Privacy preserving sentiment analysis on multiple edge data streams with Apache NiFi

Abhinay Pandya
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
abhinay.pandya@oulu.fi

Panos Kostakos
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
panos.kostakos@oulu.fi

Hassan Mehmood
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
hassan.mehmood @oulu.fi

Marta Cortes
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
marta.cortes@oulu.fi

Ekaterina Gilman
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
ekaterina.gilman@oulu.fi

Mourad Oussalah
Center for Machine Vision and Signal Analysi
University of Oulu
Oulu, Finland
mourad.oussalah@oulu.fi

Susanna Pirttikangas
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
susanna.pirttikangas@oulu.fi

***Abstract*—Sentiment analysis, also known as opinion mining, plays a big role in both private and public sector Business Intelligence (BI); it attempts to improve public and customer experience. Nevertheless, de-identified sentiment scores from public social media posts can compromise individual privacy due to their vulnerability to record linkage attacks. Established privacy-preserving methods like *k-anonymity, l-diversity* and *t-closeness* are offline models exclusively designed for data at rest. Recently, a number of online anonymization algorithms (CASTLE, SKY, SWAF) have been proposed to complement the functional requirements of streaming applications, but without open-source implementation. In this paper, we present a reusable Apache NiFi dataflow that buffers tweets from multiple edge devices and performs anonymized sentiment analysis in real-time, using randomization. The solution can be easily adapted to suit different scenarios, enabling researchers to deploy custom anonymization algorithms.**

Keywords—Apache NiFi, social media, anonymization, IoT privacy, sentiment analysis.

I. Introduction

Online social networks (OSNs) allow users to share their opinions, thoughts, and sentiments, and this wealth of information has attracted the attention of data analysts supporting applications in business, politics, healthcare, and security surveillance. The bulk of current research on social media has traditionally focused on microblogging platforms where social interactions, attitudes, and emotions can be freely monitored and evaluated. However, these platforms offer only coarse controls for users to manage their data. As a result, despite the public nature of users' posts, there is a potential risk that information collected for an intended purpose may be used to achieve other, potentially malevolent goals and that users could be targeted for persecution for their beliefs and opinions. Evidently, despite the data being public in nature, many privacy experts have resigned over privacy and security [1,2,3].

European Commission grants 770469 and 815362.

While the content of a text message can be analyzed to reveal information such as the age, gender, and political orientation of individuals [4-6] or the general mood of people [7,8], image analysis may uncover even more specific information such as the place where a photo was taken [9]. Network structure can also be exploited to re-identify users even if all other data is anonymized [10]. It is possible that the behavioral signature of a user can be extracted, even in the absence of the content of the users' messages. For example, Twitter API allows users to access a wealth of metadata such as the number of times a message was re-tweeted, liked, replied-to; date of creation of account; or the number of friends, followers, etc. This metadata alone can be used to uniquely identify the users from a massive dataset [11]. Data owners who do not wish to make their dataset open to the public have various anonymization models at their disposals (k-anonymity, l-diversity and t-closeness). Furthermore, with the advent of new technological spectrums like IoT, 5G, edge computing, anonymization of social data has become a crucial issue. Especially, in the case of streaming data, there is an increased need for real-time anonymization.

Although privacy-preservation in data publishing has attracted a lot of attention and several advancements such as k-anonymity, l-diversity, and other algorithms have been proposed, most existing work focus on static data tables and not easily extendable to the streaming data. Recently, a few models have been proposed for streaming data anonymization: SKY [12], CASTLE [13], SWAF [14]. Yet this technology currently lacks open-source implementation and cannot be easily adopted by data scientists who do not have the acumen to develop the infrastructure required for their implementation.

In this paper, we tackle the problem of individual and group privacy by anonymizing streaming data such that the anonymized data still can be used for extracting useful insights. We develop an experimental setup for anonymization of streaming data using the Apache NiFi platform for real-time data

ingestion and processing in a local network of Raspberry Pi's [15]. The rest of the paper is organized as follows: Section II outlines the closely related work; Section III describes our approach for streaming data anonymization; Section IV details our experimental set up; and Section V presents our conclusions and show near-future work.

II. Related work

Identification risk in publicly released data has been an active area of research for nearly four decades [16-18]. A number of anonymization methods exist to conceal the identifying attributes, allowing only quasi-identifiers (QI) to be available to an adversary. Examples of identifying attributes are passport number or social security number. Other attributes like zip code, birthdate, or job title are QI. While each of these QIs alone cannot identify an individual, a combination of these has a potential to make identification a risk. Furthermore, QIs are also vulnerable to be linked with other datasets that have been already acquired to gain sensitive information. Therefore, most approaches have focused on anonymizing Qis, using several anonymity operations like generalization, suppression, anatomization, permutation, and perturbation.

The k-anonymity [16] property requires that each record in a dataset cannot be distinguished from at least another (k-1) record with respect to the values of quasi-identifiers of dataset after a series of anonymity operations. K-anonymity has been extensively studied in recent years [17]. However, it is still susceptible to linkage attack. L-diversity proposes an effective solution for the attribute linkage attack by reducing the correlation of QI and sensitive attributes [18].

The most common implementations of k-anonymity are achieved with the application of generalization and suppression operations. Generalization is replacing the targeted value of quasi-identifiers with a more general value. Suppression uses special symbols to replace QI value (e.g. *, &, #), and makes the value meaningless. Other operations like anatomization and permutation decrease the correlation between attributes without modifying the original dataset. The goal of perturbation, on the other hand, is to substitute an original value for synthetic data, ensuring the statistical characteristic of the original dataset is unchanged. And due to this property, perturbation (or randomization) is most suitable for accurate data analytics on an anonymized dataset.

In generalization, the goal is to replace the values of the identifying attributes by category indices and in randomization, by adding noise. For example,

Generalization of Age: 21 => [20 – 24]

Randomization of Age: 21 => 23

However, methods available for streaming data anonymization [12-14] use generalization, which is not useful for accurate sentiment analysis. In this paper, we propose a method that buffers the data in batches and then anonymize it by randomization using a kernel density estimator.

III. Our approach

In order to overcome the identification risk posed by publicly streaming data, an online real-time anonymization is developed. Consider, as an example, social media message streams arriving at a server for analysis. Even if the source of each stream removes the identifying attributes and the content of the posts, the remaining metadata and additional domain knowledge (e.g., that the streams contain messages from a particular location) may be sufficient to breach the privacy of the users.

In a more realistic scenario, consider the application that builds the public sentiment profile for a global event by analyzing social media messages from designated locations. It is important for the application to know the overall sentiment from each of these locations, but to hide the individuals' shared sentiments about the event. The data collected by the system requires anonymization such that while the anonymized data allows for aggregate evaluations (e.g., sentiment scores), it disallows identification / tracking of individual users.

Essentially, the randomization method works by reconstructing the probability density function using a set of given data points. Two methods for density estimation are tested: histogram [19] and kernel density estimation [20]. Histograms are a discrete representation of numeric data in equal sized bins. Algorithms to create histograms require: width of the bins, and starting and end points of the bins. Because of this, histograms are not smooth and are heavily dependent on the size of the bins and bin boundaries. We overcome these problems by using kernel density estimators (KDEs). Unlike density estimation using histograms, KDEs are non-parametric density estimators.

More formally, kernel density estimation (KDE) has the effect of smoothing out each data point into a resultant smooth region, the shape of which is determined by the kernel function $K(x)$. Then, KDE sums over all these regions to obtain a density estimator. Let $X_1, \ldots, X_n \in R$be an independent, identically distributed univariate random sample from an unknown distribution F with a density function p, then KDE can be expressed as

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x-x(i)}{h}\right) \quad (1)$$

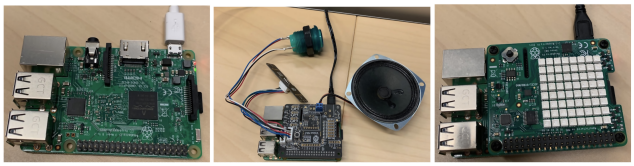

Fig 1. Hardware at the edge of the network using MiNiFi instatnces.

where K is the kernel function and $h > 0$ is the smoothing bandwidth that controls the amount of smoothing. Also, the condition $\int K(t)dt = 1$ is imposed to ensure that the estimates $f(x)$ integrates to 1. The kernel function K is non-negative smooth function with a peak at value 0. Most common choice for a multivariate kernel function is Gaussian. In order to apply KDE to estimate the underlying probability density and to use it to generate random samples according to it, we here need a multivariate Kernel density estimator. Furthermore, in order for the data to remain useful for analytics even after randomization, we need to estimate a conditional multivariate density function and use it to sample the anonymized dataset.

Specifically, let us consider variables x, y, z and w. Let w be the independent variable and x, y, and z be dependent variables. Say w is location value and x, y, z are demographic variables. Then, in order to anonymize this dataset comprising of these four variables, we need to randomize the values such that the statistical distributions remains identical before and after randomization. This can be achieved by estimating a conditional distribution

$$p(x, y, z|w) = p(x|w)p(y|w, x)p(z|w, x, y) \quad (2)$$

Here, p is the probability density function. However, in the multivariate case, since it involves correlated random variables of non-normal distribution, we need to use forward Rosenblatt transformation,

$$R(x, y, z, w) = [F(w), F(x|w), F(y|x, w), F(z|x, y, w)] \quad (3)$$

where each F is a cumulative distribution function. Next, from the available input data, we use the multivariate normal kernel to estimate variables $\mathrm{R}(x, y, z, w)$, using the maximum-likelihood bandwidth selection. We then sample this distribution to generate the anonymized dataset. The method was implemented in Python3 using scipy.stats, statsmodel.api, and chaospy libraries [20].

IV. Experiments and results

The experiment aims to simulate a real-life scenario in which a business deploys smart speakers in different reception areas. Each device is able to capture interactions between customer service and customer and determine the overall sentiment score of the customer using speech recognition software. The challenge is to push actionable sentiment analytics to a cloud, enabling Business Intelligence (BI) operations, while making sure that data on the edge device are anonymized and the identity of the customer and employee protected.

To make the scenario more operational, we decided to collect public tweets from three different locations in the UK, using hashtags that reflect users' political affiliation. The intuition is that an adversary is able to compromise individual and group privacy by tracking support or condemnation of Brexit. This is a realistic scenario applicable also to smart city platforms where a hacker gets hold of sentiment scores and tracks users who are easily reidentified via a linking attack.

A. System Architeture

The system architecture resembles an IoT setup in which three smart devices capture textual data and perform localized sentiment analysis and anonymization in real time before pushing data to a BI platform. Our experimental setup consists of one regional node and three edge nodes communicating over HTTP on a local area network. The regional node, running NiFi server and Elasticsearch, receives data from three edges, each one running MiNifi instances. A Mac mini is used to host the regional NiFi and Elasticsearch servers. Each MiNiFi dataflow collects streaming tweets using the Filter Endpoint from three regions in the UK and pushes the data over HTTP to the regional NiFi installation where data is stored in Elasticsearch.

B. Apache NiFi Dataflow

Using NiFi processors, we were able to fetch live streams from the source system (Twitter) into IoT devices deployed in three different locations in our lab (Figure 1). Particularly, we deployed the MiNiFi dataflows for nine hours and collected 1,758 Brexit-related tweets from 662 unique locations within three predefined bounding boxes of coordinates in the UK. Four classes of sentiments (positive, negative, neutral, and compound) were calculated using the NLTK VADER algorithm [19]. The compound sentiment scores computed in the three edge devices are as follows: 578 tweets with negative polarity, 700 tweets positive polarity, and for 480 tweets the compound polarity summed up to zero, all of which indicate the diversity of opinions captured in our sample.

We used the GetTwitter processor to fetch streaming tweets based on location and keyword criteria. Each incoming *flowfile* (tweet) is first filtered and run through the VADER sentiment intensity analyzer and anonymization modules before porting the anonymized scores and locations of the tweet into the regional node for ingestion to the Elasticsearch. VADER is a lexicon and rule-based sentiment analysis specifically designed for social media texts. Each word in their lexicon has been assigned a sentiment value indicative of its intensity of positive or negative polarity. The scores 'pos', 'neg', and 'neu' are computed as a normalized sum of the values of words present in the text.

A list of keywords was selected to capture tweets related to Brexit. The three locations (bounding boxes) were selected based on the 2016 UK European Union membership referendum to reflect a broader spectrum of opinions between the supporters of the leave and the remain camps. A separate GetTwitter processor was used to fetch tweets from Sample Endpoint that had been deleted by the user. Tweets which had been removed or deleted by users were then removed from the Elasticsearch using "delete_by_query" POST request served over with a NiFi ExecuteStreamCommand. The algorithmic representation of the MiNifi dataflow that runs locally in each device along with the reusable XML template is available here [19].

C. Anonymization

Because of a lack of space, in Table I, we show only a sample of the original data ported into the regional node from various edge nodes. As described earlier, edge nodes calculate the sentiment scores of each tweet, removing any identifying attributes of the authors of these tweets. However, despite these omissions, a re-identification attack can still put the individual at the risk of being tracked because sentiment scores can be reproduced by an attacker with background knowledge about the system. We overcome this problem by proposing to randomize the values of the sentiment scores using our multivariate conditional kernel density estimator. The results for the chosen sample are shown in Table II.

In order to quantify the information loss, we measured the mean change in sentiment values per location after anonymization. In order to compute this, we use conditional cross entropy (conditional on Location attribute) of the sentiment scores. Measuring unconditional entropy (or KL divergence) of the multivariate data before and after anonymization is unsuitable since we want the conditional

distributions preserved. This is required for the target analytics application, say, summarizing overall sentiment about an event per location value. We calculate the information loss before and after anonymization H, conditioned upon the location variable as follows:

$$H(P(x|L), Q(x|L)) = -\sum_{x \in X}^{N} p(x|L) \log q(x|L) \quad (4)$$

where $P(x|L)$and $Q(x|L)$are conditional distributions of variable X given location L. Variable X is positive, negative, neutral, or compound sentiment score of a tweet and N are the total number of tweets.

Table III shows the mean squared-error and conditional cross entropy measurement averaged across all locations for (a) naïve method and (b) our method. It is evident from the table that both average conditional cross entropy loss and MSE are less when using our method of anonymization compared to the popular histogram method. This suggests that our method does not incur much information loss, even after randomization of the data, rendering it useful for accurate data analytics aiming for more reliable results.

TABLE I. DATA COLLECTED AT A REGIONAL SERVER

ID	Timestamp	Location	Com	Neg	Neu	Pos
3f6	1561391573734	Bel/hton	0.4019	0.0	0.526	0.474
0b3	1561391588687	Cheadle	-0.625	0.339	0.661	0.0
78c	1561391636395	Bilbao	0.0	0.0	1.0	0.0

TABLE II. ANONYMIZED DATA FROM TABLE I

ID	Timestamp	Location	Com	Neg	Neu	Pos
3f6	1561391573734	Siran	0.492	0.399	0.003	0.609
0b3	1561391588687	Cove Bay	-0.033	0.0	0.01	0.99
78c	1561391636395	Richmond	0.441	0.0	-0.01	0.89

TABLE III. COMPARISON OF OUR METHOD WITH BASELINE (CCE = CONDITIONAL CROSS ENTROPY, MSE = MEAN SQUARED ERROR)

Variable	Avg CCE of variable	MSE and Avg CCE using Histogram based multivariate density estimation		MSE and Avg CCE using our approach	
		MSE	CCE	MSE	CCE
Compou	2.324	1.21	5.969	0.07	2.718
Positive	3.243	1.34	3.506	0.09	2.763
Negative	2.102	0.91	2.848	0.01	1.960
Neutral	0.976	1.98	2.867	0.11	1.114

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a solution that strengthens individual and group privacy in streaming data analytics using the Apache NiFi platform. Our multivariate kernel density estimator results in low average information loss when compared with the baseline, rendering it useful for accurate data analytics aiming for reliable results. While our experimental set up is currently for Twitter, our system architecture potentially support other applications such as voice data, video surveillance, and others.

ACKNOWLEDGMENT

This work is (partially) funded by the European Commission grant 770469-CUTLER and 815362-PRINCE

REFERENCES

[1] ZDNet. (2019). ADHA privacy boss reportedly quits as My Health Record faces first big test | ZDNet. [online] Available at: https://www.zdnet.com/article/adha-privacy-boss-reportedly-quits-as-my-health-record-faces-first-big-test/ [Accessed 5 Aug. 2019].

[2] T. Deschamps, "Sidewalk Labs advisory panel member Saadia Muzaffar quits, citing 'deep dismay'," Financial Post, 05-Oct-2018. [Online]. Available: https://business.financialpost.com/technology/sidewalk-labs-advisory-panel-member-saadia-muzaffar-resigns-citing-deep-dismay. [Accessed: 05-Aug-2019].

[3] Newman, D. (2019). Are Privacy Concerns Halting Smart Cities Indefinitely?. [online] Forbes Available at: https://www.forbes.com/sites/danielnewman/2019/01/08/are-privacy-concerns-halting-smart-cities-indefinitely/#6bfb7f8069ba [Accessed 5 Aug. 2019].

[4] Pandya, Abhinay, et al. "On the use of URLs and hashtags in age prediction of Twitter users." 2018 IEEE International Conference on Information Reuse and Integration (IRI). IEEE, 2018.

[5] Morgan-Lopez, Antonio A., et al. "Predicting age groups of Twitter users based on language and metadata features." PloS one 12.8 (2017): e0183537.

[6] Sloan, Luke, et al. "Who tweets? Deriving the demographic characteristics of age, occupation and social class from Twitter user meta-data." PloS one 10.3 (2015): e0115545.

[7] Bollen, Johan, Huina Mao, and Alberto Pepe. "Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena." Fifth International AAAI Conference on Weblogs and Social Media. 2011.

[8] Tang, Jie, et al. "Quantitative study of individual emotional states in social networks." IEEE Transactions on Affective Computing 3.2 (2011): 132-144.

[9] Hays, James, and Alexei A. Efros. "IM2GPS: estimating geographic information from a single image." 2008 ieee conference on computer vision and pattern recognition. IEEE, 2008.

[10] Narayanan, Arvind, and Vitaly Shmatikov. "De-anonymizing social networks." arXiv preprint arXiv:0903.3276 (2009).

[11] Perez, Beatrice, Mirco Musolesi, and Gianluca Stringhini. "You are your metadata: Identification and obfuscation of social media users using metadata information." Twelfth International AAAI Conference on Web and Social Media. 2018.

[12] Li, Jianzhong, Beng Chin Ooi, and Weiping Wang. "Anonymizing streaming data for privacy protection." 2008 IEEE 24th International Conference on Data Engineering. IEEE, 2008.

[13] Cao, Jianneng, et al. "Castle: Continuously anonymizing data streams." IEEE Transactions on Dependable and Secure Computing 8.3 (2010): 337-352.

[14] W. Wang, J. Li, C. Ai and Y. Li, "Privacy protection on sliding window of data streams," 2007 (CollaborateCom 2007), New York, NY, 2007, pp. 213-221.

[15] https://nifi.apache.org/minifi/getting-started.html

[16] Samarati, Pierangela, and Latanya Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. technical report, SRI International, 1998.

[17] Dewri R., k-anonymization in the presence of publisher preferences, Knowledge and Data Engineering, IEEE Transactions, 23, 1678-1690 (2011).

[18] Machanavajjhala, Ashwin, et al. "l-diversity: Privacy beyond k-anonymity." 22nd International Conference on Data Engineering (ICDE'06). IEEE, 2006.

[19] https://github.com/PanosKostakos/Sentiment-anonymisation-Apache-NiFi

[20] https://github.com/abhinayoulu/anonymizati