such that, while fulfilling the desired level of privacy, minimizes the loss of information resulting from hiding sensitive data. In our method, the level of privacy is stated by the sanitization threshold, so that none of the terms appearing in a sanitized document provide more information than β. At the same time, sanitized terms could retain upto their original information (i.e., semantics), so that they are still useful while being general enough to minimize the disclosure risk. To achieve these, we rely on term generalization to reduce the amount of information provided by sensitive terms while retaining a degree of their semantics.

Given a set Ψ of terms to sanitize, we propose replacing them by their generalizations that provide the maximum information while fulfilling β. By picking up the generalized term that is less informative than the sensitive one and provides some information, the sanitized document retains maximum semantics and, hence, utility. To do so, each NP_i in Ψ is mapped to its conceptual abstraction in the KB. When found, the KB returns a hierarchy of generalizations $H_i = h_{i1}, h_{i2}, \ldots, h_{il}$ to which NP_i belongs. For example, if we look for *"iPhone"* (covered by ODP, but not by WordNet), ODP will return the hierarchy: "iPhone"→ "Smartphones" → "Handhelds" → "Systems" → "Computers*"*. Then, our method selects the generalization that sanitizes NP_i by looking for the h_{ij} in H_i that provides maximum IC while fulfilling β.

- Steps to get the generalized term

    1. The detected sensitive terms are first searched in the knowledge base to get a generalization hierarchy.

2. If the hierarchy is present, then select the term that provide less IC value than the threshold value from the hierarchy.
3. If the term is absent, then look for its simpler forms by iteratively removing adjectives/nouns starting from the one most on the left. e.g., "metastatic pancreatic cancer"→"pancreatic cancer" →"cancer"
4. If the simplest form of NP is not found in any of the KBs (for example, if it is misspelled), it will be replaced by the most abstract generalization.

   e.g., *"world"*

This process provides optimum sanitizations, regarding the fulfillment of the desired privacy level and the maximization of the document's utility, in an efficient manner with regards to β and the background KBs.

4. CONCLUSION

Publishing of textual documents is essential for various purposes such as research, decision making and due to regulations. But publishing the document that contains confidential information is illegitimate. The existing method only removes the independent sensitive terms from the document. But the presence of correlated terms infer the already sanitized sensitive terms so privacy is a main issue in existing system. Our method focuses both on detecting these independent sensitive terms and also identifies the correlated non-sensitive terms that infer the independent sensitive terms. The utility of the document is preserved by using generalization technique instead of removing or suppressing the sensitive terms. Privacy is preserved by finding maximum number of terms that discloses the already detected independent sensitive terms.

The correlation can exist not only between two terms but also between more terms in a context. The correlation of such terms can also infer the already sanitized sensitive terms in a document. This work can be extended to support relationships of larger cardinalities. The utility of the terms can be further improved in the generalization phase, by analyzing the generalization hierarchy in a deeper level, in the future. The correct identification of the level of abstraction reduces the utility loss and improves time efficiency. Also a log file can be integrated in the system as an extension to store the already detected sensitive terms and their generalization hierarchy in order to reduce the fetching time from the knowledge Base.

5. REFERENCES

[1] A.shamir,"How to share a secret", comun ACM,vol 22,no.11,pp,612-613,1979

[2] F. Baiardi, A. Falleni, R. Granchi, F. Martinelli, M. Petrocchi, and A. Vaccarelli, "Seas, a secure e-voting protocol: Design and implementation," Comput. Security, vol. 24, no. 8, pp. 642–652, Nov. 2005..

[3] A. Friedman, R. Wolff, and A. Schuster, "Providing k-anonymity in data mining," VLDB Journal, vol. 17, no. 4, pp. 789–804, Jul. 2008..

[4] Q. Xie and U. Hengartner, "Privacy-preserving matchmaking for mobile social networking secure against malicious users," in Proc. 9th Ann. IEEE Conf. Privacy, Security and Trust, Jul. 2011, pp. 252–259.

[5] D. Chaum, "Untraceable electronic mail, return address and digital pseudonyms," Commun. ACM, vol. 24, no. 2, pp. 84–88, Feb. 1981.

[6] Sánchez, D., Batet, M., and Viejo, A."Detecting sensitive information from textual documents: An information theoretic approach", Modeling decisions for artificial intelligence. 9th international conference, mdai ,Springer,2012 (Vol.7647, pp.173-184)

[7] D. Sánchez, M. Batet, A. Viejo, "Automatic general-purpose sanitization of textual documents", IEEE Transactions on Information Forensics and Security 8 (2013) 853–862.

[8] C. Cumby and R. Ghan, "A machine learning based system for semi-automatically redacting documents," in Proc. 23rd Innovative Application of Artificial Intelligence Conf., 2011, pp. 1628–1635.

[9] B. Anandan, C. Clifton, W. Jiang, M. Murugesan, P. Pastrana-Camacho, and L. Si, "t-plausibility: Generalizing words to desensitize text," Trans. Data Privacy, vol. 5, pp. 505–534, 2012.

[10] D. Abril, G. Navarro-Arribas, and V. Torra, "On the declassification of confidential documents," in Proc. Modeling Decisions for Artificial

[11] DARPA, New Technologies to Support Declassification Request for Information (RFI) Defense Advanced Research Projects Agency. Solicitation Number: DARPA-SN-10-73, 2010..

[12] S. M. Meystre, F. J. Friedlin, B. R. South, S. Shen, and M. H.Samore, "Automatic de-identification of textual documents in the electronic health record: A review of recent research," BMC Med. Res.Methodology, vol. 10, pp. 70–86, 2010

[13]] Nat. Security Agency, Redacting With Confidence: How to Safely Publish Sanitized Reports Converted From Word to pdf, Tech. Rep. I333- 015R-2005, 2005.

[14] L. Sweeney, "Replacing personally-identifying information in medical records, the scrub system," in Proc. 1996 American Medical Informatics Association Ann. Symp., 1996, pp. 333–337.

[15] M. M. Douglass, G. D. Cliffford, A. Reisner, W. J. Long, G. B.Moody, and R. G. Mark, "De-identification algorithm for free-text nursing notes," Proc. Computers in Cardiology'05, pp. 331–334, 2005.

[16] V. T. Chakaravarthy, H. Gupta, P. Roy, and M. Mohania, "Efficient techniques for document sanitization," in Proc. ACM Conf. Information and Knowledge Management'08, 2008, pp. 843–852

[17] D. Abril, G. Navarro-Arribas, and V. Torra, "On the declassification of confidential documents," in Proc. Modeling Decisions for Artificial Intelligence'11, 2011, pp. 235–246.