Using Part-of-Speech N-grams for Sensitive-Text Classification

Graham McDonald
University of Glasgow
Scotland, UK
g.mcdonald.1@
research.gla.ac.uk

Craig Macdonald
University of Glasgow
Scotland, UK
craig.macdonald@
glasgow.ac.uk

Iadh Ounis
University of Glasgow
Scotland, UK
iadh.ounis@
glasgow.ac.uk

ABSTRACT

Freedom of Information legislations in many western democracies, including the United Kingdom (UK) and the United States of America (USA), state that citizens have typically the right to access government documents. However, certain sensitive information is exempt from release into the public domain. For example, in the UK, FOIA Exemption 27 (International Relations) excludes the release of Information that might damage the interests of the UK abroad. Therefore, the process of reviewing government documents for sensitivity is essential to determine if a document must be redacted before it is archived, or closed until the information is no longer sensitive. With the increased volume of digital government documents in recent years, there is a need for new tools to assist the digital sensitivity review process. Therefore, in this paper we propose an automatic approach for identifying sensitive text in documents by measuring the amount of sensitivity in sequences of text. Using government documents reviewed by trained sensitivity reviewers, we focus on an aspect of FOIA Exemption 27 which can have a major impact on international relations, namely *information supplied in confidence*. We show that our approach leads to markedly increased recall of sensitive text, while achieving a very high level of precision, when compared to a baseline that has been shown to be effective at identifying sensitive text in other domains.

1. INTRODUCTION

Freedom of Information (FOI) laws exist in many countries around the world, including the United Kingdom (UK)[1] and the United States of America (USA)[2]. FOI states that government documents should be open to the public. However, many government documents contain *sensitive* information, such as *personal* or *confidential* information. Therefore, FOI laws make provisions that exempt sensitive information from being open. To avoid the accidental release of sensitive information, it is essential that all government documents are *sensitivity reviewed* prior to release.

Sensitivity reviewers are required to identify any *sequences* of sensitive text in a document, for example individual terms, sentences, paragraphs or the full content of the document, so that the sensitive text can be *redacted* or the document can be *closed*. However, the recent increase in volume of digital government documents means that the traditional manual review process is not feasible for digital government documents. Therefore, the UK and the USA governments have recently recognised the need for automatic tools to assist the sensitivity review process [2, 3].

In this work, we address the problem of automatically identifying sensitive information in government documents by directly classifying the text *within* documents. We present an approach that uses POS n-grams to measure the amount of sensitivity contained within a sequence of text, i.e. its *sensitivity load*. Moreover, we show how our approach can be used to deploy an effective sensitivity classifier that classifies sensitivity at the *term-level*.

We propose to perform classification at the term-level because correctly identifying partial sensitive sequences will be beneficial to a sensitivity reviewer. For example, by drawing a reviewer's attention to the positions of sensitivities within a document it will likely reduce the time taken to review the document.

We initially focus on an aspect of FOIA Exemption 27 *International Relations*, namely "information supplied in confidence", since this sensitivity has a clear potential to cause damage to international relations if inadvertently released into the public domain. We show that our approach can markedly improve recall of *in confidence* sensitivities compared to a baseline approach that has been shown to achieve high levels of recall for sensitive text in other domains [11].

The remainder of this paper is as follows. We present related work in Section 2. We present our approach in Section 3, before presenting two classification methods, in Section 4 and Section 5, that implement our approach. In Section 6 we present a baseline approach that we compare our results against. We present our experimental setup in Section 7 and our results in Section 8. Finally, we present our conclusions and future work in Section 9.

2. RELATED WORK

Gollins *et* al. [5] provided an overview of the challenges presented by digital sensitivity review and how these challenges can be addressed by Information Retrieval (IR) techniques. In that work, they noted that although the *type* of

[1]http://www.legislation.gov.uk/ukpga/2000/36/contents
[2]http://www.foia.gov

ICTIR'15, September 27–30, Northampton, MA, USA.

ACM 978-1-4503-3833-2/15/09 ...$15.00.
DOI: http://dx.doi.org/10.1145/2808194.2809496 .

sensitivity can be identified, for example *personal information*, the existence of many sensitivities rely not only on the terms or entities in the document but also on the context of the information, i.e. *what* is said about an entity, *how* it is said and *who* said it.

In [9], we presented an initial approach to address the issue of context dependent sensitivity, highlighted by Gollins *et* al., by deploying a text classification approach with additional features such as the entities in the document, a country *risk* score and a subjective sentences count to identify documents that contained *personal information* and *international relations* sensitivities. Differently from that work, in this paper we investigate automatically identifying sequences of sensitive text, *within* documents, that relates to information supplied in confidence.

Outwith the field of sensitivity review, most work on automatically identifying sensitive text in documents is in the field of document sanitization [1, 4]. Document sanitization tries to automatically *mask* personal information, or information that could reveal the identity of the person the document is about. A popular approach for this is Named Entity Recognition (NER). NER typically identifies *person*, *location* and *organisation* entities in text and NER approaches to document sanitization typically assume that all named entities in a document are sensitive. However, in this work we try to automatically identify information that has been supplied in confidence and, as such, we need a more general solution that can identify sensitivities in what is said about, or by, an entity.

Sánchez *et al.* [11] presented an approach to sensitive text identification that is more general than NER. They assumed that sensitive text is likely to be more specific than non-sensitive text, and used the Information Content (IC) of noun phrases as a measure of how sensitive the phrase is. Sánchez *et al.* focused on identifying *personal information* sensitivities. However, they also identified textual phrases that are potentially *confidential*. Therefore, their work is more closely aligned to identifying FOIA Exemption 27 sensitivities than NER approaches. Moreover, identifying all potential sensitivities in government documents is the first stage of the sensitivity review process and Sánchez *et al.* found that approach achieved higher recall for sensitive information when compared to NER. For these reasons, we use that approach as a baseline for comparing our work against and we describe our implementation of it in Section 6.

3. IDENTIFYING SENSITIVITY LOADED PART-OF-SPEECH N-GRAMS

The approach we present in this paper uses the *sensitivity load* of POS n-grams to identify sequences of sensitive text in documents. The approach is inspired by Lioma and Ounis [8] who showed that the distribution of POS n-grams in a corpus can indicate the amount of information they contain. More specifically, Lioma and Ounis showed that high frequency POS n-grams are typically *content rich* and removing *content poor* POS n-grams from search engine queries results in an improved overall retrieval performance. Differently from their work, we use the content load of POS n-grams to try to measure the sensitivity load of specific sequences of text.

Our intuition is that certain POS sequences might be more frequent in specific sensitivities. For example, sensitivities relating to information supplied in confidence would likely contain variations of the sequence *noun - verb - pronoun*, indicating that someone has supplied information to someone else. Our approach uses the distribution of POS n-grams to try to identify sequences that are specific to this sensitivity.

To do this, we first represent documents by the POS n-grams they contain. For example, the sequence "The envoy will report on Tuesday" results in the POS tags "DT NN MD VB IN NN". Representing this sequence as POS 3-grams results in the following "DTNNMD NNMDVB MDVBIN VBINNN"

Having represented the documents by their POS n-grams, we then use a probabilistic method to measure the sensitivity load of a POS n-gram. More specifically, following the work of Li *et al.* [7], we first construct a 2-way contingency table as shown in Table 1, where pos, s is the number of documents in which the POS n-gram appears in sensitive text, $pos, \neg s$ is the number of documents in which the POS n-gram appears in non-sensitive text, $\neg pos, s$ is the number of documents that do not contain the POS n-gram in sensitive text and $\neg pos, \neg s$ is the number of documents that do not contain the POS n-gram in non-sensitive text.

Table 1: 2-way contingency table used to calculate the Chi-square statistic of a part-of-speech n-gram.

	sensitive	non-sensitive
Containing POS	pos, s	$pos, \neg s$
Not Containing POS	$\neg pos, s$	$\neg pos, \neg s$

Having constructed the contingency table, we use the Chi-Square test of independence to measure the degree of dependency between a POS n-gram and sensitive text. The Chi-square score for a POS n-gram, (X^2_{pos}) is calculated as follows,

$$X^2_{pos} = \frac{N_d(p(pos,s)p(\neg pos,\neg s) - p(pos,\neg s)p(\neg pos,s))^2}{p(pos)p(\neg pos)p(\neg s)p(s)} \quad (1)$$

where $p(pos, s)$ is the probability that the sensitive text of a document contains the POS n-gram, $p(pos, \neg s)$ is the probability that the non-sensitive text of a document contains the POS n-gram, $p(\neg pos, s)$ is the probability that the sensitive text of a document does not contain the POS n-gram, $p(\neg pos, \neg s)$ is the probability that the non-sensitive text of a document does not contain the POS n-gram, $p(pos)$ is the probability that a document contains the POS n-gram, $p(\neg pos)$ is the probability that a document does not contain the POS n-gram, $p(s)$ is the probability that text in the collection is sensitive, $p(\neg s)$ is the probability that text in the collection is not sensitive and N_d is the total number of documents in the collection.

The Chi-square test of independence measures how much the observed frequency of a POS n-gram diverges from its expected frequency within a corpus. If a POS n-gram's Chi-Square score is greater than the Chi-Square distribution's critical value for a 95% confidence level with one degree of freedom, then we assume that the distribution of the POS n-gram in the corpus is related to sensitive text. We refer to these n-grams as being *sensitivity loaded*.

By applying our approach, the identified sensitivity loaded POS n-grams can then be used as features in any method for automatic sensitive-text classification. In Section 4 and Section 5, we present two such methods that integrate our approach.

4. SENSITIVITY LOAD FILTERING

Confidential information in documents is rare. Indeed, 95% of terms in our collection are in fact part of sequences that the reviewers did not believe to be sensitive. Moreover, many documents adhere to template structures in which certain sections of a document are particularly unlikely to contain confidential information, for example the header section of an email stating the *sender*, *recipients* and a *subject title*.

Therefore, we can use the sensitivity loaded POS n-grams identified by our approach, presented in Section 3, to filter out sequences of text that are most likely to contain non-sensitive information.

To do this, we count the number of times a sensitivity loaded POS n-gram appears in sensitive and non-sensitive text and use the ratio of sensitive and non-sensitive occurrences to identify POS n-grams that are representative of non-sensitive sequences. Equation 2 shows how we calculate the ratio, $lRatio_{pos,s}$, where Sensitive correspondes to when $lRatio_{pos,s} > 1$ and Non-Sensitive corresponds to when $lRatio_{pos,s} < 1$.

$$lRatio_{pos,s} = \frac{p(pos,s)p(\neg pos,\neg s) - p(pos,\neg s)p(\neg pos,s)}{p(pos)p(s)} + 1 \quad (2)$$

To classify terms in a given sequence of text k, we represent k by its POS n-grams k_{pos}. Then, for any k_{pos} in the set of previously identified non-sensitive POS n-grams, each term t in k_{pos} is classified as being *Non-sensitive*, all other terms are classified as *sensitive*. We refer to this method as the $Filtering$ method.

5. SENSITIVITY LOAD SEQUENCES

The identified sensitivity loaded POS n-grams can be used to generate features for a Conditional Random Fields (CRF) sequence tagger [6]. A CRF is a probabilistic framework that models the conditional distribution $p(y|x)$ in sequential data, where x is a sequence of observations, an observation is a term with a set of features that describe the term and y is a sequence of class labels.

To deploy the CRF method we use two term features. The first feature we use is the term's POS tag. Additionally, we use the sensitivity loaded POS n-grams identified by our approach to generate a feature *tag* that indicates whether the term is part of a sequence that maps to a sensitivity loaded POS n-gram.

To generate the feature tag, we look at a sequence of n (POS tagged) terms at a time and check if the sequence maps to a sensitivity loaded POS n-gram. If a mapping exists, we tag each term in the sequence. We then move the sliding window by one term to the next sequence and repeat the process.

To illustrate this, we return to our example from Section 3. Recall that the sequence "The envoy will report on Tuesday" is represented by the POS 3-grams "DTNNMD NNMDVB MDVBIN VBINNN". If our approach identifies "NNMDVB"

```
The       DT
envoy     NN  Sensload
will      MD  Sensload
report    VB  Sensload
on        IN
Tuesday   NN
```

Figure 1: Illustration showing the POS tag and generated sensitivity loaded feature tags for a sequence.

as the only sensitivity loaded n-gram, this would result in the sequence being tagged as shown in Figure 1.

When the learned CRF model is deployed, it predicts class labels for each term in a sequence based on its previous observations. We refer to this method as $CRF{+}POS{+}TAG$. We also present the results for the CRF method without the sensitivity loaded feature tag, $CRF{+}POS$, and the simple CRF (i.e. using just the term) as CRF.

6. INFORMATION CONTENT

As previously mentioned in Section 2, we compare our approach against a baseline from the literature that uses the Information Content (IC) of a noun phrase as a measure of the phrase's sensitivity [11]. A noun phrase is a sequence of terms that has a *noun* or *indefinite pronoun* at the *head* of the phrase. IC measures the amount of information provided by the sequence of terms, within the context of a background corpus. More *specific* term sequences are considered as having a higher likelihood of being sensitive.

To calculate the IC of noun phrases in a document, the document is first parsed to extract its syntactic structure. Noun phrases are then extracted from the resulting syntax tree and submitted to a Web search engine as a query. The IC of the noun phrase is calculated using the number of returned results as an indication of the phrase's specificity. Formally, the IC of a noun phrase, np, is computed as $IC_{(np)} = -\log_2 p_{(np)} = -\log_2 \frac{res(np)}{totalpages}$, where $res(np)$ is the number of returned search results and $totalpages$ is the number of sites indexed by the search engine. For our experiments, each term within a noun phrases with an IC score greater than an empirically defined threshold, β, is classified as being sensitive. All other terms are classified as non-sensitive. This baseline is referred to as $InfContent$.

7. EXPERIMENTAL SETUP

Collection: The collection consists of government documents with "information supplied in confidence" sensitivities. The documents have been sensitivity reviewed by trained sensitivity reviewers. Reviewers were asked to annotate the sensitive sequences within the documents, therefore the documents have term-level class labels i.e. each term within a sensitive sequence is labelled *sensitive* and all terms that were not annotated are labelled *non-sensitive*. There are a total of 231893 terms in the collection with 10838 sensitive terms and 221055 non-sensitive terms, in a set of 143 documents. For our experiments, we split the collection at the document level to retain the context of the terms and perform a 5-fold Cross Validation.

Baselines: We compare our approach against the IC baseline presented in Section 6. We use Open NLP[3] to extract noun phrases from documents and for calculating the IC score of noun phrases we use the Bing search engine[4] and set the total number of sites indexed, *totalpages*, to 3.5 billion. For our experiments we test IC threshold values of $\beta = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, 50, 75, 100\}$. In future work, we intend to investigate learning the β parameter.

We also report the results of a simple classifier that classifies all terms as the majority class, referred to as *All Non-Sensitive*. Conversely, we also report a classifier that classifies all terms as sensitive, referred to as *All Sensitive*.

[3]https://opennlp.apache.org
[4]http://www.bing.com/

Sensitivity Load: For identifying sensitivity loaded POS n-grams in Section 3, we use the TreeTagger[5] for POS tagging and, following Lioma and Ounis [8], we use a reduced set of 15 POS tags. We calculate the Chi-Square statistic on the training data for $n = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. In future work, we intend learning the optimum value for n.

Classification: For Sensitivity Load Filtering (Section 4) and Sensitivity Load Sequences (Section 5), we test the methods for each value of $n = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. For the Sensitivity Load Sequences classification method, we use the Mallet[6] linear chain CRF tagger.

Metrics: We report balanced accuracy (BAC) due to the imbalanced nature of our collection. Moreover, sensitivity review is a recall-oriented task since reviewers must identify all sensitivities to avoid inadvertent release. Therefore, we report the F2 measure that provides a weighted average of precision and recall where recall is attributed more importance and, therefore, a greater weight. We also report the standard accuracy, precision and recall metrics.

8. RESULTS

Table 2 shows the best achieved performance for the Filtering method ($Filtering$), the CRF method using the POS tag and sensitivity load as features ($CRF{+}POS{+}TAG$), the CRF method with the POS tag only ($CRF{+}POS$) and the simple CRF (CRF). Table 2 also shows the best performance for the Information Content baseline ($InfContent$) and the *All Sensitive* and *All Non-Sensitive* classifiers scores.

The first thing we note from Table 2 is that the CRF classification metod using the sensitivity loaded POS n-grams identified by our approach outperforms the IC baseline for all metrics. The CRF method using sensitivity loaded 10-grams ($CRF{+}POS{+}TAG_{n=10}$) achieves 0.4573 recall. Importantly, the method also achieves 0.9992 precision and, therefore, achieves a balanced accuracy of 0.7282. On closer inspection of the results we found that the CRF method with this setting identified 99% of sensitive text in 67% of the documents and, therefore, we believe this method would be useful in assisting the sensitivity review process. Moreover, we note that the CRF method without the sensitivity loaded POS n-gram feature ($CRF{+}POS$) correctly identified less than 5% of the sensitive text. This further demonstrates the effectiveness of our approach.

The next point we note from Table 2 is that the Filtering approach also achieves its highest recall (0.9129) when n is set to 10. However, the approach achieves a low precision score (0.0563) resulting in an F2 score of (0.2257). This low level of precision shows that, on this collection, this method is markedly over-predicting sensitivity.

We also see from Table 2 that the CRF and the Filtering methods achieve their best recall scores using POS 10-grams.

Finally, we note that the IC baseline, $InfContent_{ic>7}$, achieved 0.0116 recall and 0.2564 precision. The IC baseline identifies syntacticly complex phrases and specific terms as being sensitive. However, not all complex or specific phrases are confidential and, moreover, not all sensitivities are noun phrases. We also note that the classification methods that use our approach for identifying sensitive sequences of text achieve markedly better recall, and notably better balanced accuracy, than the IC baseline.

[5]http://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger
[6]http://mallet.cs.umass.edu

Table 2: Results for the Filtering and CRF methods. The table also shows the performance of the *All Sensitive*, *All Non-Sensitive* and Information Content baselines.

	accuracy	balanced Acc	precision	recall	F2
all Non-Sensitive	0.9533	0.0000	0.0000	0.0000	0.0000
all Sensitive	0.0467	0.5234	0.0467	1.0000	0.1969
$InfContent_{ic>7}$	0.9457	0.1340	0.2564	0.0116	0.0143
$InfContent_{ic>8}$	0.9457	0.1245	0.2388	0.0103	0.0127
$InfContent_{ic>9}$	0.9457	0.1188	0.2281	0.0094	0.0116
$InfContent_{ic>10}$	0.9458	0.1279	0.2466	0.0093	0.0115
$InfContent_{ic>20}$	0.9459	0.0983	0.1911	0.0055	0.0069
$Filtering_{n=7}$	0.3763	0.3805	0.0579	0.7032	0.2178
$Filtering_{n=8}$	0.3360	0.4007	0.0573	0.7441	0.2191
$Filtering_{n=9}$	0.2616	0.4434	0.0570	0.8298	0.2236
$Filtering_{n=10}$	0.1815	0.4846	0.0563	**0.9129**	0.2257
CRF	0.9226	0.0390	0.0179	0.0283	0.0201
$CRF{+}POS$	0.9221	0.0647	0.0856	0.0446	0.0494
$CRF{+}POS{+}TAG_{n=7}$	0.9600	0.5702	0.8310	0.3094	0.3539
$CRF{+}POS{+}TAG_{n=8}$	0.9637	0.6264	0.8932	0.3595	0.4083
$CRF{+}POS{+}TAG_{n=9}$	0.9657	0.6608	0.9454	0.3762	0.4277
$CRF{+}POS{+}TAG_{n=10}$	**0.9712**	**0.7282**	**0.9992**	0.4573	**0.5129**

9. CONCLUSIONS AND FUTURE WORK

In this work, we proposed an approach for automatically detecting *information supplied in confidence* in government documents. Our approach uses part-of-speech n-grams to measure the *sensitivity load* of sequences of text. On a collection of sensitivity-reviewed government documents, we showed that the sensitivity load of these sequences can be used to accurately classify *in confidence* sensitivities within government documents. In particular, using a CRF sequence tagger with the sensitivity load of POS n-grams as features, this approach achieved over 0.45% recall and 0.99% precision, markedly outperforming a baseline approach from the literature that has been shown to achieve high levels of recall for sensitive text in other domains. As future work, we intend to conduct a user study to quantify the benefits of our approach for assisting in the sensitivity review of digital government documents.

10. ACKNOWLEDGMENTS

The authors would like to thank the sensitivity reviewers for their help in constructing the test collection.

11. REFERENCES

[1] D. Abril, G. Navarro-Arribas, and V. Torra. On the declassification of confidential documents. In *Mod. Dec. for Art. Intel.*. 6820:235–246, 2011.
[2] D. A. R. P. Agency. Darpa, new technologies to support declassification. *Request for Information*, DARPA-SN-10-73, 2010.
[3] A. Allan. Records Review. *UK Government*. 2014.
[4] J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proc. of ACL*, 2005.
[5] T. Gollins, G. McDonald, C. Macdonald, and I. Ounis. On using information retrieval for the selection and sensitivity review of digital public records. In *Proc. of PIR at CIKM*, 2014.
[6] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*, 2001.
[7] Y. Li, C. Luo, and S. Chung. Text clustering with feature selection by using statistical data. In *Trans. on Knowledge and Data Engineering*, 20(5):641–652, 2008.
[8] C. Lioma and I. Ounis. Examining the Content Load of Part-of-Speech Blocks for Information Retrieval. In *Proc. of COLING/ACL*, 2006.
[9] G. McDonald, C. Macdonald, I. Ounis, and T. Gollins. Towards a classifier for digital sensitivity review. In *Proc. of ECIR*, 2014.
[10] T. Mendel. *Freedom of information: a comparative legal survey.* Paris: Unesco, 2008.
[11] D. Sánchez, M. Batet, and A. Viejo. Detecting sensitive information from textual documents: an information-theoretic approach. In *Mod. Dec. for Art. Intel.*, 7647:173-184, 2012.