Privacy Disclosures Detection in Natural-Language Text Through Linguistically-Motivated Artificial Neural Networks

Nuhil Mehdy, Casey Kennington, and Hoda Mehrpouyan(✉)

Boise State University, Boise, ID 83702, USA
{akmnuhilmehdy,caseykennington,hodamehrpouyan}@boisestate.edu

Abstract. An increasing number of people are sharing information through text messages, emails, and social media without proper privacy checks. In many situations, this could lead to serious privacy threats. This paper presents a methodology for providing extra safety precautions without being intrusive to users. We have developed and evaluated a model to help users take control of their shared information by automatically identifying text (i.e., a sentence or a transcribed utterance) that might contain personal or private disclosures. We apply off-the-shelf natural language processing tools to derive linguistic features such as part-of-speech, syntactic dependencies, and entity relations. From these features, we model and train a multichannel convolutional neural network as a classifier to identify short texts that have personal, private disclosures. We show how our model can notify users if a piece of text discloses personal or private information, and evaluate our approach in a binary classification task with 93% accuracy on our own labeled dataset, and 86% on a dataset of ground truth. Unlike document classification tasks in the area of natural language processing, our framework is developed keeping the sentence level context into consideration.

Keywords: Privacy · Security · Natural language processing · Machine learning

1 Introduction

In this era of global communication, individuals often share stories, news, and information with each other. It is not easy for these users to keep track of what information they have shared, whether or not that information was a private disclosure, and to whom they shared that information. While the importance of user centric privacy management systems is being widely studied [22,32,33,38,40], only some of this work are concerned with real-time text analysis and identifying text that contains private information. An important step in constructing an effective privacy management system is to concentrate on identifying and discriminating private information from public information.

J. Li et al. (Eds.): SPNCE 2019, LNICST 284, pp. 152–177, 2019.
https://doi.org/10.1007/978-3-030-21373-2_14

For example, a very common medium of social communication between people is messaging using text; e.g., email, SMS/text messages, chat, social media, etc. While interacting, people sometimes disclose personal and sensitive information, unintentionally. For example a sentence, *Let's meet at the Joe's Coffee Shop tonight at seven* is disclosing someone's meeting place along with the time. Whether or not these disclosures are intentional, it could potentially be an unwanted security threat and cause for alarm–or for harm. This example illustrates a common problem in a multitasking environment where users are simultaneously using in both public and private communication mediums. Our approach serves as an automated privacy check in these kinds of situations, warning individuals regarding risky communications in both private and public contexts. This framework could also be effective while processing large amount of off-line text documents. An example case study could be filtering out all the privacy disclosures from a batch of documents that belongs to a person before it's disposal or archival.

Privacy concerns exist wherever personally identifiable information (e.g., name, address, age) or other sensitive information (e.g., health, finance, mental status) is involved [27]. Therefore, improper disclosure control can be the root cause for many privacy issues and the negative consequences of disclosing information could be immense [9]. A recent data scandal involving Facebook and Cambridge Analytica shows how personally identifiable information of up to 87 million Facebook users influenced voter opinions [20,50].

The requirements for privacy measures to protect sensitive information about organizations or individuals has been researched extensively [6,21,34,48]. One approach to protect the disclosure of private information is to detect them in textual data. However, automating the process of classifying private information prior to their disclosure is challenging [1]. One of the difficulties results from the volume of textual data that would need to be processed, and further the automation process is complicated even more by the number of real-time requirements that need to be analyzed [2,49]. Moreover, it remains a challenge to analyze and dissect the details of private information from the text data due to the ambiguities that arise from natural language [19].

In this paper, we identify a potential approach that brings this challenge within reach: recognizing disclosures in a piece of text, which could be a short phrase (i.e., a sentence) within a longer content (i.e., a paragraph or document). Specifically, we focus on identifying whether or not sentences have disclosures in them. Our approach enriches text data with linguistic features such as part-of-speech tags, syntactic dependency parse information, and entity relation information using off-the-shelf language processing tools. We then use these features to train a Convolutional Neural Network (CNN) to learn a mapping from the features to a binary label: disclosure/non-disclosure. This is a structured approach to train a machine learning model for detecting privacy disclosures and then automating that knowledge to classify certain types of privacy breaches.

The contributions of this paper can be summarized as follows:

- **Sentence level privacy disclosure identification:** While there exists similar techniques for classifying an entire document as private (i.e., confidential) or public, most of these approaches rely only on the existence of the privacy related keywords in a document regardless of their semantics. In this paper, we consider detecting privacy disclosure at a sentence level, which is based on not only the existence of privacy related keywords (i.e. disclosure related entities) but also on the valid grammatical structure of each sentence. This reduces false positive results by verifying the construction of a statement.
- **Disclosure Related Entity recognizer:** A Disclosure-Related Entity Recognizer (DRER) is developed by extending a trainable Named Entity Recognizer (NER) model. The developed DRER is later utilized to prepare a unique labeled dataset as well as to provide tagged entities for learning word embedding (i.e. similarities among disclosure related entities).
- **Case study and performance comparison:** We represent a comparison of the efficiency of different neural network architectures to detect privacy disclosure. Further, the proposed framework was evaluated to other similar datasets for a baseline comparison.

In the following section, we review some related work. In Sect. 3, the methodology along with data collection and pre-processing steps are explained. Later in this section, the neural network model and its architecture is described in detail. In Sect. 4, evaluations of the model are explained and experimental results are given. A test implementation and the usability of the proposed framework is also detailed in this section. Finally, some limitations of the approach and a baseline comparison are discussed in Sect. 5, following the conclusion.

2 Related Work

In this paper we focus on the state of the art research on privacy disclosure, which has been studied across different domains, e.g. financial disclosure [5,30] where economical status such as salary, debt, bank balance etc., could be disclosed. Similarly, sensitive business information of an organization could be sorely disclosed if their loss, profit, or inventory price is shared through their website or employees. Furthermore, location (e.g., home address, meeting point) [4], health information (diagnosis report, health status) [11,15,37,47] are considered sensitive or private information. The rest of this section reviews the techniques and approaches that help in identifying privacy disclosures.

2.1 Information Theory and Global Search

In the context of sensitive information detection, Sánchez et al. utilized information theory along with large corpus of words [45] to automatically detect sensitive information from textual documents regardless of the information context.

This approach determines the sensitivity of terms (e.g., person name, disease name, country name) according to their amount of contributed information in a context (e.g., a document). For example, specific terms (e.g., pancreatic cancer) provide more Information Content (IC) than those more general ones (e.g., disease, America). So, they compute the IC of each term by the inverse of the probability of encountering the term in a corpus (e.g. TFIDF = term frequency inverse document frequency). One advantage of this approach is that the disclosure detection does not depend on a finite set of named entities; however, the technique introduces some weaknesses. For example, the proposed framework removes the stop words from the documents, which could demolish the grammatical validity of the sentences. As a result, it is possible to inaccurately cause the algorithm to fail by providing a document containing sensitive terms in a random and meaningless order. In the proposed approach in this paper, we retain punctuations to determine the structural validation and to reduce the false positive outcomes.

The other group of privacy disclosure techniques are built on rule-based approaches, e.g., [53] for conducting global search. In this technique, personally-identifiable information (e.g. name, address) is first detected using pre-specified patterns and templates. This extends to how addresses are written, how phone number is formatted, etc. One of the weaknesses of this approach is that it only focuses on the recognition and removal of personally-identifiable information regardless of the association of the entities with the subjects. For example, a medical document could contain phone number and address of a hospital which should not be considered as sensitive information because they are considered public information. Our approach takes care of both recognition of entities and association of themselves, before giving a decision on how confident the model is.

2.2 Leveraging Dictionaries

The second category of research, utilizes the linguistic resources such as privacy dictionary to automate the content analysis of privacy related information. A privacy dictionary is used with existing automated content-analysis software such as LIWC [31]. Vasalou et al. proposes a technique that uses such a dictionary of individual words or phrases which are assigned to one or more privacy domains [55]. They showed that the dictionary categories could distinguish differences between documents of privacy discussions and general language by measuring unique linguistic patterns within privacy discussions (e.g., medical records, confidential business documents). Although, they prepared the dictionary by sampling from a rich variety of contexts (e.g., self reported privacy violations, health records, social network sites, children's use of the Internet) their approach relies only on the count of sensitive words in a document. Thus, this model could categorize privacy conditions based on a set of words to different privacy domains, however, it fails to consider the context that these words are used in.

2.3 Machine Learning, Probabilistic and Statistical Models

Detection of privacy leaks has also been well-addressed by statistical techniques such as association rule mining [8]. In such an approach, (Chow et al.) employs a model of inference detection using a customized web based corpus as reference where inferences are based on word co-occurrences. The model is then provided a topic (e.g. HIV - human immunodeficiency virus) and said to identify all the associated keywords. This approach is suitable for identifying privacy related keywords (i.e., health information in this case) by utilizing corpus based association rules, but without contextual concern. For example, if a keyword *gp120 (an envelope glycoprotein)* from the reference collection is fed then the system returns more related sensitive tokens such as *gp120-HIV* and *gp120-Flu* without considering their neighboring words and overall meaning. Again, this makes the system inappropriate for valid and precise identification of privacy disclosure.

Hart et al. (2011) utilize machine learning techniques to classify full documents as either sensitive or non-sensitive information by automatic text classification algorithms. They introduce a novel training strategy called *supplement and adjust* to create an enterprise-level classifier based on support vector machine (SVM) with a linear kernel, stop word elimination, and unigram methodology. The weaknesses of this approach is that it classifies private information only based on a set of keywords. Also, the proposed supervised machine learning models are not trained based on the proper set of labeled dataset (e.g., wikileaks data set were assumed to be private and normal web sites data are assumed to be public information). Thats why no clear visualization is presented about the learned features of these models.

Caliskan et al. (2014), describes a method for detecting private information and collective privacy behavior in a large social network. The authors introduce a novel learning based approach to determine if a given text contains private information by combining topic modeling, named entity recognition, privacy ontology, sentiment analysis, and text normalization [7]. In this approach, all the data are labeled by Amazon Mechanical Turk (AMT) workers and then different machine learning approaches are tested for generic classification of privacy score.

A further combination of linguistics and machine learning techniques are studied to detect Personal Health Information (PHI) disclosure detection [43]. Razavi et al. compiled a list of patterns/keywords which are related to persons' health information which resulted in a list of health information entities. Then by applying Key-word combinatorial web search, and filters on PHIs and Personally Identifiable Information (PII)s the disclosure of health information is detected. Secondly, machine learning layer was implemented to the system to detect and model any possible type of latent semantic PII/PHI patterns in the annotated dataset. In addition, Mao et al. studied privacy leaks on Twitter by automatically detecting vacation plans, tweeting under the influence of alcohol, and revealing medical conditions [35]. For the classifier, they implemented two machine learning algorithms; Naive Bayes and SVM.

Most of the above statistical methods are trained and tested on a relatively larger piece of content (i.e. a paragraph of sentences) and look for togetherness

of keywords in any part of the whole paragraph. A disclosure related entity (e.g., age) might not reveal someone's privacy when standing alone in a sentence, however, it is considered sensitive when it is combined with other entity (e.g., person with age). The proposed approaches in this section also neglect the sentence coherence and ignore grammatical validation.

2.4 Impact Analysis of Privacy Disclosure

Along with the development of disclosure identification systems, some research has studied the impact of disclosure in the society [46,56]. Schrading et al. [46] provide an analysis of domestic abuse discourse using the data collected from the social and news-aggregation website (reddit.com). Before experimenting with the impact, they developed a disclosure identification system in order to discover the semantic and lexical features salient to abusive relationships. They used one single SVM algorithm but fed it different combination of input features for producing more than one models of other variants. The classifiers were designed specially for identifying texts that contain discussion on domestic abuse. Utilizing different combination of n-gram attributes (1-gram, 2-gram, 3-gram) and semantic role attributes (role, predicates), their linear SVM classifier was able to identify 72% to maximum of 92% abusive relationship from text (72% using predicates only, 92% using n-grams). The disclosure (abusive) identification methodology of this research work is an excellent approach for a specific privacy domain but not a perfect fit across varied domains or contexts which has been addressed in our work.

Andalibi et al. investigate sensitive self disclosures on online social media (Instagram) and the responses they attract [3]. For the identification of self disclosures in that specific social media, they worked on both the visual and textual qualitative content analysis and statistical methods. They analyzed people's comments, feedback on posts and also the relationship among the them. The methodology is mostly dependent on hash-tag (#depression) based keywords that people usually include in the description of their posted photos. Thus, this approach also suffers to precisely identify a disclosure event. For example, someone could tag a public photograph with some depression related hash-tags that does not explicitly disclose his own situation. Hence, the limitations we discussed already (e.g. not looking into sentence structure, relying on existence of keywords only, domain dependency etc.) have also been propagated to these works. Although these research work, related to impact analysis of privacy disclosure highly inspire us toward developing our proposed model that can identify meaningful privacy disclosures.

3 Methodology

In this paper, we leverage a multichannel convolutional deep neural network (DNN) to utilize lexical and sentence level features. Our model takes all of the word tokens, part-of-speech tags, and dependency parse tree information of a

sentence as input. First, lexical analysis are done in sentence level. Then, the tokens are transformed to word vectors by learning word embeddings. Later, these features are concatenated to form the final feature vector. Finally, sentence level structure, and privacy related keywords are learned using the convolutional approach.

In this paper, privacy related keywords are defined as disclosure related entities (DREs). These fall into the super set of all possible named entities (NE) but contextually different (i.e., not all Named Entities are Disclosure Related Entities by our definition). We develop a DRE recognizer by extending an off-the-shelf NE recognizer tool to assist the proposed model.

Definition 1 ***(Disclosure Related Entities)****. Let sentence S be a set of words, $S = \{w_1, w_2, \cdots, w_n\}$. A word w_i is considered to be a DRE if it indicates private information such as name of disease, amount of debt, location of meeting, time of outing etc.*

However, dis-joined existence of such entities in any random part of a sentence does not always prove the occurrence of a valid disclosure of private information *(e.g. My son nothing morning no sense makes spoofing not $100 dollars).* A sentence has to carry a reasonable meaning after being constructed by disclosure related entities (DRE) *(e.g. We are planing to leave for Paris on 31st December in early morning).* Moreover, non-machine learning methods seemed to perform well based on rules and reference datasets, but they are not scalable and adaptable when time comes to analyze large amount of data. In order to overcome these challenges, this paper employs a framework which is based on typical convolutional neural network with extended capabilities. It first looks for disclosure related entities in a sentence, retrieves syntactic information, identifies grammatical validation, learns semantic information, and then determines the occurrence of disclosure or non-disclosure of information.

3.1 Data

The proposed framework consists of a neural network model that requires labeled data to learn patterns of disclosure and non-disclosure sentences from text data. Unfortunately, no particular data set with ground truth (i.e., set of sentences labeled as disclosure/non-disclosure) is available so far to work with. Therefore, after collecting textual data we use a state of the art Natural Language Processing (NLP) Toolkit named Spacy [51] to conduct a preliminary labeling (i.e., labeling raw dataset for training) of the dataset as well as to pre-process before feeding into the DNN model. The left section of Fig. 3 demonstrates the usage of the NLP Toolkit for both data labeling and pre-processing; the following subsections describe the process in detail.

3.2 Data Collection

In order to collect data from different domains, we consider online platforms where people post reviews, ask questions, post tweets, and discuss from a first-person perspective. Online forums like medical, psychiatric, and relationship

communities mostly contain private information through users conversations. However, we also wanted to see whether private information is disclosed by an user unintentionally in public forums (e.g. Stackoverflow, Amazon). This is why we introduce domain diversity here to give the model more generalized data. We sampled the same number of user posts from each domain such as medical forums, social sites, food reviews, place and service reviews etc. All of the domains are selected randomly. This is summarized in Table 1. All the posts are written in English language, and each of them are comprised of 4 to 15 sentences. Average sentence length throughout the whole data set is 9 words. As this research requires data that are related to privacy, we carefully avoided any sensitive resource that could have caused privacy violation. Anonymity has also been assured while collecting these data sets from reliable public sources.

Table 1. Summary of data sources.

Source	Amount of posts
Medhelp forum posts [57]	3000
Amazon product reviews [13]	3000
Amazon food reviews [36]	3000
Hotel reviews [12]	3000
Place of interest reviews [18]	3000
Psychiatric forum posts [41]	3000
Twitter posts [42]	3000
Stack overflow questions [17]	3000
Total	**24000**

In each of the above mentioned domains, people shared their views, feedback, or comments in a set of sentences (i.e., a product review, a twitter status, a question regarding health). Thus they expressed their overall opinions about a product, location, situation etc. Our focus is to analyze each piece of content, and evaluate whether or not an individual is disclosing private information through any of the sentences while expressing his pronouncements. Some examples of private disclosures and public information can be found in Table 2.

3.3 Data Labeling

As mentioned above, no ready-made labeled dataset is found for our experiment where various types of sentences are marked as discloser or non-discloser. Both the privacy policy of available data sources and complexity in classification of such textual data, might be the cause. Yet, this is the most important factor from the model's perspective which learns in a supervised fashion. So, our collected dataset is labeled using an algorithm that is built upon the idea of rule-based approach used by [53,55], and obeying following definitions.

Table 2. Example disclosure and non-disclosure sentences

	Text	Is disclosure
1	I have been living in W Boise Avenue for last few months	Yes
2	I got unexpected divorced after 2 years of relationship	Yes
3	1 pound is equivalent to 1.41 dollars	No
4	My company lost $1 million dollar revenue in last quarter	Yes
5	Spending $100 dollars for a lunch in restaurant is too bad	No
6	Our meeting will be at 3 pm in the US Bank building	Yes
7	Yesterday to garbage keywords am nothing Houston more keywords	No
8	I got the Flu	Yes
9	My son nothing morning no sense makes spoofing not $100 dollars	No
10	We are planing to leave for Paris on 31st December in early morning	Yes
11	Houston is a very populated city to live in	No

Definition 2 ***(Disclosure Related Entity Type)***. *Each* $DRET_f$ *is a set of DREs that belong to a type* f*, where* $f \in F = \{$*Person, Location, Money, Health, Date, Time, Interpersonal Relationship, Business Information*$\}$*. Having* D *as an infinite set of all possible DREs then*

$$\forall\, DRE_d \in D \nexists i, j \in F \text{ where } i \neq j, DRE_d \in DRET_i \cap DRET_j$$

By applying an entity and relation extraction tool [51], we implemented the following formal definition of disclosure to classify the dataset:

Definition 3 ***(Disclosure)***. *Let sentence* S *be a set of words,* $S = \{w_1, w_2, \cdots, w_n\}$*.* S *is disclosing if it satisfies the following condition:*

$$\exists\, w_i, w_j \in S \text{ where } i \neq j, w_i \in DRET_{Person} \wedge w_j \in \bigcup_{f \in F} DRET_f$$

In order to label a sentence as disclosure (Definition 3), we examine the sentence. If it contains one or more entities (i.e., mention of a person, place, location, etc., explained below) and if one of those entities is of type *person* then its labeled as disclosure. This is a simple, yet effective rule which allows us to label our data set with disclosure/non-disclosure classes. A more structured guideline for manual labeling is given below:

1. Start with an example sentence
   (a) Look if that contains one or more DRE (by Definition 1) which falls into the set of DRET (by Definition 2).
   (b) If Count of DRE > 1 AND at least one of the DREs is type of PERSON go to Step 2 otherwise label it as a Nondisclosure sentence.
2. Is it a grammatically valid sentence?
   (a) If YES go to Step 3 otherwise label it as a Nondisclosure sentence.
3. Label the sentence as Disclosure and return to step 1.

This produced 5000 disclosure sentences and 5000 non-disclosure sentences from the collected dataset (Table 1), that yields proper labeled information with ground truth. Human evaluation on the labeled examples (i.e. 20% of the data) was also done for the verification of the applied techniques. We use this data to train our model which we hypothesize will generalize to new data, that we show in our evaluation. Although, those 24,000 posts contained more than 100 thousands of sentences, we picked only those with disclosure related entities in it. Hence, the final quantity becomes lower after eliminating most of the sentences with non-disclosure content.

At this stage of our work, we consider the following entity types while discriminating sentences with privacy disclosure: `Person` *(e.g. I, He, Robert)*, `Location` *(e.g. Starbucks, Airport, Main Street)*, `Money` *(e.g. $100, 1 million)*, `Date` *(e.g. Tomorrow, 31st December)*, `Time` *(e.g. 7 pm, Evening)*, `Interpersonal Relationships` *(e.g. Married, Divorced)*, `Health Information` *(e.g. Flu, Pregnant)*, and `Business Information` *(e.g. Revenue, Loss, Profit)*. It's worth mentioning that the types mentioned above are just few from all possible categories that might be related to privacy and security. The number of considerable categories could be extended or reduced as per problem domain.

3.4 Data Pre-processing

As can be seen from the examples in Table 2, many DREs (e.g., I, divorce, 3 pm, $100 dollar, Houston) can be used in both private disclosures and in public posts. This makes the problem particularly challenging because we cannot simply rely on the lexical items in the text; we have to consider the intent of the author of the text, and somehow determine if the intent was for the text to be public (i.e. DRE used in a public statement) or private (i.e. DRE used in personal context). To this end, we do special tokenization and enrich our data with additional information using linguistic details such as part-of-speech tags and syntactic dependency relations. We make use of the NLP toolkit Spacy [51] for all of our data pre-processing. This tool is also used for feature enrichment by creating synthetic features (e.g. dependency tree, POS tags) out of existing features (i.e. word tokens, sentences).

Tokenization. In many text-based natural language processing tasks, the text is pre-processed by removing punctuation and stop words, leaving only the lexical items. However, we found that the way people punctuate their texts helps give the clues as to whether or not it is a valid private or public information. That is, we considered tokens from an example sentence like *Ok... I will meet you; tomorrow morning,, in-front of the Coffee Shop!... :)* are *["Ok", "I", "will", "meet", "you", "tomorrow", "morning", ",", "in", "front", "of", "the", "Coffee", "Shop"]*. Therefore, we use the NLP Toolkit to tokenize the sentences in a customized way that ignores redundant tokens such as *",,", ";", "!", ":)"* but keeps the important ones. This step of considering all the valid sequential tokens helps our model learn important arrangement of tokens for validating relationships of entities. This is somewhat in contrast to other text analysis literature

where clearing off all the punctuation tends to improve task performance. However, keeping the punctuations showed better performance than removing them, throughout our experiment.

Syntactic Structure. Present linguistic theory, classifies certain formal properties of language as "purely stylistic." That is, two sentences can have different forms but express the same meaning [16,44]. For example, a sentence with the structure *subject verb direct-object preposition object* is semantically equivalent to *subject verb object direct-object*, though they are syntactically different. Also, as per our experiments, dependency parse information, and parts of speech tags are two synthetic features that improved the performance of the neural network model. This helps the model to observe common sequence of tokens as well as co-occurrence of dependency tags. We use a Dependency Parser (DP) Toolkit [51] to extract the syntactic relation information (which is different from, but in some ways similar to, entity relation information). This allowed us to enrich our data with dependency parse information.

Parts-of-Speech. Even though we use syntactic structure, we also include parts of speech as a slightly less structured representation of the input text that is also non-lexical. (We found, however, that including Parts-of-Speech did not dramatically increase the performance of our model.)

Figure 1 shows an example of the linguistic feature enrichment for the example sentence *Me and Steve will meet you tonight* for parts-of-speech (which appear below the words) and the dependency parse tree. Figure 2 shows the entities with their tagged entity types.

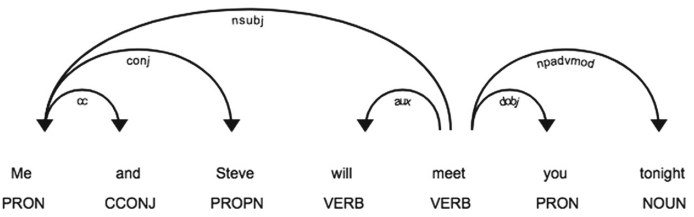

Fig. 1. Parts-of-speech and dependency parse tree of an example sentence.

Fig. 2. Recognized entities in an example sentence.

In summary, our data set is comprised of the original tokenized text enriched with parts-of-speech, tagged entities, and syntactic information.

3.5 Model and Approach

Our model composes together multiple channels of a convolutional neural network to perform the disclosure/non-disclosure classification task, where each channel refers to different representations (i.e., word tokens, dependency parse tree, parts-of-speech tags) of the same candidate piece of text. All the channels use similar hyper parameters (e.g., input/output dimension, activation function, dropout) applied to them to keep computational consistency. Shared input layers are combined together at the first stage of the neural network which is described in this section.

3.6 Neural Network Architecture

The primary task is a supervised optimization problem while minimizing error of classifying disclosure/non-disclosure sentences. An overview of our proposed framework, along with the core model, is represented in Fig. 3. We explain most of the important constituents of the system below.

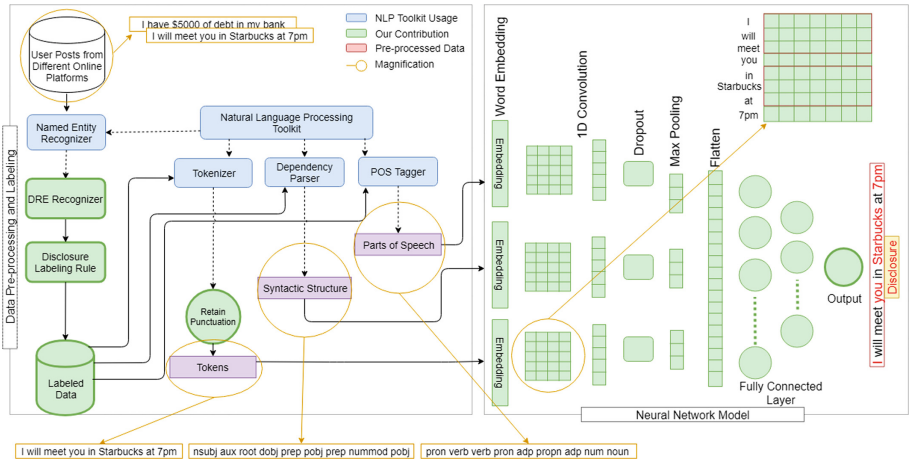

Fig. 3. The bigger picture of the whole framework combining linguistics and neural network stages.

Word Embedding Layer. Word embedding represents words as a dense vector representation in high-dimensional space [10,23,54]. Unlike the typical bag-of-words model, where words are represented as very sparse high-dimensional (e.g. 1-hot) vectors, in word embeddings, words are represented by dense vectors where a vector represents the projection of the word into a continuous vector space. The most important benefit of utilizing word embedding is that the position of a word or token within the vector space is learned from text and is based on the words that surround the word where it is used. This is useful because

words that have similar semantic meanings are close (in terms of Euclidean distance) to each other; which is more semantically useful than one-hot encodings, in which all words are semantically equidistant from each other.

In the proposed neural network architecture, we apply word-embedding as the first layer of the model to learn embeddings through training. Specifically, three separate embedding layers are used as the first hidden layer of each of the multichannel input of the network. We prefer this technique of learning embeddings because we did not observe better accuracy while using pre-trained word vectors like GloVe [39], rather it caused computational overhead. Glove for example, contains 800 billion of tokens which in turns incorporate 800 billion of word vectors. On the other hand, these embedding layers learn semantic relationships from DREs, words, and tags from our data throughout the process. This is particularly crucial, as we apply embeddings not just to words, but also to three types derived linguistic features: parts-of-speech, entities, and dependency parses, as explained in Sect. 3.1). We observed better performance while implementing this approach.

Convolution Layer. CNN is a neural network architecture which is useful in mapping 'togetherness' of information (i.e., image of objects, sentence of tokens) onto class labels. These are feed-forward neural networks that became popular in image processing by work of LeCunn et al. [29]. While traditional CNNs used in image processing are 2D, 1D CNNs can be successfully used for sequence processing [26,28]. This is because, text data (e.g. a sentence of tokens) have a strong 1D (sequential) locality that can be successfully extracted by convolution. LSTM neural network seems a good fit for this task at first place, these networks are more computationally intensive than CNN-based networks. In this work, sequences of tokens in-between entities is observed deeply by utilizing one dimensional convolution with smaller kernel for learning about valid syntactic structure among entities in a way where one or more entities are modifying other entities.

Another challenge that makes the problem of validating sentence structure difficult is that the sequences (i.e., the input sentences and accompanying linguistic features) can vary in length. Sequences could be short as 2–3 words in length or, as long as 8–10 words. Its obvious that the model needs to learn the co-occurrence of tokens or dependencies between symbols in the input sequence. Unlike two-dimensional convolution in an image processing area which focuses on spatial visual structure, a one dimensional convolution suits perfectly in this approach for looking into sentences. As with the word embedding layers, there is a convolution layer for each channel–one for each linguistic feature type.

Following each convolution layer, we introduce a dropout layer, a pooling layer and, a flatten layer before going into the concatenation layer where inputs from different channels are merged.

Concatenation. In this layer, the three channels are brought together. Our final goal is a single, composed neural network that uses the three linguistic feature

types then performs a single binary classification task. Concatenation is the simplest form of bringing these different channels together by simply representing the output layer of the respective CNNs from each channel as a single input into the following layer.

Fully Connected Layer. After concatenation, we apply several densely connected network layers. These hidden layers are comprised of one hundred neurons in the input, then ten neurons in the hidden layer and, finally an output neuron for binary classification at the end. We implemented the well-known Rectified Linear Unit (ReLU) neurons for the first two layers and Sigmoidal neuron in the output layer.[1] Our final resulting model is depicted in Fig. 3 where three separate channels take in three different linguistic feature sequence types, each channel begins with an embedding layer, followed by a CNN layer; those three layers are concatenated, then a three-layer feed-forward network made up of dense layers (using standard ReLU and sigmoid activations) outputs a distribution over a binary class.

In summary, the model is not only learning about the private information but also learning about the correct grammatical structure of such sentences. We train it with words themselves as well as with two other representations (i.e. parts of speech and dependency tree) of the example sentences. This helps the machine learning model to learn both privacy related tokens and pattern of a correct sentence.

4 Experiment

This section and the subsequent portions contain details about the experimental environments and tools, along with implementation of the proposed model in the processed data set and results from an off-line evaluation.

4.1 Data Preprocessing

In the data pre-processing step, we applied Spacy [51] to derive the linguistic features of each sentence. This tool comes with several features to analyze natural language text. Parts of speech tagging, deriving syntactic structure, and tokenization are done by this toolkit. The reasons behind selecting Spacy include - its trainable statistical model (we trained its existing NER model), dependency parser, tokenizer, noun chunk separator in a single toolkit. Two peer-reviewed papers in 2015 confirm that spaCy offers the fastest syntactic parser in the world and that its accuracy is within 1% of the best available. It also contains a statistical entity recognition model in it, but does not have an entity recognizer

[1] It is worth mentioning that we get little fluctuation on the accuracy value while changing the number of neurons in these layers. It seems obvious because, this layer might have needed more neurons for better non-linearity understanding when it sees relatively more data.

for more specific types in which we are interested, such as Interpersonal Relationships, Health Information and, Business Information. The default model identifies a variety of named and numeric entities, including companies, locations, organizations and products, falling somewhat short of identifying some additional entities according to our problem scope.

For example, out of the box, it can not identify *flu* as a disclosure-related entity, whereas it should be identified as a Health Information type entity as a task of the first step toward the whole disclosure recognition system. We were, however, able to leverage Spacy's model extension provisions [51], resulting in an extended entity recognizer model that was trained to identify Interpersonal Relationships, Health Information and, Business Information such as *divorce*, *marriage*, *flu*, *cancer*, *fever*, *loss*, *profit* etc. as valid recognizable entities. An annotator tool by Spacy called Prodigy [52] is used to train the NER model further for identifying these new types of entities. Prodigy has a loop model architecture by which it shows relevant keywords based on the annotation of previous steps.

After this, text encoding is done using Keras [25]. At the end of integer encoding, post padding with zeros are also done for all the sequences or sentences to a certain value which is the maximum length of a sentence in the whole training data set. The post padding is needed to make all the input sequences same length which is required by the later neural network architecture.

4.2 Neural Network Implementation

For implementing the word embeddings we use the *Embedding* layer of Keras [23] that turns positive integers into dense vectors of fixed size [23]. As per its requirement, the integer encoding of all text data is completed on the earlier stages. At the beginning, the embedding layers are initialized with random weights and then learn embeddings for all of the words in the training dataset.

For the *Convolution* layer, we use the Conv1D layer of Keras. To avoid the over-fitting problem of this neural network, we applied 20% dropout rate after each convolution layers using Dropout layer of Keras. This is a common practice which means setting the values of 20% input units to 0 at each update during each iteration of the training life cycle. A pooling layer is also added just after the dropout layer by utilizing *Pooling* followed by a *Flatten* layer of Keras.

The Keras functional API provides some methods to define complex model structure such as multi input and or multi output models that best suits our case. The `concatenate` method of Keras takes all the output vectors from the convolution layers and merges them into a single vector which then acts as the input to the later fully connected layers [24].

4.3 Model Hyper Parameters

This section describes all the needed model hyper parameters and intuition behind the selection of those parameters and associated values. First of all, random seeding is used for maintaining reproducibility while experimenting with

different architectural values. For the `Input` layers that define the shape for each of the three multi channel inputs, is determined by the length of the longest sentence (by tokens).

In each of the three embedding layers all the mandatory parameters are chosen as follows: input dimension is the vocabulary size and, output dimension that describes the size of output vectors where words are embedded is 100 and increased to 200 while working with more than twenty thousand sentences.

Convolution layers are comprised of `32` filters with kernel size of `4`, and `relu` as activation function, keeping all other parameters to default values as determined by Keras. Some default parameters are worth mentioning such as, `valid` (no padding) as padding type, `1` as the strides and dilation rate, `zeros` as bias initializers, with no kernel regularization (regularizers allow to apply penalties on layer parameters during optimization).

Pooling layers are responsible for the max pooling operations on the temporal data which are comprised of `2` as pooling window and, strides for downscaling. This layer uses `valid` as the padding type by default. To prepare the data for concatenation, we flatten all the multi channel inputs separately after the max pooling.

We use ReLU (Rectifier Linear Unit) as the activation function for all the neurons in the dense hidden layers, whereas Sigmoid is used as the activation function in the only neuron of the output layer where we get a probability value towards disclosure or, non-disclosure. The model is trained using 50 epochs, with a batch size of 100.

4.4 Model Summary

A high level summary of the multi channel convolutional neural network goes as follows - each embedding layer produces 100 dimensional word embeddings, and connected to the earlier input layers. Also, each of the convolution layers contains 32 filters with no padding. After the convolution, dropout layers and pooling layers are employed. Later, three separate flatten layers are used. Eventually, a concatenation layer merges all the input vectors to a single one, and forwards to the fully connected layers. Finally, the output layer that contains a single neuron produces the probability score for the desired binary classification.

4.5 Task and Procedure

Our task is a binary classification task of identifying whether a piece of short text contains a personal disclosure or not. We compare our model (as described above) to several other known classification models after the data pre-processing step (i.e., all models had the same inputs). Procedure of applying those models and their outcomes are described below.

Simple Convolutional Neural Network. A simple CNN with only word tokenization is first applied for identifying disclosure and non-disclosure events. This simple network also uses a word embedding layer along with 32 filters with

kernel size of 3 by maintaining same padding for convolution, max pooling of 2, using binary cross entropy as loss function and, ReLU as the activation function. This network serves as our baseline.

LSTM Recurrent Neural Network. We also compare to a recurrent neural network, LSTM, because LSTMs have been shown to produce good results in sequential language processing tasks. We use a word embedding, LST (with 100 neurons), and dropout (20%) layer.

CNN with LSTM Network. We also compare to a combination of the CNN and LSTM models as they are explained above. This allows the model to combine the benefits of the sequential LSTM and filters from the CNN in a single model. The data of this experiment contains one-dimensional spatial structure in the sequence of words in conversational text and the CNN (Convolutional Neural Network) tries to pick out invariant features for disclosure and non-disclosure events. This learned spatial features is then treated as sequences by the subsequent LSTM layer. This combined neural network shows very good improvement in accuracy but going through an obvious computational overhead.

The Multichannel CNN. Eventually, our proposed multichannel convolutional neural network is applied for the classification of disclosure and non-disclosure sentences by providing word tokens in one channel, dependency parse tree to another channel, and parts of speech tags to the third channel. This is the final model we integrate in the proposed framework (after the data simplification stage) because of it's ideal performance. Its worth mentioning that, a multichannel LSTM recurrent neural network was also applied for the classification of the data set Just like the final multichannel CNN. This network also gets different data representations into different channels but could not beat the final model. Even though, LSTM based network seems best suit for learning pattern from sequential data, our convolutional network makes best use of learning togetherness of tokens on the pre-processed data and outperformed all of our other experimental models.

4.6 Metrics

Classification accuracy (Eq. 1), F-Measure, and Receiver Operating Characteristic (ROC) are used as the evaluation metrics. We consider these different types of evaluation metrics because we take it as a binary classification task where accuracy, precision, recall, and diagnostic ability of disclosure identification are equally important. We use labeled data to train our model in a supervised fashion, and evaluation is also based on similarly labeled data-set (actually a split from the original data set by 30%). Remaining 70% of data was used as training and validation set, containing 50% and 20% in each group respectively.

$$Accuracy(ACC) = \frac{\sum Truepositive + \sum Truenegative}{\sum Totalpopulation} \tag{1}$$

For observing the precision and recall of our final model, we consider F-Score as per following equation (Eq. 2). We try to look how precise our model is, while identifying disclosure sentences as well as its capability of pulling out disclosure sentences as much as possible from the test data set.

$$F_1 = \frac{2}{\frac{(TP+FN)}{TP} + \frac{TP+FP}{TP}} = \frac{2TP}{2TP + FP + FN} \tag{2}$$

A ROC curve is used to evaluate the association of true positive rate against the false positive rate to examine the sensitivity, and fall-out of the model. We also calculate the AUC (area under curve) value of the ROC curve.

4.7 Results

For experimenting with different models to achieve a strong classification result, the model variants described above with different architectures are applied in the same data set. Each variant gets the same simplified and entity marked data.

The simple convolutional neural network that uses only word tokenization shows 69.2% accuracy in identifying disclosure and non-disclosure occurrence. Simple LSTM network shows 70.6%, and the combined neural network of convolution and LSTM layers shows 74.1% of accuracy. The multi-channel LSTM neural network model achieved 81% accuracy.

Our proposed model that uses multi-channel inputs and convolution layers along with word embeddings shows 93.72% accuracy on the data set of labeled disclosure and non-disclosure sentences. Also, it shows significant learning improvement on the amount of training data set. Figure 4 shows the comparison of accuracy among all the experimented models along with the final proposed one. Accuracy is measured on the test data that is basically a split of the whole data set and unseen to the model while training.

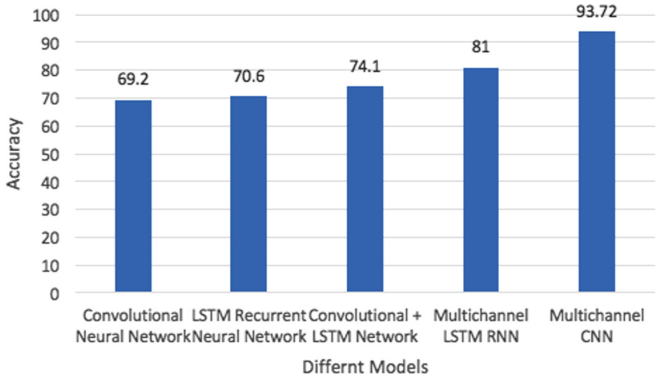

Fig. 4. Comparison among different models

The model shows 0.94 F-Score on disclosure label and, 0.93 on non-disclosure with an overall weighted F-Score of 0.93. Figure 5 shows the ROC curve that is generated as per the predicted labels and, true labels of the test data set. We find significantly large area under the curve which is 0.98 that clearly indicates the strength of the classification model. The ROC curve tells us where we can reliably set the model to disallow false negatives. Its important to know because in this particular task the system should notify users about information disclosure in a lower threshold (i.e., positive if prediction beyond 0.40) to be strict in information leakage.

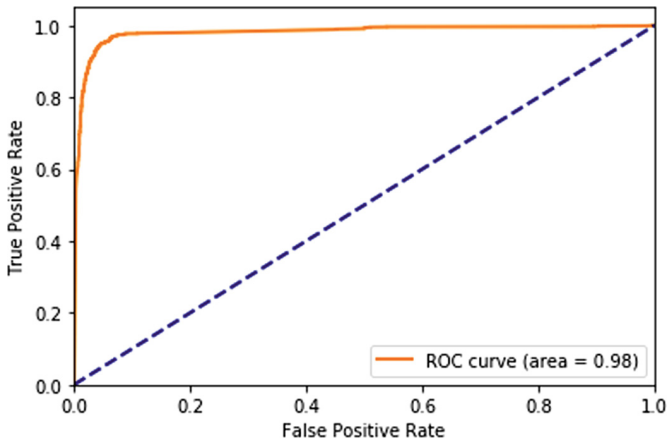

Fig. 5. Receiver operating characteristic curve

These are overall positive results. They show that, despite a lack of large amounts of labeled data, we can train a classifier that goes beyond simple keyword spotting and uses linguistic features to determine if a text contains a disclosure or not with an useful degree of accuracy.

Table 3 shows how we get different accuracy scores on the same data set based on the effect of different input channels.

These results, however, are only applied to learning the automatically labeled data. We further evaluate on 200 manually labeled data (i.e., English sentences which may or may not have the same characteristics required for our labeling rule, as described above) yielded 86.4% accuracy in disclosure identification. This dataset of ground truth was labeled by human who had no idea about the working principle of this model. Those were evaluated from natural perspective of the human agents. This experiment simulates one of the many possible case studies of the developed disclosure identification system.

In order for the proposed framework to be integrated into a global solution for the end users' privacy management problem, a web browser extension is developed to detect privacy disclosures as users are typing their text messages. The implementation is based on a server based request-response architecture.

Table 3. Impact of using multichannel data.

Channel	Accuracy %
Single channel with word tokens	70.6
+ dependency parse tree information as second channel	87.4
+ parts of speech tags as third channel	89.0
Multi-channel input	93.7

The client (i.e. Browser Extension) captures user side text and sends to the server for classification where the trained model is already deployed. If any sentence contains privacy disclosure then the color of that text changes to red, as depicted in Fig. 6. On the other hand, as represented in Fig. 7, the color of the text does not change, since no disclosure is detected.

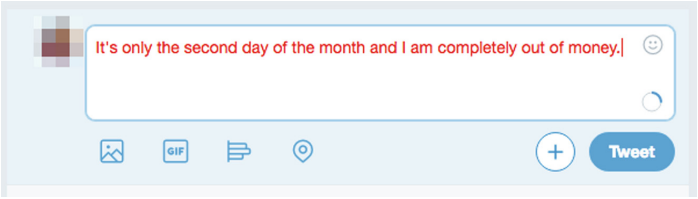

Fig. 6. Information disclosure marked as red automatically by the browser extension. (Color figure online)

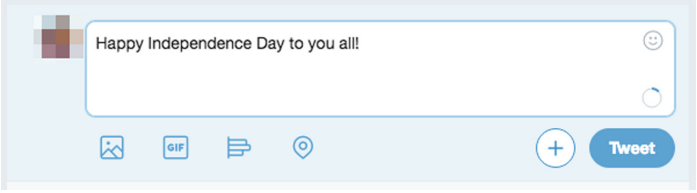

Fig. 7. Non-private information keeps default color. (Color figure online)

This implementation of the proposed framework is one of the many possible use cases. It is also important to note that we recognize the limitation of the developed tool, since sending personal data to a remote server for a classification purposes might result in user's privacy violation. For the future version of this tool we will implement an architecture based on a pre-trained model stored in the client side (e.g. Using TensorflowJS). Source code of this implementation (i.e. the web browser extension and the API server) along with other resources regarding this work are made available for interested researchers[2].

[2] https://anonymous.4open.science/repository/3c84ab7b-02ce-4fd7-b982-f278d6f3c4f4/.

5 Discussion and Analysis

For a baseline evaluation and assessment of the generalizability of the proposed framework, a dataset that was created by Schrading et al. and Choudhury et al. [14,46] is utilized to detect privacy disclosures in Reddit users' posts and comments. This dataset was created mainly to analyze and study the dynamics of domestic abuse in electronic social media (i.e. Reddit). This dataset is comprised of posts and comments from Reddit users under several sub-reddits such as *abuseinterrupted, domesticviolence, survivorsofabuse, casualconversation, advice, anxiety, anger, relationships, and relationship_advice.* All the posts and comments are labeled with one of the above classes.

For the purpose of creating a comparable result, we divided the posts into two classes of Disclosure or Non-disclosure. Submissions under the sub-reddits - *abuseinterrupted, domesticviolence, survivorsofabuse, and relationship* are considered as *Disclosure* class and *casualconversation, and advice* as *Non-disclosure* (Table 4). With this new binary classification, the proposed framework was able to detect each post or comment as a disclosure or non-disclosure with the accuracy of 95%.

Further analysis revealed that even if a post is labeled as an *Abuse*, not all the sentences in the post represent the labeled class and that is the limitation of the work by [14,46]. However, the framework proposed in this paper is able to classify text at a sentence level and provide a more detail analysis. Therefore, in order to be able to compare the result of our classifier with the work of [14,46], we implemented a rule that if at least 70% (i.e. 7 out of 10 sentences of a submission) of the sentences of a post are classified as disclosure, then that entire post is classified as a disclosure. The result of the classifier was assumed correct if that same post was classified as abuse by [14,46].

From all our experiments and the evaluation explained in this section, we have been able to recognize that with considering each sentence as the unit piece of information, the proposed framework faces some limitations while working on conversational context. At the moment, discourse information that spans beyond sentences cannot be handled in the way the system works. For example a chat conversation like - *:How is your son? ...*, *:Bad ...*, *Got flu ...* can mislead the whole system for identifying both the disclosure and the actual nominal subject of this context. Whereas the understandable and rephrased version of the sentence is actually *My son got the flu* and is certainly a disclosure. One possible workaround is to implement that exact same procedures with an extended lookup window. For example, an information extraction step can be implemented in a sliding window style where each window will contain more than one phrases or utterances. Thus, it might be able to find the semantics, and the dependency parse tree of the conversation.

Another limitation of this proposed system is related to incorrectly (i.e., grammatically) written sentences. People often do not care about sentence structure while texting (which is more like speech than standard text) with close friends, and family members. On the other hand, this system moderately depends on sentence structure, specifically structure in-between entities.

Table 4. Summary of the reddit dataset.

Sub-reddit	Class	Quantity	Target class
abuseinterrupted	Abuse	1653	Disclosure
domesticviolence	Abuse	749	Disclosure
survivorsofabuse	Abuse	512	Disclosure
relationship	Relationship	8201	Disclosure
Total	–	11,115	–
casualconversation	Not-abuse	7286	Nondisclosure
advice	Not-abuse	5913	Nondisclosure
Total	–	13,199	–

6 Conclusion and Future Work

A practical model of privacy protection is in dire need by users in the era of social networks that results in activities such as posting online, chatting, text messaging, blogging, and playing online games, etc. Therefore, the development of algorithm and tools that helps users to identifying privacy disclosure in textual data is important. While many research studies in this area mainly focus on classifying textual data as public or private at the document or paragraph level, only few of those are concerned with the privacy detection at the sentence level analysis. Hence, these approaches can not be used for managing privacy for users and they are mostly designed for privacy protection of organization and corporations.

To address this limitation, this paper proposes a privacy disclosure identification framework, comprised of neural network model with linguistics. The proposed framework is capable of: (I) detecting disclosure related entities more effectively by utilizing natural language processing techniques rather than relying on random keywords from an unbound set of tokens, (II) conducting disclosure detection analysis only on sentences with correct subject-verb agreement to increase performance time.

For the proof of concept, we conducted several experiments, examining various machine learning based algorithms Fig. 4 with the different types of data preprocessing techniques, and parameter tunning approaches, while experimenting with various neural network architectures. Throughout this process it was proven that the entity based evaluation, and enriching the input data with additional underlying features helped improving the performance of the model. Convolution over the feature vectors resulted in learning about the sentence structure as well as to overcome the computational overhead.

The future work will concentrate on extending the number of Disclosure Related Entity Types (DRET) to improve the disclosure detection process. Further, the proposed framework will be made more intelligent to be able to infer from the text analysis the interpersonal relationship (i.e., relationship among

friends, family members, colleagues, and public), the context in which the disclosure occurs, and the timing of disclosure to provide an effective privacy management tools an algorithms for users. In order to achieve this objective, an inter-annotator agreement measures and annotation guidelines will be used to ensure consistent annotations, while developing a generalized dataset that will include human annotation through crowdsourcing.

Acknowledgments. The authors would like to thank National Science Foundation for its support through the Computer and Information Science and Engineering (CISE) program and Research Initiation Initiative(CRII) grant number 1657774 of the Secure and Trustworthy Cyberspace (SaTC) program: A System for Privacy Management in Ubiquitous Environments.

References

1. Abril, D., Navarro-Arribas, G., Torra, V.: On the declassification of confidential documents. In: Torra, V., Narakawa, Y., Yin, J., Long, J. (eds.) MDAI 2011. LNCS (LNAI), vol. 6820, pp. 235–246. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22589-5_22
2. Agerri, R., Artola, X., Beloki, Z., Rigau, G., Soroa, A.: Big data for natural language processing: a streaming approach. Knowl. Based Syst. **79**, 36–42 (2015)
3. Andalibi, N., Öztürk, P., Forte, A.: Sensitive self-disclosures, responses, and social support on Instagram: the case of #depression. In: CSCW, pp. 1485–1500 (2017)
4. Bettini, C., Wang, X.S., Jajodia, S.: Protecting privacy against location-based personal identification. In: Jonker, W., Petković, M. (eds.) SDM 2005. LNCS, vol. 3674, pp. 185–199. Springer, Heidelberg (2005). https://doi.org/10.1007/11552338_13
5. Boyd, V.: Financial privacy in the United States and the European union: a path to transatlantic regulatory harmonization. Berkeley J. Int'l L. **24**, 939 (2006)
6. Buchanan, T., Paine, C., Joinson, A.N., Reips, U.D.: Development of measures of online privacy concern and protection for use on the internet. J. Assoc. Inf. Sci. Technol. **58**(2), 157–165 (2007)
7. Caliskan Islam, A., Walsh, J., Greenstadt, R.: Privacy detective: detecting private information and collective privacy behavior in a large social network. In: Proceedings of the 13th Workshop on Privacy in the Electronic Society, pp. 35–46. ACM (2014)
8. Chow, R., Golle, P., Staddon, J.: Detecting privacy leaks using corpus-based association rules. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 893–901. ACM (2008)
9. Christofides, E., Muise, A., Desmarais, S.: Information disclosure and control on facebook: are they two sides of the same coin or two different processes? Cyberpsychol. Behav. **12**(3), 341–345 (2009)
10. Word Embedding Wikipedia Contributors: Word embedding — Wikipedia, the free Encyclopedia (2018). https://en.wikipedia.org/w/index.php?title=Word_embedding&oldid=836044700. Accessed 7 May 2018
11. Costello, J.: Nursing older dying patients: findings from an ethnographic study of death and dying in elderly care wards. J. Adv. Nurs. **35**(1), 59–68 (2001)
12. Datafiniti: Hotel reviews — Kaggle (2018). https://www.kaggle.com/datafiniti/hotel-reviews. Accessed 01 May 2018

13. Dave, K., Lawrence, S., Pennock, D.M.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of the 12th International Conference on World Wide Web, pp. 519–528. ACM (2003)
14. De Choudhury, M., De, S.: Mental health discourse on reddit: self-disclosure, social support, and anonymity. In: ICWSM (2014)
15. DeCew, J.W.: The priority of privacy for medical information. Soc. Philos. Policy **17**(2), 213–234 (2000)
16. Evans, D.A., Zhai, C.: Noun-phrase analysis in unrestricted text for information retrieval. In: Proceedings of the 34th Annual Meeting on Association for Computational Linguistics, pp. 17–24. Association for Computational Linguistics (1996)
17. Stack Exchange: Stack exchange data dump. Stack Exchange, Inc.: Free Download, Borrow, and Streaming: Internet Archive (2018). https://archive.org/details/stackexchange. Accessed 01 May 2018
18. Ganesan, K., Zhai, C.: Opinion-based entity ranking. Inf. Retrieval **15**(2), 116–150 (2012)
19. Groves, T.: Why is analyzing text so hard? (2018). http://www.ibmbigdatahub.com/blog/why-analyzing-text-so-hard. Accessed 01 Feb 2018
20. Hern, A.: Far more than 87m Facebook users had data compromised, MPs told (2018). https://www.theguardian.com/uk-news/2018/apr/17/facebook-users-data-compromised-far-more-than-87m-mps-told/-cambridge-analytica. Accessed 01 May 2018
21. Joinson, A.N., Reips, U.D., Buchanan, T., Schofield, C.B.P.: Privacy, trust, and self-disclosure online. Hum. Comput. Interact. **25**(1), 1–24 (2010)
22. Joshaghani, R., Mehrpouyan, H.: A model-checking approach for enforcing purpose-based privacy policies. In: IEEE Symposium on Privacy-Aware Computing (PAC), pp. 178–179. IEEE (2017)
23. Keras: Embedding layers - Keras documentation (2018). https://keras.io/layers/embeddings/. Accessed 01 Feb 2018
24. Keras: Guide to the functional API - Keras documentation (2018). https://keras.io/getting-started/functional-api-guide/. Accessed 01 Feb 2018
25. Keras: Text preprocessing - Keras documentation (2018). https://keras.io/preprocessing/text/#tokenizer. Accessed 01 Feb 2018
26. Kravchik, M., Shabtai, A.: Anomaly detection; industrial control systems; convolutional neural networks. arXiv preprint arXiv:1806.08110 (2018)
27. Krishnamurthy, B., Wills, C.E.: On the leakage of personally identifiable information via online social networks. In: Proceedings of the 2nd ACM Workshop on Online Social Networks, pp. 7–12. ACM (2009)
28. LeCun, Y., Bengio, Y., et al.: Convolutional networks for images, speech, and time series. Handb. Brain Theor. Neural Netw. **3361**(10), 1995 (1995)
29. LeCun, Y., et al.: Handwritten digit recognition with a back-propagation network. In: Advances in Neural Information Processing Systems, pp. 396–404 (1990)
30. Leyshon, A., Signoretta, P., Knights, D., Alferoff, C., Burton, D.: Walking with moneylenders: the ecology of the UK home-collected credit industry. Urban Stud. **43**(1), 161–186 (2006)
31. LIWC: Linguistic inquiry and word count (2018). https://liwc.wpengine.com/. Accessed 01 February 2018
32. Madden, M.: Privacy management on social media sites. In: Pew Internet Report, pp. 1–20 (2012)
33. Madden, M., et al.: Teens, social media, and privacy. Pew Res. Center **21**, 2–86 (2013)

34. Malhotra, N.K., Kim, S.S., Agarwal, J.: Internet Users' Information Privacy Concerns (IUIPC): the construct, the scale, and a causal model. Inf. Syst. Res. **15**(4), 336–355 (2004)
35. Mao, H., Shuai, X., Kapadia, A.: Loose tweets: an analysis of privacy leaks on twitter. In: Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society, pp. 1–12. ACM (2011)
36. McAuley, J.J., Leskovec, J.: From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 897–908. ACM (2013)
37. Meerabeau, L.: The management of embarrassment and sexuality in health care. J. Adv. Nurs. **29**(6), 1507–1513 (1999)
38. Mehrpouyan, H., Azpiazu, I.M., Pera, M.S.: Measuring personality for automatic elicitation of privacy preferences. In: IEEE Symposium on Privacy-Aware Computing (PAC), vol. 00, pp. 84–95, August 2017. https://doi.org/10.1109/PAC.2017.15, doi.ieeecomputersociety.org/10.1109/PAC.2017.15
39. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
40. Milberg, S.J., Burke, S.J., Smith, H.J., Kallman, E.A.: Values, personal information privacy, and regulatory approaches. Commun. ACM **38**(12), 65–74 (1995)
41. Milne, D.N., Pink, G., Hachey, B., Calvo, R.A.: CLPsych 2016 shared task: triaging content in online peer-support forums. In: Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology, pp. 118–127 (2016)
42. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: LREc, vol. 10 (2010)
43. Razavi, A.H., Ghazinour, K.: Personal health information detection in unstructured web documents. In: IEEE 26th International Symposium on Computer-Based Medical Systems (CBMS), pp. 155–160. IEEE (2013)
44. Sachs, J.S.: Recopition memory for syntactic and semantic aspects of connected discourse. Percept. Psychophys. **2**(9), 437–442 (1967)
45. Sánchez, D., Batet, M., Viejo, A.: Detecting sensitive information from textual documents: an information-theoretic approach. In: Torra, V., Narukawa, Y., López, B., Villaret, M. (eds.) MDAI 2012. LNCS (LNAI), vol. 7647, pp. 173–184. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34620-0_17
46. Schrading, N., Alm, C.O., Ptucha, R., Homan, C.: An analysis of domestic abuse discourse on reddit. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 2577–2583 (2015)
47. Serenko, N., Fan, L.: Patients' perceptions of privacy and their outcomes in healthcare. Int. J. Behav. Healthc. Res. **4**(2), 101–122 (2013)
48. Siegel, A.: In pursuit of privacy: laws, ethics, and the rise of technology. Wilson Q. **21**(4), 100 (1997)
49. Singh, J., Nene, M.J.: A survey on machine learning techniques for intrusion detection systems. Int. J. Adv. Res. Comput. Commun. Eng. **2**(11), 4349–4355 (2013)
50. Solon, O.: Facebook says Cambridge Analytica may have gained 37m more users' data (2018). https://www.theguardian.com/technology/2018/apr/04/facebook-cambridge-analytica-user-data-latest-more-than-thought. Accessed 01 May 2018
51. Spacy: Linguistic features (2018). https://spacy.io/usage/linguistic-features. Accessed 01 Feb 2018
52. Spacy: Named entity recognition (2018). https://prodi.gy/features/named-entity-recognition. Accessed 01 Feb 2018

53. Sweeney, L.: Replacing personally-identifying information in medical records, the scrub system. In: Proceedings of the AMIA Annual Fall Symposium, p. 333. American Medical Informatics Association (1996)
54. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. J. Artif. Intell. Res. **37**, 141–188 (2010)
55. Vasalou, A., Gill, A.J., Mazanderani, F., Papoutsi, C., Joinson, A.: Privacy dictionary: a new resource for the automated content analysis of privacy. J. Assoc. Inf. Sci. Technol. **62**(11), 2095–2105 (2011)
56. Wang, Y.C., Burke, M., Kraut, R.: Modeling self-disclosure in social networking sites. In: Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing, pp. 74–85. ACM (2016)
57. Yang, C.C., Tang, X.: Estimating user influence in the MedHelp social network. IEEE Intell. Syst. **27**(5), 44–50 (2012)