How Unique Are You On Twitter - Understanding The Tradeoffs Between Privacy And Utility

A Thesis
submitted to the Faculty of the
College of Arts and Sciences
of Georgetown University
in partial fulfillment of the requirements for the
degree of
Bachelor of Science
in Computer Science

By

Konrad M. Rauscher, Computer Science Undergraduate

Washington, DC
May 1, 2018

How Unique Are You On Twitter - Understanding The Tradeoffs Between Privacy And Utility

Konrad M. Rauscher, Computer Science Undergraduate

Thesis Advisor: Dr. Lisa Singh

Abstract

Individuals share information freely online, often with little regard for their privacy. We are interested in understanding the significance of such behavior from the perspective of data distinguishability, the level of similarity between individuals given their online posts and data utility in the context of data mining tasks. This thesis investigates this tradeoff between individual privacy and data utility. We begin by quantifying the uniqueness of a sample of 5600 Twitter users posting over 4 million tweets. We then attempt to reduce their distinguishability by generating different semantic projections of the users' data. The projections we consider include frequent words, emotion content, and highly distinguishing words. We analyze the privacy-utility tradeoff of these projections, and find that Twitter users have low privacy with a Bag of Words representation of their tweet stream, but high utility. Reasonable privacy only exists for projections involving a small number of semantic features. Unfortunately, in these cases, the data utility decreases rapidly with respect to three standard data mining tasks.

Index words: Privacy, Text Mining, Text Compression

Contents

Chapter 1

Introduction

Social media has become a ubiquitous aspect of modern life, particularly for millennials. It is now common for an individual to have accounts across several social networks, where entire aspects of their lives (political views, favorite shows, music taste, and even personality and emotions) are represented in great detail. In this individual sense, it can be said that many individuals already lack a significant degree of privacy. Moreover, we now live in a world in which seemingly trivial attributes have been used to construct informative, nuanced, and comprehensive models for distinguishing individuals to shape sentiments across entire societies [4][2][10]. Not only can the public data corresponding to an individual, accumulated through their social media, be used to develop models of an individual's personality traits and susceptibility to being influenced, these data are often easy to obtain. Applications have been known to pull social media data from friends of a user [1] and Twitter's own privacy policy states that it may share user information with other companies, as examples [17].

In the context of data privacy, measures are often used to evaluate the extent to which an attribute can be attributed to an individual and the extent to which an individual, is uniquely identifiable from a set of individuals [9]. In this work, we are interested in the extent that specific attributes of posts (number of posts, emotionality of content, characteristics of punctuation use, etc.) contribute to the distinguishability of an individual. If representations of individuals comprised of such

attributes are transformed by use of lossy compression techniques, such that specific attributes are left out, or the detail of a given attribute is decreased, then privacy (in terms of distinguishability of an individual) may be gained. However, from a data mining utility perspective such gains in privacy may be offset by a resultant loss of utility in terms of specific data mining tasks, such as frequent itemset mining, clustering, or anomaly detection.

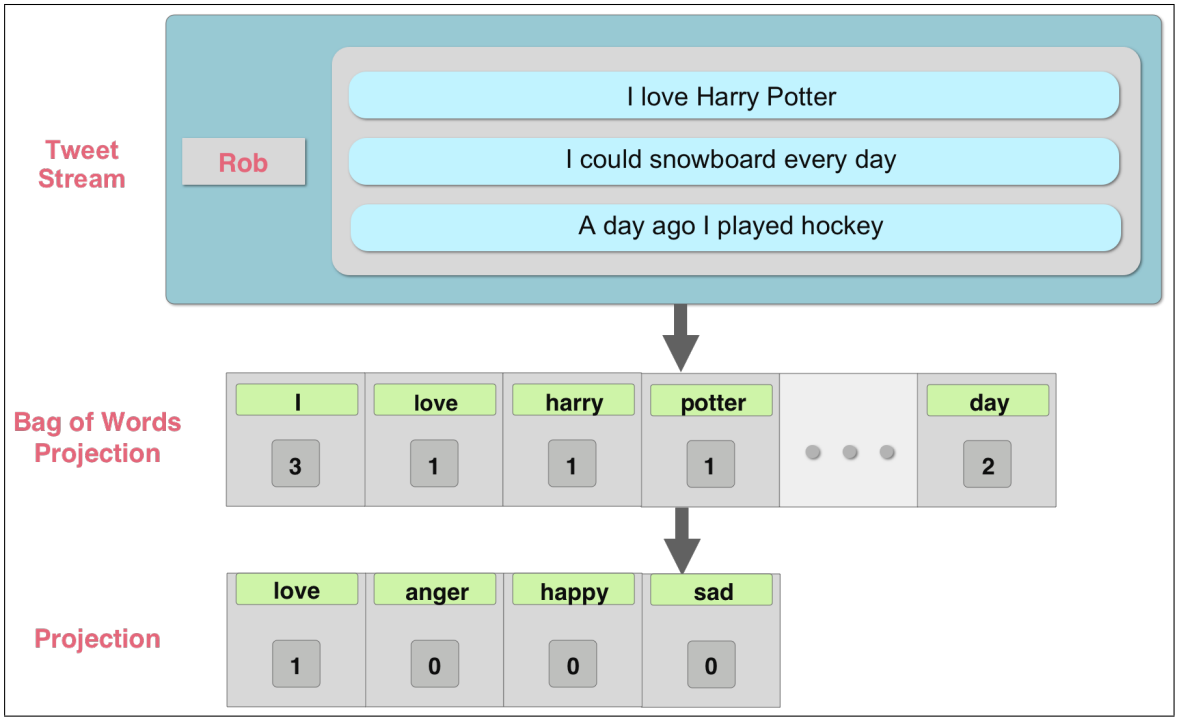

Figure 1.1: Example - Transformation of Tweets to Vector Projections

An example of such a lossy transformation can be seen in Figure 1.1. The Twitter user Rob has three tweets in his tweet stream. This tweet stream can be projected as a 'Bag of Words' feature vector, such that Rob is represented by a count of the occurrence of all features present in his tweet stream. Rob can also be represented by a lossy projection, if he is represented according to a subset of the features present in the Bag of Words projection of his tweets. The goal of this work is to investigate this tradeoff between privacy and utility for social media data.

We begin with a fundamental question: What makes an individual unique on Twitter? We focus on the following four components: (1) the words used by a Twitter user (handle), (2) the substantive content discussed by a user (e.g. '#nascar', using hashtags as a proxy for substantive content domains), (3) the amount of emotion expressed by a user, and (4) the extent to which a user engages in conversation on Twitter (i.e. tweet frequency). We select these components by observing that what distinguishes individuals on Twitter is the substantive content they discuss, and how they talk about this substantive content. We use these four characteristics to analyze (1) what distinguishes a Twitter user from others (i.e. privacy), and (2) what features make a Twitter user useful for specific data mining tasks (i.e. utility). In an ideal world, we will remove all the distinguishing features of each user and not lose any accuracy for different data mining tasks.[1]

To summarize, our contributions are as follows. To begin with, to the best of our knowledge this is the first work that analyses social media distinguishability. Second, we propose a framework for analyzing the privacy-utility tradeoff for social media text posts in the context of traditional data mining tasks. Finally, we conduct an empirical analysis on 5626 Twitter users of 4 million tweets and show that users are only private if represented by a small number of features and that a high degree of data utility is lost across our projections of tweet streams for anomaly detection, frequent itemset mining, and clustering tasks.

In general, this thesis helps evaluate the extent to which an individual can be tracked across their expressions in online contexts, given the publicly available information corresponding to said individual. The remainder of this thesis is structured as follows: we begin by reviewing related literature in Section 2. We then formalize the problem and notation in Section 3. Section 4 details our proposed methodology for

[1]This scenario is the maximum privacy, maximum utility scenario.

determining the distinguishability of individuals represented by social media data, and describes the projections we construct. In Section 5, we empirically evaluate our approach using a Twitter data set. Finally, conclusions and future directions are presented in Section 6.

Chapter 2

Related Literature

In this chapter, we review the relevant literature. We divide the literature into two subsections, methods for improving and understanding privacy in Section 2.1 and methods for summarizing text in Section 2.2, emphasizing emotion since that is one data projection we consider in this work.

2.1 Privacy

Singh et al. propose methods for determining what information and level of confidences about an individual can be determined using only publicly accessible data [16]. Although many papers exist that the explore techniques for exploiting information leakage—i.e., data that are exposed through accidental channels, the authors address a more basic question: how much information is revealed by directly publishing data on the web. Using public data from Twitter, Google+, LinkedIn, and FourSquare, they identify the accuracy of different user's webfootprints (a set of beliefs about a user's attributes that may be inferred by an adversary using only public sources of information). A result of significance was the finding that there is enough variation in common attributes to uniquely identify people with high accuracy if an adversary (a malicious agent attempting to acquire more information about an individual) knows even a small number of these attributes. This work differs from ours in that it proposes an algorithm for constructing user web footprints. Our work is attempting to

understand how unique profiles in a Twitter sample are and what transformations help provide more privacy while maintaining data utility.

Li and Li propose a methodology for measuring privacy loss and utility loss arising from anonymization techniques utilized in microdata publishing, and subsequently evaluate the tradeoff between privacy and utility of different anonymization strategies and privacy requirements [9]. The authors first identify and discuss several observations about the nature of privacy and utility that address the misconceptions underling misguided approaches to evaluating these two concepts. The first of these observations is that specific knowledge (that about a small group of individuals) has a larger impact on privacy, while aggregate information (that about a large group of individuals) has a larger impact on utility. Their second observation is that privacy is an individual concept and should be measured separately for every individual, while utility is an aggregate concept and should be measured accumulatively for all useful knowledge. These two observations inform their position that it is inappropriate to directly compare privacy with utility, because an anonymized dataset is safe to be published only when privacy for each individual is protected; on the other hand, utility gain adds up when multiple pieces of knowledge are learned.

Given their conclusion that privacy and utility cannot be directly compared, the authors propose a method for evaluating the tradeoffs between privacy and utility that borrows the efficient frontier concept from Modern Portfolio Theory. Just as with privacy and utility, risk and expected return cannot be directly compared when attempting to balance expected return with risk in the construction of a portfolio. One can, however, use points on a two-dimensional plane (one dimension as risk or privacy, and the other being expected return or utility, as examples) to represent portfolios (or a given tradeoff between privacy and utility, as achieved by an anonymization technique, and determined by given privacy or utility metrics in our case) and the efficient

frontier consists of all portfolios such that there does not exist another portfolio with both lower risk and higher expected return (which would be more efficient). In their framework, privacy loss is quantified by the adversary's knowledge gain about the sensitive values of specific individuals, where the baseline is the trivially-anonymized data for which all quasi-identifiers are removed and utility loss is measured by the information loss about the sensitive values of large populations, with the baseline being the original data.

To evaluate their approach, the authors used an implementation of Mondrian [17] to enforce four privacy requirements: k-anonytmity, l-diversity, t-closeness, and semantic privacy across seven attributes, given both generalization and bucketization anonymization techniques, on the adult data set from the UCI Machine Learning Repository. The authors results demonstrate the similarity between the privacy-utility tradeoff in data publishing and the risk-return tradeoff in financial investment, such that there exists an efficient frontier in data publishing, which consists of all anonymized datasets such that there does not exist another anonymized dataset with both lower privacy loss and lower utility loss. In our work, we (1) drew upon the concept of equivalence classes for the k-anonymity privacy metric to create a metric for distinguishability in terms of the number of collisions (i.e. size of an equivalence class) across different representations of the public Twitter data corresponding to individuals, and (2) approach privacy as an individual concept, and utility as an aggregate concept, as suggested by the authors.

2.2 Text Summarization

There are a number of different types of signals that we can extract from text or different ways that text can be summarized. Here we describe literature related to extract-

ing signals that we believe may be interesting for maintaining properties of tweets. Specifically, we focus on determining emotion and identifying meaningful words from a corpus.

Witten et al. investigate the resulting trade-off between the subjective quality of a projection of text, which has distinct implications for the utility retained by the projection, and the compression factor of said projection and propose two novel techniques that utilize techniques from the field of lossy image compression, Thesaurus substitution and Generative compression, for the lossy compression of text [18]. Because any word-based compression approach, such as their Thesaurus substitution method, limits the amount of compression that can be achieved if used in isolation, the authors also evaluated syntactic techniques that utilize fractal compression for the generation of approximate text (i.e. Generative compression). By examining the potential of such image compression techniques as transform coding, vector quantization and fractal approximation for improving lossy text compression, the authors were able to demonstrate the benefits of adapting these techniques from the image compression world to that of lossy text compression. The author's 'dataset' consists of short texts (excerpts from a paper cited in their work and a book written by Hemingway), used in the examples exhibited in this paper to merely indicate the underlying power of their techniques, and not establish results of statistical significance. In our work, we draw from their approach by considering different word based compressions. However, our goal is privacy and our utility is measuring using data mining tasks.

Horspool and Cormack evaluate generalizations of compression algorithms for text that are word-based, instead of the norm of character-based approaches to text compression [6]. Most text compression algorithms perform compression at the character level. These algorithms are at a disadvantage because they ignore longer range correlations. This may also lead to better compression performance, both in terms of

an achieved compression factor and the speed of the compression. In this paper, the authors explore the use of words as the basic unit (defined as either an alphanumeric string or a punctuation string, a generalization that allows for the decomposition of many types of text files into a sequence of words) for compression algorithms. The authors compare the performance of four word-based text compression methods that utilize concepts discussed in this work against the UNIX compress program as a benchmark.

The overall result of this comparison is that these four approaches achieve consistently better performance than the authors' benchmark, the UNIX compress program. This is explained by the authors as arising from an apparently sufficient recurrence of words in the files used in their evaluation (online manual pages for both both csh and make, a LaTeX file, and a C source file), such that any scheme for compressing repeated references to words will perform well. This work differs from ours in that the authors discuss word-based text compressions in terms of the speed and relative size reduction that novel compress techniques achieve at a document level, and not the impact these techniques have on utility and distinguishability of a set of documents or samples. In our work, we utilize an approach similar to that of the authors', by primarily using larger units than single characters (excluding a few key characters, such as emoji and orthographic markers) as the basic store of elements for our text-based compression techniques.

Qadir and Riloff propose a bootstrapping algorithm to learn hashtags that convey emotion. Given that hashtags are a common occurrence in tweets (of a sample collected in 2011 of 0.6 million tweets, 14.6% of tweets contained at least one hashtag), effective usage of hashtags towards emotional classification is desirable [15]. Further, in tweets that express emotion, it is common to find hashtags representing the emotion felt by the tweeter. They began with a small number of seed hashtags for each

emotion, which were used to automatically label tweets as initial training data. They then trained emotion classifiers and used them to identify and score candidate emotion hashtags; they selected the hashtags with the highest scores, and used them to automatically harvest new tweets from Twitter, and repeat the bootstrapping process. Qualitative observations included: (1) many hashtags include multiword phrases, rather than mere words (2) elongated forms of words are used in hashtags to put emphasis on emotion state (i.e. #yaaaaay) (3) words are often spelled creatively by replacing a word with a number or replacing some characters with phonetically similar characters (i.e. #only4you). These stylistic variations make it difficult to create a repository of emotion hashtags manually. The corpus of hashtag lists created through their bootstrapping algorithm consistently improved recall across all five emotions; compared to the their baseline unigram classifier, their N-gram classifier, in union with the lookup in the learned lists of emotion hashtags showed substantial gains. This approach improved recall over the baseline unigram classifier by 17% for affection, 19% for anger/rage, 20% for fear/anxiety, 6% for joy, and 9% for sadness/disappointment, with precision remaining about the same as compared to the baseline classifier. The authors' observations that (1) hashtags occur frequently in tweets, (2) the regularity of hashtags representing the emotion expressed in emotional tweets, and (3) and the impact that the evaluation of hashtags can have in increasing the recall of emotion classification of tweets informed our placing greater weight on hashtag words. One of our data projections focuses on a feature space comprised primarily of hashtags.

Doddsa et al. present evidence of a deep imprint of human sociality in language, such that the words of natural human language posses a universal bias towards the prevalence, significance, and diversity of positive words [5]. Using an exhaustive, data-driven analysis of positivity bias, they were able to confirm the Pollyana hypotheses, which states that positivity bias exists in human communication. They began with

the construction of 24 corpora spread across 10 languages, sourced from a broad range of texts that encompassed books, news outlets, social media, web, television and movie subtitles, and music lyrics. In order to examine positivity bias, the authors first focused on understanding what words are most commonly used by people, and then measured how those same words were received by individuals through the use of human labelers. To achieve the first task, they selected between 5k and 10k of the most frequently used words for each of their corporate, choosing the exact numbers such that they obtained approximately 10k words for each language. To accomplish the second, they paid native speakers to rate how they felt in response to individual words on a nine-point scale, with 1 corresponding to most negative or saddest, 5 to neutral, and 9 to most positive or happiest. Overall, 50 ratings per word were collected, for a total of around 5 million individual human assessments. Their results clearly showed the prevalence of a happiness bias, with two results of particular significance: (1) for all corpora, the median happiness score clearly exceeded the neutral score of 5 and (2) average word happiness is largely independent of word use frequency. Given the positivity bias described in this work, we expect positive emotional expressions to increase levels of privacy of individuals, relative to other emotional categories.

Roberts et al. propose several challenges unique to the classification of individual emotions and discuss the importance of incorporating lexical features for detecting emotions [**?**]. The objective of their work was to contribute towards the design of novel emotion detection techniques that account for linguistic style and psycholinguistic theories. They drew tweets from across 14 topics believed to evoke emotion to create a corpus of 7000 labeled tweets for 7 emotions, yielding 500 tweets for each of their topics. In order to remove both duplicates and highly similar tweets during the construction of this corpus, the authors utilized a de-duplication method based on Dice's coefficient. For their evaluation, the authors used a series of binary SVM

classifiers with 10-fold validation that took as inputs the best-performing set of features for each emotion from the following features (chosen greedily, with the next-best feature being required to increase the F1 score in order for the feature to be used for a given emotion): Unigrams: after filtering, Bigrams, Trigrams, Contains !, Contains ?, WordNet synsets, WordNet hypernyms (All recursive hypernyms for each synset), topic scores (the scores for each LDA topic with 100 topics used), and significant words (unigrams judged to have a high pointwise mutual information (PMI) with at least one emotion in the training data). The best performing emotion was FEAR, which was also the least frequent. Furthermore the FEAR classifier uses only two features (unigrams and topics). This suggests this emotion is highly leixicialized with less variation than the other emotions, as it has comparable recall, but significantly higher precision. The second least frequent emotion, SURPRISE, had the worst performance despite using the second greatest number of features. However, this emotion often involved a great deal of real-world knowledge. For example, given a (bogus) tweet such as "Napoleon was actually six feet tall", the only lexical clue is the word actually. Otherwise, one would have to know that Napoleon was perceived as being short in order to understand that SURPRISE is being evoked. This work was informative for our work as it provides a reasonable k agreement metric for emotion labeling, and establishes exceptions and contributing factors for emotional classification performance across specific emotions in the context of tweets.

Yang, Hsin-Yih, and Chen propose methods for the construction and evaluation of emotional lexicons and contribute the observation that the average lengths of articles with tagged emotions that contain emoji are shorter than those that do not, because of a seeming 'truncation of expression' that occurs when emoji are used to express emotion [19]. This may have significance for the usefulness of the introduction of a 'shorter_than_average' term feature injection, that could be significant for emotion

detection, if certain conditions are met. The authors sourced emotional expressions for their dataset from blog articles posted on the Yahoo! Kimo Blog service, which has 40 emoji that a blogger can easily insert when editing an article. These 40 emoji were treated by the authors as distinct emotion categories and taggings on corresponding text expressions. Their dataset consisted of 5,422,420 Yahoo! Kimo Blog articles, corresponding to 336,161 bloggers, and published from January to July, 2006, spanning a period of 212 days. To construct their emotion lexicons, the authors adopt a variation of pointwise mutual information to measure collocation strength between their 40 emotions and the words present in 1,185,131 sentences containing only one emoticon (i.e. emotion) from their dataset. After determining the collocation strengths across these word emotion pairs, the top n collocations were selected to create different lexicons. To evaluate these lexicons, the authors measured the classification performance achieved by ascribing an emotional labels across a test set of 307,751 sentences. This process was undertaken across three sets of emotional categories: one of size 40 (corresponding to the set of Yahoo! Kimo Blog emoji), one of size 4 (obtained by dividing the 40 emoji-emotions across Thayer's emotion categories), and one of size 2 (achieved by dividing the 40 emoji-emotions across a positive and negative class). As a baseline for evaluating classification performance of their lexicons across these sets of emotional categories, the authors used the precision of predicting the majority emotion category (in their case either 'happy' or 'positive'). It is worth noting that this baseline method exhibits an increase in precision as the number of emotion classes decreases. In terms of results, the authors demonstrate that one of their lexicons achieves better performance in terms of precision and recall than the basline, across all three emotional category sets. This work guided our approach to pre-processing and the construction of our own emotional lexicon, used to create the feature space of one of our projection techniques. Although we expected that emoji

would be highly informative for emotion classification on an a-priori basis, the authors' observation that online expressions containing emoji are likely to have fewer words (and thus fewer features informative for emotional classification), and their ability to construct high-performing emotional lexicon using word-emoji collocations underscored the importance of emoji in our approach to emotional classification. Accordingly, our pre-processing involves the injection of emoji features into tweets in a manner that reduces emoji expressing similar nuances of emotion into a single feature. Our emotional lexicon differs from those of the authors, however, in that it consists of words and emoji features, not word-emotion collocation strengths. This is because our approaches are primarily 'count-based' in nature.

Chapter 3

Definitions and Notation

In this chapter, we present the necessary definitions and notations that will be used through the thesis. We are interested in determining the tradeoffs between privacy and utility when we construct different lossy projections $\hat{W}$ of the tweet streams for each user (handle) in our sample.[1]

3.1 Terminology & Notation

Given a set of Twitter users or handles H, each handle h^i in H publishes a tweet stream T_i that contains a set of tweets, $T_i = t_i^1, t_i^2, \ldots, t_i^n$, where n is the number of tweets in the tweet stream for h_i. Let $N_i = |T_i|$ = the number of tweets published by h_i. We refer to the full tweet stream of a handle h_i as the *raw tweet stream*. We refer to the unique words in the tweet stream as the *Bag of Words* representation of the tweet stream. A tweet stream for h_i can be represented as a set of words. Let W be the set of words, emojis, and punctuation used in T. Suppose the raw tweet streams for each handle h_i in H are mapped to W. Let ω_i represent the vector mapping of h_i's tweet stream to the words in W. Then $\omega = \omega_1, \omega_2, \ldots, \omega_{|H|}$ is the projection of all the users in H for W. A projection $\hat{W}$ is a subset of W that may be useful for different data mining tasks, $\hat{W} \subseteq W$.

[1] Although we are interested in evaluating individuals on Twitter, we acknowledge that many Twitter handles correspond to entities (companies, causes, etc.) and bots. Accordingly, we use the terms 'handle' or 'handles' interchangeably with the term 'user' to refer to Twitter accounts and individuals on Twitter in this work.

$\mathbb{P}$ is defined as the level of privacy or distinguishability of each user h_i in H given $\hat{\omega}$, determined by the *k-anonymous vector* privacy measure. Definition, k-anonymous vector privacy measure: each release of data $\hat{W}$ must be such that every combination of values in a projection must be the same for at least k users. If k is 1, we say that a user's privacy is exposed. As k increases, so does the privacy of the user. $\hat{W}$ is said to satisfy k-anonymity if and only if for each user's projection $\hat{\omega}$, the sequence of values in $\hat{\omega}_i$ appear at least k times in $\hat{\omega}$.

$\mathbb{U}$ is defined as the level of data utility of the tweet streams in the $T_1, T_2, \ldots, T_j$ dataset. We compute this level of data utility differently for each of our three utility methods. For the clustering task, data utility is calculated as follows: we first compute the number of handles with the same cluster label in both the projection being evaluated and the baseline, for each cluster. Data utility is defined more specifically based on the data mining task (see Chapter 4, Section 4.5). In general, a utility value of 0 indicates that the results of the data mining task are so inaccurate (as compared to baseline results), that it is unusable.

3.2 Problem Statement

Given a set of Twitter users H, determine the tradeoff between $\mathbb{P}$ and $\mathbb{U}$ by considering different projections $\hat{W}$ of W for users in H.

Chapter 4

Methodology for Evaluating the Privacy and Utility of Twitter Handles

The goal of this work is to assess the tradeoffs between privacy and data mining utility for different data transformations of a Twitter handle's tweet stream. An ideal transformation would exhibit high utility and low distinguishabiltiy (i.e. high individual privacy). In this chapter, we discuss our methodology for measuring the privacy and utility of Twitter handles.

4.1 General Algorithm

At a high level, we obtain data projections of a set of tweet streams corresponding to a set of handles. We then evaluate these projections against a baseline *Bag of Words* projection in terms of privacy and utility. We consider other data transformations for evaluating the privacy and utility of Twitter handles: Emotion, Frequency, and Importance.

We present our general algorithm, in Algorithm 1 as the pseudo code. The inputs to the algorithm are the set of handles H, the tweet streams for each handle T, and the projection type τ. The outputs of the algorithm are the privacy metrics $\mathbb{P}$ and the utility metrics $\mathbb{U}$. We begin by initializing the empty lists N and ω, the number of tweets published for a user h_i in H and the set of Bag of Words vector projections of the tweet streams in T, respectively. Next, we generate the Bag of Words feature

Algorithm 1 Computation of Privacy and Utility Metrics for Twitter Handles

Input: T, H, τ
Output: $\mathbb{P}$, $\mathbb{U}$
Function:
$N = \varnothing$
$\omega = \varnothing$
$W = generate\_feature\_space(T)$
for T_i **in** T **do**
    $\omega_i \Leftarrow transform(T_i, W)$
    append ω_i **to** ω
    append $|T_i|$ **to** N
end for
$\hat{\omega} = \varnothing$
$\hat{W} = generate\_feature\_space(T, \tau)$
for T_i **in** T **do**
    $\hat{\omega}_i \Leftarrow transform(T_i, \hat{W})$
    append $\hat{\omega}_i$ **to** $\hat{\omega}$
end for
$\mathbb{P} \Leftarrow calculate\_privacy\_metrics(\hat{\omega}, H)$
$\mathbb{U} \Leftarrow calculate\_utility\_metrics(\omega, \hat{\omega}, T, H)$
return $\mathbb{P}, \mathbb{U}$

space W from the set of words, emojis, and punctuation T. Each tweet stream T_i in T is then transformed into a Bag of Words feature vector ω_i, according to the features in W and appended to the Bag of Words projection ω. The number of tweets in tweet stream T_i is also appended to N. We then initialize the empty list $\hat{\omega}$, the set of vector projections of the tweet streams in T. The feature space $\hat{W}$ is then generated from the feature space T, according to the projection type τ specified. Each tweet stream T_i in T is then transformed into a feature vector $\hat{\omega}_i$, according to the features in $\hat{W}$ and appended to the projection $\hat{\omega}$. The privacy and utility metrics $\mathbb{P}$ and $\mathbb{U}$ are then computed and returned. This general methodology is represented in Figure 4.1.

4.2 Preprocessing of Data

The pre-processing of tweets in our dataset involved lemmatization, cleaning, feature injection, and a simple word-duplication technique intended to ascribe greater weight to (1) highly emotional words, (2) words in all caps, and (3) words preceded by the '#' symbol (i.e. hashtag words) in anticipation of using these tweets for word-count-based techniques. We describe this preprocessing approach in greater detail in Chapter 5.

4.3 Data Projection Methods

Our methodology for evaluating the privacy and utility of Twitter profiles is centered around the analysis of two types of transformations of the data (i.e. tweets) for the tweet streams of handles in our dataset: four projections (*Bag of Words*, *Emotion*, *Frequency*, and *Importance*) of handles as vectors into specific feature spaces. Our motivation for computing these projections of tweets from our corpus was that, because of the distinct characteristics of their feature spaces, they would allow for an evaluation of the implications that the (1) the words used by a handle, (2) the substantive

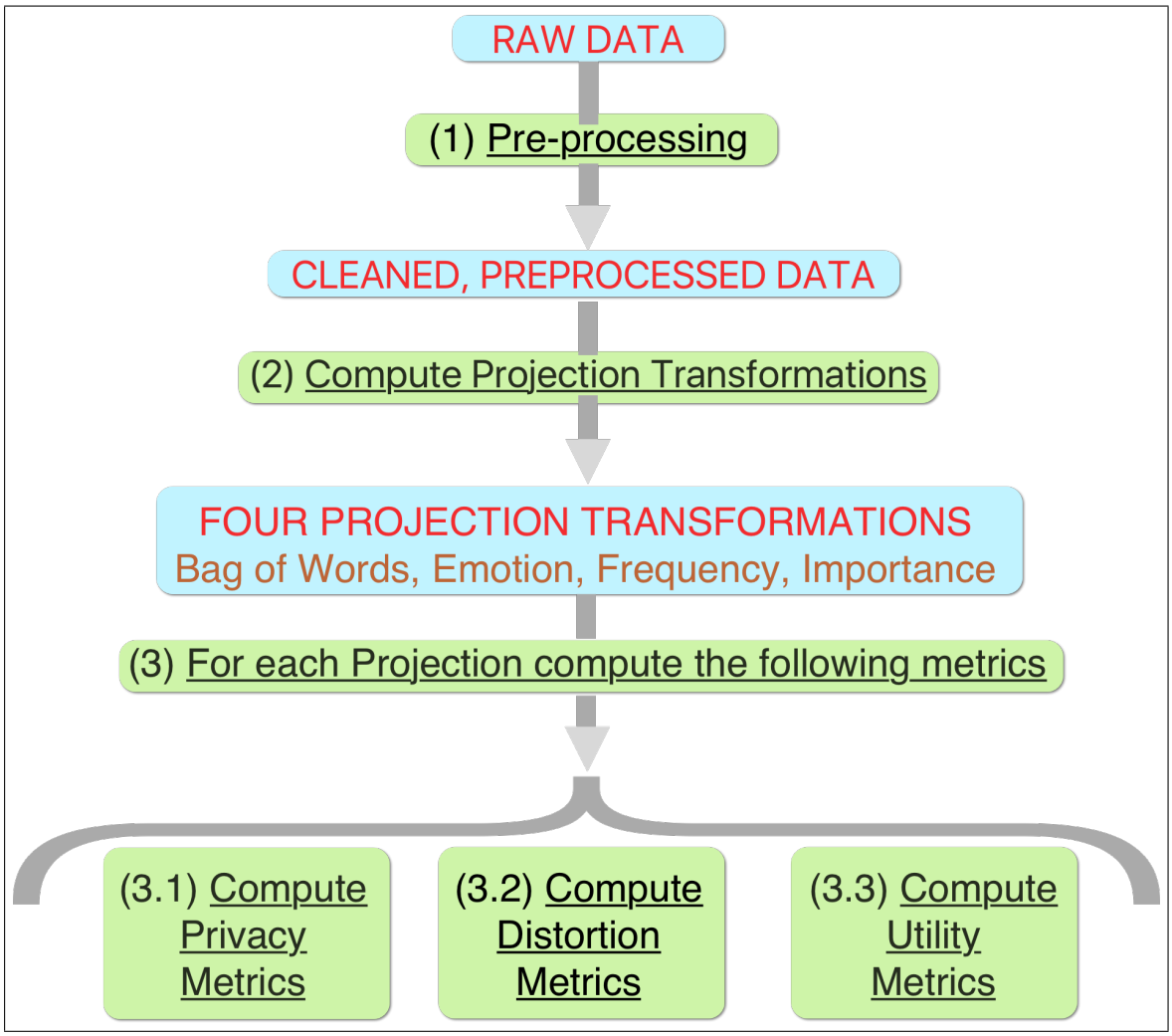

Figure 4.1: General Methodology

content discussed by a handle, and (3) the proportions of emotionality represented by a handle have for privacy and utility. In other words, by comparing the performance of these projections against one another across our privacy and utility metrics, we would be able to compare the degree that handle-level characteristics (1), (2), and (3) have on privacy and utility. Each of these transformations are compressions we

considered interesting in our context, because they vary in the types and degree of utility that they preserve, according to the specific attributes (such as emotion proportions, or word order) that they maintain or drop. Due to the sparsity of the data, we also consider reduced feature spaces for our four projections.

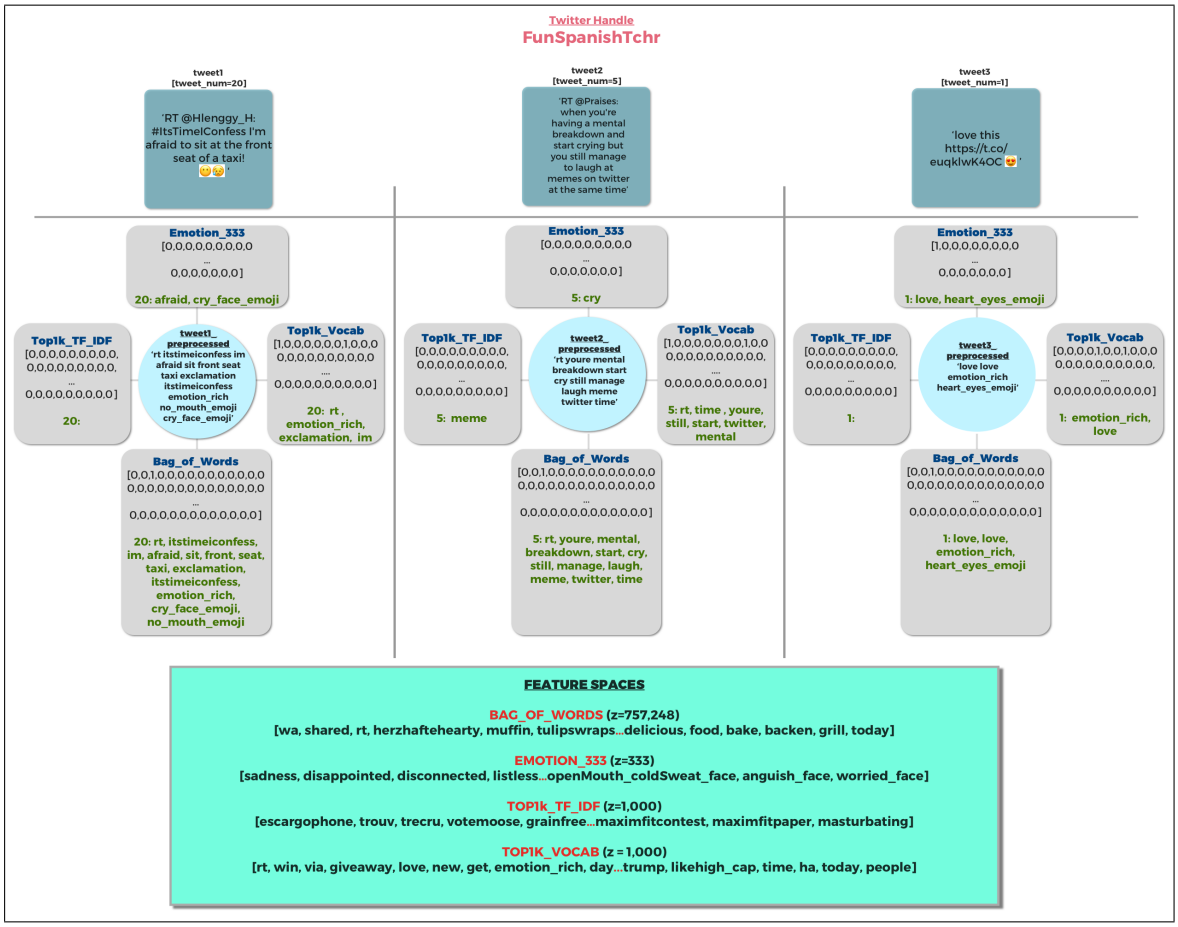

Figure 4.2: Example - Projections of Tweets in a Tweet Stream

4.3.1 Bag of Words

The feature space for our Bag of Words projection consists of the entire set of features present in the tweet streams for our sample. Accordingly, no features are lost when

preprocessed tweets are represented with the Bag of Words projection, as can be seen in Figure 4.1.

4.3.2 Emotion

The feature space for our Emotion projection consists of a set of emotional features (words, emojis, and punctuation). The selection of these features is described in greater detail in Section 5.5. Only emotional features are kept when tweets are represented with the Emotion projection, as can be seen in Figure 4.1.

4.3.3 Frequency

The feature space for our Frequency projection consists of the most frequently according features in the tweet streams for our sample. As shown in Figure 4.1, such high-frequency features are 'rt', 'exclamation', and 'twitter'.

4.3.4 Importance

The feature space for our *Importance* projection consists of the most important words in the tweet streams for our sample. The tf-idf (term frequency, inverse document frequency) measure, a numerical statistic that reflects how important a word is to a document in a collection or corpus, was used to determine the importance of features in our sample. The tf-idf$_{t,d}$ of a term t for a document d is $= (1 + \log(\text{tf}_{t,d})) \cdot \log(\frac{N}{\text{df}_t})$, where N is the number of documents in the sample, df$_t$ is the number of documents the term t occurs in, and tf$_{t,d}$ is the number of occurrences of the term t in document d. The *idf* weighting for a unique term is maximal: log(N), and for a term appearing in all documents, log(1) = 0. We describe our use of the tf-idf measure to obtain important words in greater detail in Section 5.5. Of the three tweets in Figure 4.1, only one feature was kept by our Importance projection, the word 'meme'.

4.4 Privacy Methods

Because privacy is an individual concept [9], we evaluate the impact that a given transformation has on the privacy of a set of handles in terms of the number and sizes of equivalence classes observed across said handles. In this work, an equivalence class is defined as the number of handles sharing the same feature vector. Similar to the *k-anonymity* privacy measure [9], the more members of an equivalence class, the greater the privacy of users with handles corresponding to that vector representation according to our *k-anonymous vector* privacy measure. We evaluate the impact a given transformation has had on the privacy of handles it represents by comparing the distribution of equivalence class sizes for handles represented by a given transformation against the same distribution observed for the *Bag of Words* projection.

Due to the fact that a match cannot occur between two feature vectors with different sets of zero-valued features and feature vectors of Twitter handles with low tweet frequency can be expected to be represented by feature vector across all transformations that are more sparse (i.e. a higher incidence of zero-valued features), we also evaluate equivalence class distributions for handles that belong to the same tweet frequency category (either low, medium, or high). This expectation arose from the observation that fewer tweets corresponding to a handle corresponds to fewer words and other features such as emojis occurring in the tweet stream of a handle. The same expectation holds for the inverse of this dynamic: Twitter handles that have a high tweet frequency.

Additionally, because an exact match across the feature values of any vectors corresponding to a large feature space is an unrealistic expectation, and our desire to account for the distinguishably of individuals that can arise from utilizing the co-occurrence of specific values for a set of features, we also computed the *binary*

k-anonymous vector measure for each of the projections, achieved by converting all non-zero-valued features to 1's before comparing the feature vectors for each user. Such a binary comparison of feature vectors counts matches according to two handles sharing the same set of observed features, and does not require both feature vectors to have the same counts for each feature.

4.5 Utility Methods

Because utility is an aggregate concept [9], we evaluate the utility of our different transformations on the performance of three unsupervised data mining tasks (Clustering, Frequent Itemset Mining, and Anomaly Detection). These specific tasks were selected for our purposes because of the relevance that their characteristics and tasks possess for our context. Our approach for measuring utility is comparing (i) the clusters of handles, (ii) the frequent itemsets for the set of tweet streams for each user in our data set, and (iii) the rankings of handles according to abnormality obtained from the projection under consideration to the Bag of Words projection.

4.5.1 Clustering

The data mining task of clustering is the assignment of a set of observations into subsets (called clusters) so that observations in the same cluster are similar in some sense. We use the clustering method (*K-Means* [8]) for our evaluation. The K-means algorithm starts by randomly defining k centroids. From there, it works in iterative (repetitive) steps to perform two tasks: (i) assign each data point to the closest corresponding centroid, using the standard Euclidean distance (in layman's terms this is is the straight-line distance between the data point and the centroid), and (ii) for

each centroid, calculate the mean of the values of all the points belonging to it. The mean value becomes the new value of the centroid.

For our methodology, a clustering task allows for the evaluation of potential differences in performance (i.e. cluster quality) arising from differences in terms characteristics, assumptions, strengths, and weakness could be utilized in the analysis of our transformations. Clustering was chosen for evaluating utility because by applying tweets or collections of tweets to a clustering method, one can ascertain the features being highlighted by a given transformation. In this sense, clustering techniques are useful for examining the extent to which attributes under consideration (the incidence of *anger_ disgust* emotionality across the set of tweets corresponding to a handle, as an example) map to categories thought to be relevant for determining distinguishability. It should be noted that this clustering method is being used in a general sense; we are trying to see if it is *possible* to obtain meaningful categories for handles in our sets, rather than applying these techniques for specific tasks.

We compute the data utility lost for clustering from projecting the tweet streams in our sample to one of our feature spaces as follows. For $\omega_i \subseteq \omega$ & $\hat{\omega}_i \subseteq \hat{\omega}$, utility loss $\mathbb{U}$ is defined as the normalized count of cluster labels shared by ω and $\hat{\omega}$:

$$\mathbb{U} = \frac{\sum_{j=1}^{|C|} |\hat{\omega}_j \subseteq \omega_j|}{|C|}$$

This results in a value in the range of 0 and 1, with a 0 meaning no matches between the evaluated projection and the baseline, and a 1 meaning every handle sharing the same cluster between the evaluated projection and the baseline.

4.5.2 Frequent Itemset Mining

For our Frequent Itemset Mining task, the computation of the most frequently occurring sets of features across all tweet streams in our data set, we performed the *BitDrill*

[14] implementation of frequent itemset mining algorithm at a corpus level for our each of our transformations, such that we could evaluate how frequent (and therefore non-distinguishing) feature co-occurrences varied across our transformations.

We compute the data utility lost for frequent itemset mining from projecting the tweet streams in our sample to one of our feature spaces as follows. For ω and $\hat{\omega}$, utility loss $\mathbb{U}$ is defined as the normalized difference in the set of frequent itemsets S_ω and $S_{\hat{\omega}}$ mined from ω and $\hat{\omega}$:

$$\mathbb{U} = \frac{\frac{|S_{\hat{\omega}} \cap S_\omega|}{|S_{\hat{\omega}}|} + \frac{|S_\omega \cap S_{\hat{\omega}}|}{|S_\omega|}}{2}$$

This equation is the average of two proportions: (i) proportion of frequent items sets in $S_{\hat{\omega}}$ from projection $\hat{\omega}$ present in *Bag of Words* frequent itemsets S_ω (precision) and (ii) proportion of *Bag of Words* frequent itemsets in S_ω present in frequent itemsets of projection $\hat{\omega}$ (recall). This computation results in a value between the range of 0 and 1, with a 0 meaning the evaluated projection exhibits high utility loss relative to the baseline, and a 1 meaning the evaluated projection exhibits no utility loss relative to the baseline.

4.5.3 Anomaly Detection

For Anomaly Detection,[1] defined for our purposes as the identification of handles (i.e. feature vectors) whose tweet streams are very different from those of other handles, we used the *Local Outlier Factor* [3] estimator, an unsupervised outlier detection method which computes the local density deviation of a given sample with respect to its neighbors. This data mining task is useful for our purposes because the number of feature vectors classified as outliers for a given transformation can be used as a

[1]The identification of items, events or observations which do not conform to an expected pattern or other items in a dataset

proxy for evaluating the distinguishability of handles within that transformation. We compute the data utility lost for anomaly detection from projecting the tweet streams in our sample to one of our feature spaces as follows. For the set of abnormality rankings R for ω_i in ω and the set of abnormality rankings $\hat{R}$ for $\hat{\omega}_i$ in $\hat{\omega}$, utility loss $\mathbb{U}$ is defined as the Kendall rank correlation coefficient [12] for R and $\hat{R}$:

$$\mathbb{U} = \frac{1}{|H|(|H|-1)} \sum_{i \neq j} sgn(\hat{r_i} - \hat{r_j}) sgn(\hat{r_i} - \hat{r_j})$$

A Kendall Tau coefficient of 1 would indicate a perfect match between the abnormality rankings of the baseline and the projection being evaluated, while a value of -1 would indicate a complete mismatch (i.e. completely opposite orderings) between the abnormality rankings.

Chapter 5

Evaluation and Discussion

In this chapter, we will conduct an empirical evaluation of the tradeoffs between the privacy and utility for representations of a set of Twitter handles. We begin by describing our collection of public data from Twitter handles and introduce the dataset we used to evaluate our methods in section 5.1. We then describe the preprocessing techniques that we applied to this dataset and detail our division of handles according to tweet frequency in 5.3. We subsequently describe our computation of data transformations for the tweet streams corresponding to a set of Twitter handles in Section 5.4. We then present the privacy utility tradeoffs for the different data transformations in Sections 5.5 (privacy) and 5.6 (utility). We also detail our evaluation of the distortion introduced by our data projections in section 5.7.

5.1 Data Set

We are interested in identifying individuals who have more than a minimal level of engagement. So we identified ten authority parenting and medical website (*webmd*) handles and randomly selected approximately 10,000 follower handles from each of these sites. Using the Twitter API, in total we collected 110,891 Twitter handles. Active collection of tweets for these handles was performed from mid 2016 until the end of 2017. At the commencement of active collection, the most recent 3500 (The maximum number allowed by the Twitter API) tweets were collected for each of our

selected handles. In the case that one of our selected Twitter handles had less than 3501 total tweets at this point, the total tweets corresponding to the handle were collected. Accordingly, the number of tweets corresponding to a handle in our dataset is not a direct proxy for the tweet frequency of a handle over the time period of tweet collection. For these experiments, we use a subset of these data. We sample the data to get variation in number of tweets posted by user. In the end, we run this analysis on 5626 handles and 4 million posts.

5.1.1 Preprocessing of Dataset

The pre-processing of tweets in our dataset involved lemmatization, cleaning, feature injection, and simple word-duplication technique in anticipation of using these tweets for our word-count-based techniques. The cleaning methods used are encompassed by the removal of (1) hyper-links, (2) references to other Twitter handles, (3) words comprised solely of numbers, (4) words comprised solely of numbers and punctuation, (5) stop words, (6) words comprised of a single character, (7) the replacement of contractions such as *'can't'* with *'can_ not'* (8) making all text lowercase once a pre-processing step that evaluates the proportion of capitalization of characters for each tweet has been completed and (9) removing all punctuation once a pre-processing step that evaluates the of exclamation points for each tweet has been completed. We used lemmatization (the reduction of different forms of a word to a single form) to reduce noise in our data set. For cleaning, we removed features and removed variations in expression (capitalization and contractions, for example) we deemed would contribute noise to the data set (i.e. be uninformative for the methods in our approach). For lemmatization, we used *WordNetLemmatizer* from the *nltk.stem.wordnet* package.

Our feature injection techniques were (1) the reduction of sets of emoji and ascii expressions that convey the same emotion into a single feature (several variations

of a crying face emoji were replaced with the feature *cry_face*, for example), (2) the insertion of an *extreme_exclamation* feature into a tweet in the case that 3 or more exclamation points were observed in tweet, *exclamation* in the case that only 1 or 2 were observed, and *high_cap* in the case that over 80% of characters in a tweet are capitalized and (3) the insertion of an *emotion_rich* feature into a tweet in the case that the tweet contained a feature also present in the feature space for our Emotion projection, discussed later in this chapter. This is important because we wish to determine the impact that emotional expressions have on privacy. Finally, our word-duplication technique involved appending of a copy to a given tweet of (1) words preceded by a hashtag symbol, (2) words in all caps, or (3) words present in our aforementioned Emotion feature space, such that these types of words are ascribed a greater weight in subsequent word-count-based methods.

5.2 Division of Handles according to Tweet Frequency

We are interested in levels of engagement on Twitter, i.e. how often a users posts tweets, is likely to have a large impact on privacy. Therefore, we define three categories of engagement level: high, moderate, and low.

When evaluating the privacy of a set of handles, we calculate privacy metrics for each of these engagement levels, as well as the full set of handles. This allows us to observes the extent to which frequency impacts the privacy of handles. The allocation of bucket membership was determined for each of the 5626 handles as follows:

$$low\_engagement : 20 <= number\_of\_tweets <= 200$$

$$moderate\_engagement : 200 < number\_of\_tweets <= 2000$$

$$high\_engagement : 2000 < number\_of\_tweets$$

5.3 Computation of Data Projections

Because we are interested in the implications that (1) the words used by a handle, (2) the substantive content discussed by a handle, and (3) the proportions of emotionality represented by a handle have for privacy and utility, we create four types of vector projections (*Bag of Words*, *Emotion*, *Frequency*, and *Importance*) of the pre-processed tweets, such that each tweet string could be represented as a vector of feature counts (i.e. word counts in most cases), with the composition of the feature space involved for such a vector representation corresponding to the projection method involved. Each of these transformations are compressions we considered to be interesting in our context, because they vary in the types and degree of utility that they preserve, according to the specific attributes (such as emotion proportions, or word order) that they maintain or drop.

5.3.1 Bag of Words

For our baseline *Bag of Words* projection, the feature space used was the set of unique words, plus features injected in preprocessing, across the 5626 users' tweet streams in our dataset. This resulted in 767,937 features. We use this *Bag of Words* projection as our baseline projection, as every tweet transformed in this manner retains all of the words that were present after preprocessing the tweet. Word order, however, is lost with this projection. As will be shown, this projection is unique for all but forty-seven users in our data set. These forty-seven handles share an all-zero Bag of Words projection because the only features in these handles' tweet streams were removed during preprocessing. The majority of such removed features are encompassed by handle names and urls. Although we would like a baseline projection for which all handles are unique (i.e. no preprocessing), remember our utility tasks are clustering,

frequent itemset mining, and anomaly detection; the noise introduced by not preprocessing the tweet streams in our data set would reduce the utility of our data set for these tasks. Therefore, to keep high utility, we took a preprocessed *Bag of Words* as our baseline projection.

5.3.2 Emotion

Our Emotion projection, the *Emotion 333* projection is intended to represent tweets in terms of expressed emotionality. The computation of this projection involves a translation of a set tweet stream into a set feature vectors according to a feature space of 333 emotion words, emoji features, and 3 orthographical-marker features (*high_ cap, exclamation, high_ exclamation*) injected into tweet streams during preprocessing. The emotion words and emoji features in this feature space represent the set of synonyms we constructed for 5 categories of emotion, the process for which we describe for the remainder of this subsection.

Several approaches to capturing emotion-based features that distinguish Twitter users represented in the feature space for our *Emotion 333* projection were combined. The expectation was that the development of 'emotional profiles' of Twitter users would be a worthwhile step towards developing a more comprehensive and distinguishing framework for constructing uniquely identifiable *digital fingerprints* [16] of Twitter users. Accordingly, a determination of a distinct set of emotional categories was undertaken, because of a desire to construct a set of emotional categories that had low overlap between another. We began with 8 preliminary emotional categories (*anger*, *disgust*, *fear*, *joy*, *love*, *sadness*, *surprise*, *none*) and then utilized several classification methods (SVM [7], Naive Bayes [11], and logistic regression [13]) for emotional classification across these 8 emotional categories, such that (1) a determination could be made as to the performance of each approach for classification across each of the

8 emotional categories and (2) overlap between these categories could be evaluated. Because we observed a high degree of classification overlap between the categories *anger* and *disgust*, as well as between *joy* and *love*, we merged both of these pairs, which reduced our final set of emotional categories to 6 (*anger_ disgust*, *fear*, *joy_ love*, *sadness*, *surprise*, *none*).

In order to train the methods described in both the preceding paragraph and following paragraph, we constructed a set of tweets from our data set labeled according to expressed emotionality. Our labeling process involved three phases of labeling and was undertaken by six individuals in our data lab: preliminary, real, and reliability. Phase 1 involved a preliminary label sheet with unique 10 tweets was provided to each of our labelers, with the instructions to (1) specify words deemed to be emotion_rich (i.e. expressing emotionality) and (2) ascribe one or more labels from the set:

$$anger, disgust, fear, joy, love, sadness, surprise, emotional\_ ambiguity, none$$

for each tweet. We then provided feedback to each labeler for this preliminary task, to improve and standardize their subsequent labeling behavior. For phase 2, we provided each of our 6 labelers with the same instructions and a set of 100 unique tweets from our data set. These 600 labeled tweets became our labeled tweet set. For phase 3, we had each of our 6 labelers label the same 20 tweets, using the same instructions as before, such that we could evaluate the reliability of the labelers.

We also developed a novel *Emotional_ Lexicon_ Hit* emotion classification method, which classifies tweets according to our final set of emotional categories, based on which emotional category synonym list had the most 'hits' for a given tweet, with a 'hit' being defined as the number of words in a tweet that are present in one of our five emotional category synonym lists (the emotional category of *none* having no synonym list). This classification technique achieves an average precision of 0.551

and an average recall of 0.662 across the five emotional categories. We recognize that this is low and leave improving this as a future work. The lexicons for each emotional category include both words and emoji and were constructed by (1) human selection of fundamental synonyms and (2) synonyms that appeared frequently across tweets classified by Naive Bayes as belonging to the corresponding emotional category. In terms of the size of these synonym lists, *anger_ disgust* contained 84 elements (60 *anger*, 24 *disgust*), *fear* 52 elements, *surprise* 31 elements, *sadness* 82 elements, and *joy_ love* 94 elements (62 *joy*, 32 *love*). We provide examples of the synonyms for these emotional categories in table 5.2.

Table 5.2 Example Emotional Category Synonyms

anger_disgust	fear	surprise	sadness	joy_love
anger	fear	surprise	sadness	joy
disgust	fright	amazed	sad	love
sickening	frightened	astonished	dispirited	encouraged
outraged	panic	astounded	heartbroken	smiling
ew	worry	dumbfounded	morose	laugh
angry_pout_face	fearful_face	open_mouth_face	cry_face	grin_face
stream_from_nose_face	worried_face	surprise_face	frown_face	heart_emoji
...	...	...	...	...

5.3.3 Frequency

Our Frequency projection, *Top 1k Vocab*, represents tweets according to a feature space equivalent to the top 1k most frequently occurring 1,000 words across the most frequent 1,000 words for each handle. This projection represents tweets in terms of the most frequent words used by our handles, at the corpus level. Because only words with high level frequency are retained with this projection, this approach has the potential to introduce a high degree of noise across a set of tweets translated in this manner.

5.3.4 Importance

Our Importance projection, *Top 1k TF-IDF* uses the most important 1,000 words across the set of tweet streams in our dataset, according to the tf-idf importance measure, as its feature space. Because tf-idf is a measure for how important a word is to a document in a corpus, to find the top words according to tf-idf for the entire corpus (all handles and corresponding tweets), the tf-idf for each word was calculated on a per handle basis (if the word occurred for the handle) and the maximum tf-idf score for these words across all handles was obtained. In other words, all the tweets for a single user are considered a single document. The top 1,000 words according to maximum tf-idf score are then selected as the feature space for the sparse representation of all tweets in the corpus. Because 94% of words that comprise the feature space *Top 1k TF-IDF* projection are hashtags, this projection can be thought of as representing tweets in terms of the substantive content being discussed (i.e. using hashtags as a proxy for topics).

5.3.5 Projections Summary

Our motivation for computing these projections of handles from our corpus was that, because of the distinct characteristics of their feature spaces, they would allow for an evaluation of the implications that the (1) the words used by a handle, (2) the substantive content discussed by a handle, and (3) the proportions of emotionality represented by a handle have for privacy and utility. In other words, by comparing the performance of these projections against one another across our privacy and utility metrics, we would be able to compare the degree that handle-level characteristics (1), (2), and (3) have on privacy and utility.

5.4 Computation of Privacy Metrics

We compute the distinguishability of users represented by a projection according our k-anonymous vector privacy measure. We evaluate the impact a given transformation has had on the privacy of handles it represents by comparing the k-anonymous vector privacy measure for a given transformation against the same measure observed for the *Bag of Words* projection. According to our k-anonymous vector privacy measure, the more members of the smallest equivalence class for a projection, the greater the privacy of users with represented by that projection. This is done for (i) all handles in our dataset and (ii) the set of handles for each of our three engagement levels (either low, moderate, or high). We also compute privacy gain according to our k-anonymous vector privacy measure across both (i) full and (ii) a range of reduced feature spaces for our projections. We do not count handles with all-zero tweet stream projections as an equivalence class because such users can be said to be 'private' in the sense that no features, distinguishing or otherwise, have been kept but they have zero utility.

Table 5.6 Privacy Gains from Baseline

Projection	Features Used	ℙ, Full	ℙ, High	ℙ, Med	ℙ, Low	# All-0 Full	# All-0 High	# All-0 Med	# All-0 Low
exact Bag_of_Words	767,937	0.008	0.0	0.0	0.0	47	0	0	0
binary Bag_of_Words	767,937	0.010	0.0	0.0	0.0	47	0	0	0
exact Emotion_333	all 333	0.157	0.0	0.001	0.123	1967	0	0	57
binary Emotion_333	all 333	0.186	0.0	0.001	0.186	1967	0	0	57
exact EmotionDict_5	all 5	0.342	0.0	0.020	0.600	1976	0	0	58
binary EmotionDict_5	all 5	0.999	1.0	1.0	0.999	1976	0	0	58
exact top1kTF_IDF	all 1k	0.114	0.0	0.030	0.234	2732	0	16	433
binary top1kTF_IDF	all 1k	0.171	0.0	0.043	0.395	2732	0	16	433
binary top1kTF_IDF	500	0.215	0.001	0.177	0.627	3174	1	84	669
binary top1kTF_IDF	100	0.624	0.483	0.789	0.960	3984	53	446	980
binary top1kTF_IDF	50	0.872	0.844	0.924	0.972	4890	513	782	1070
binary top1kTF_IDF	10	0.975	0.966	1.0	1.0	5420	879	909	1098
binary top1kTF_IDF	5	1.0	1.0	1.0	1.0	5555	977	939	1105
exact top1k_Vocab	all 1k	0.023	0.0	0.0	0.0	106	0	0	0
binary top1k_Vocab	all 1k	0.026	0.0	0.0	0.0	106	0	0	0
binary top1k_Vocab	500	0.0370	0.0	0.0	0.0	137	0	0	0
binary top1k_Vocab	100	0.362	0.982	0.293	0.002	276	0	0	0
binary top1k_Vocab	50	0.500	1.0	0.653	0.006	357	0	0	0
binary top1k_Vocab	10	0.988	1.0	0.998	0.979	0	0	0	1
binary top1k_Vocab	5	0.999	1.0	1.0	1.0	708	708	0	7

We provide Table 5.6 as an overview of the gains in privacy (in terms of our k-anonymous vector privacy metric $\mathbb{P}$) observed across (i) projection types (ii) the number of features retained in the projection (iii) and engagement levels. We also include the number of handles represented by all-zero vectors across the same three parameters. The results of significance, presented in this table, are discussed in the following three subsections.

5.4.1 Results - k-Anonymous Vector Privacy Metric

In this section, we present and discuss the privacy of handles represented by our projections, according to our k anonymous vector privacy metric. Because the non-baseline projections exhibit little privacy for the handles they represent, if any, we reduced the vector length (in terms of number of features) used to represent handles. We expected this would increase the size of equivalence classes. For our *Vocab* and *TF-IDF*

projections, this involved computing equivalence classes sizes for a range of reduced feature spaces sizes until we observed enough handles belonging to an equivalence class of size greater than 2, or until the size of the feature space had been reduced to 5 without any meaningful increase in the average size of equivalence classes. To reduce the feature space used for our emotion projection, we represented tweet streams using 5 features: *sadness_emotionality*, *joy_love_emotionality*, *anger_disgust_emotionality*, *surprise_emotionality*, and *fear_emotionality*. The count for each of these features was incremented for a given tweet stream for each occurrence of a word corresponding to the emotional dictionary for that emotional category.

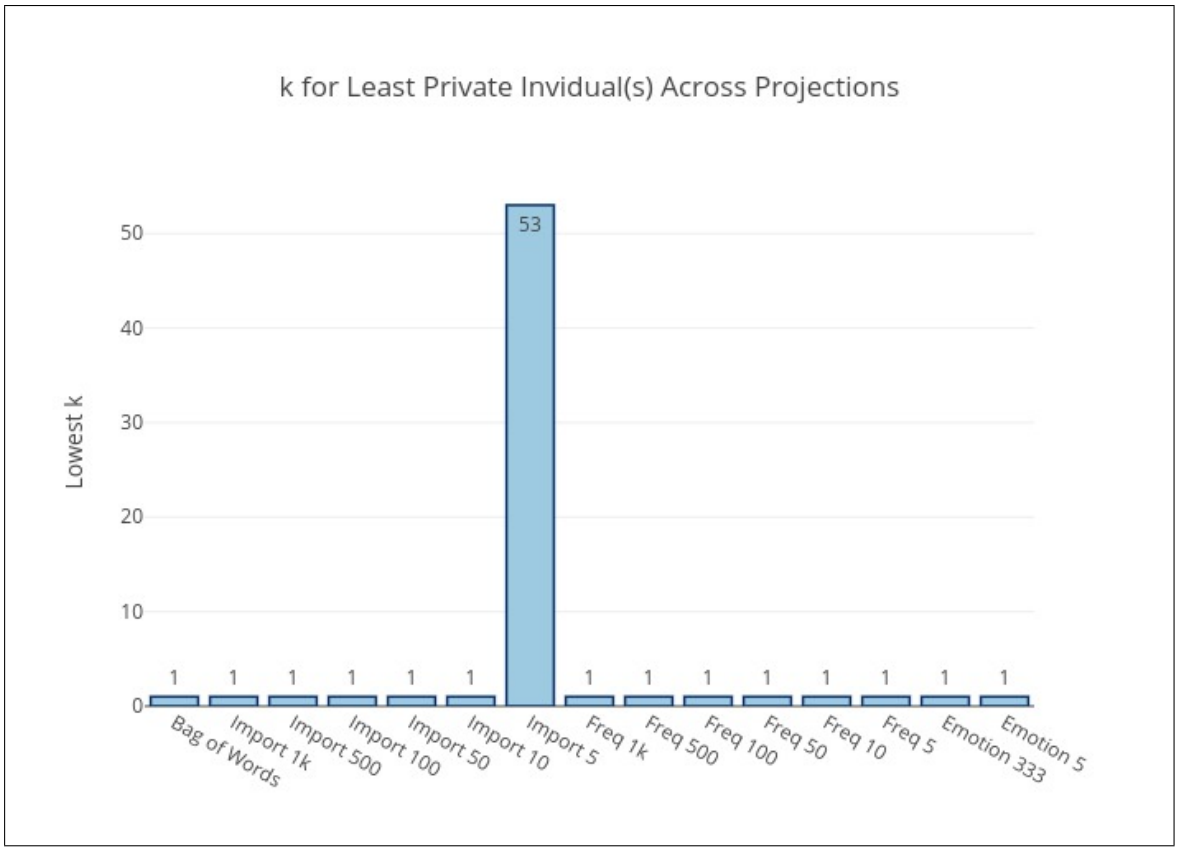

Figure 5.1: k-Anonymous Vector Privacy Metric

Figure 5.1 shows the lowest k (equivalence class size) for each projection, using binary comparison. For this figure, different projection types and feature space sizes

are shown across the x-axis. The y-axis shows the lowest k value for a given projection. Only handles with non all-zero vector representations are counted in this computation. We present results for binary comparisons of feature vectors, because only binary comparison achieves full privacy for handles, when the feature space used to represent them is reduced for each projection. In this sense, binary comparisons are more interesting. Only the Importance projection with a reduced feature space of five (i.e. handles are represented by the most frequent five features) achieves privacy for all handles according to our k-anonymous vector privacy measure. The handles belonging to the smallest equivalence class (of size $k = 53$) for this projection have tweet streams containing each of the five most important features (escargophone, trouv, trecru, votemoose, grainfree) in the full Importance feature space.

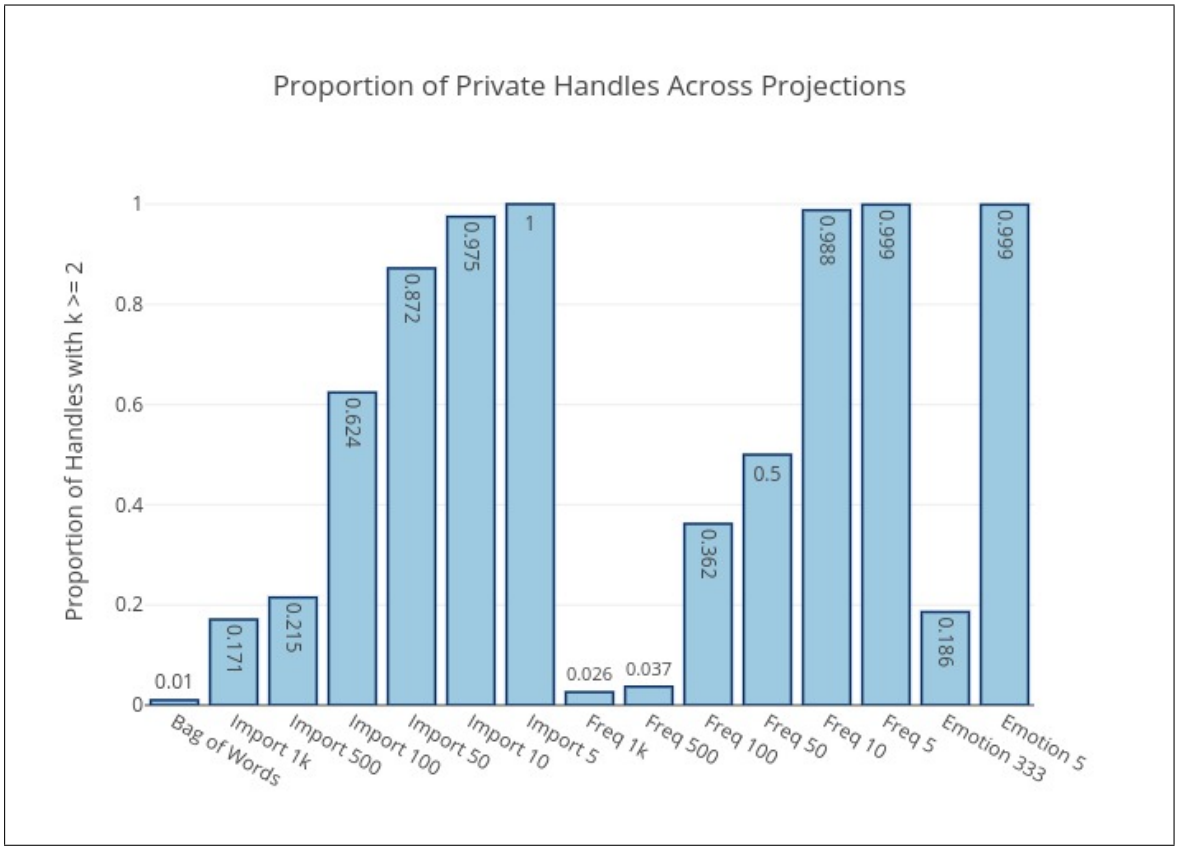

Figure 5.2: k-Anonymous Vector Privacy Proportions

Figure 5.2 shows the proportion of handles with privacy (i.e. $k >= 2$) for each projection, using binary comparison. For this figure, different projection types and feature space sizes are shown across the x-axis. The y-axis shows the proportion of handles in the dataset belonging to an equivalence class of $k >= 2$ for a given projection; handles with all-zero vectors are not included in the computation of this proportion. Handles posses low privacy on average unless smaller feature spaces are used to represent their tweet streams. Only handles with non all-zero vector representations are counted towards this proportion. Again, only the Importance projection with a reduced vector length of five (i.e. handles are represented by the most frequent five features) achieves privacy according to our k-anonymous vector privacy measure. The Frequency and Emotion projections achieve privacy for almost all handles when the size of the feature space used to represent their tweet streams is reduced. The higher privacy of handles represented by the Importance projection can be explained by the high proportion of all-zero vectors this projection results in, particularly when the size of the feature space used is reduced. Accordingly, only a small number of handles (those with non all-zero vectors) are considered when computing the proportion of handles with privacy, such that a proportion of handles with privacy of 1 is more likely.

5.4.2 Results - Equivalence Class Size Distributions

For the graphs in this section, we provide the total number of handles in equivalence classes of size greater than 1 (i.e. *protected* in terms of privacy), and the number of handles corresponding to a vector comprised of all zero-valued features. For these graphs, the x-axis shows the total number of protected handles (i.e. handles that belong to an equivalence classes with a k greater than 1), and the number of handles corresponding to a vector comprised of all zero-valued features. Equivalence class sizes

for handles with a non-all-zero vector projection is shown on the y-axis. For each engagement level, the distribution of equivalence class sizes is shown by a underlying-data-box plot. For this type of box plot, each equivalence class is represented by the set of dots adjacent to each underlying-data-box plot, with the size of each equivalence class represented by the dot's relation to the y-axis. Additionally, triangles are used to represent the standard deviation of the distribution of equivalence class sizes for each engagement level, where the height of a triangle in relation to the y-axis is the standard deviation of handle class sizes. In cases of low variance, this representation of standard deviation is often too small to be seen. Because these triangles are representing the degree of variance present in the observations (and not the distribution itself), these triangles can can go below zero even though there are no negative equivalence class sizes.

In Figure 5.3 we compare the level of privacy for users who tweet similar amounts for our *Bag of Words* projection, where equivalence class membership is computed by exact comparison of feature vectors. There are 46 handles in equivalence classes of size greater than 1 for the full handles set that contain more than 1 handle, and no handles in equivalence classes of size greater than 1 for the full handles set that contain more than 1 handle for the tweet frequency handle buckets. The lack of equivalence classes of size greater than one for our low, moderate, and high handle engagement levels, despite equivalence classes of size greater than one in the full set of handles, is explained by handles with equivalence class sizes greater than one, for the full set of handles, all have fewer than 20 tweets (and are thus more likely to have an all-zero tweet stream projection). The handles in the low, moderate, and high tweet frequency handle buckets, on the other hand, all have 20 or greater tweets in their tweet streams, because we require a minimum of 20 tweets in a tweet stream, for a

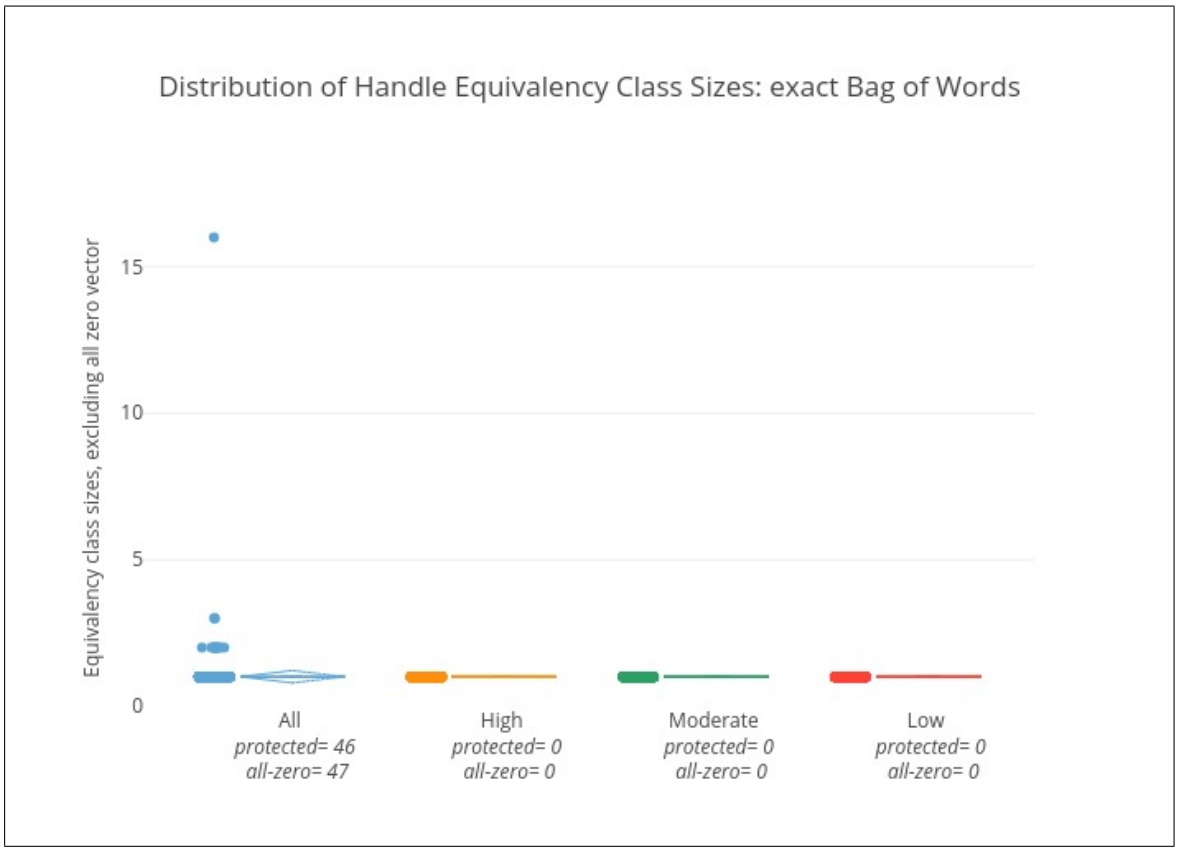

Figure 5.3: Exact Bag of Words

handle to be assigned an engagement level. This projection exhibits no variation in size between the equivalence classes.

In Figure 5.4 we compare the level of privacy for users who tweet similar amounts for our *Bag of Words* projection, where equivalence class membership is computed by binary comparison of feature vectors. The size 16 equivalence class for both binary and exact full bucket of handles is comprised of handles with only one tweet in their tweet stream - each of these tweets contained only one feature after preprocessing - the feature 'rt'. Other equivalence classes of size greater than 1 for the *Bag of Words*

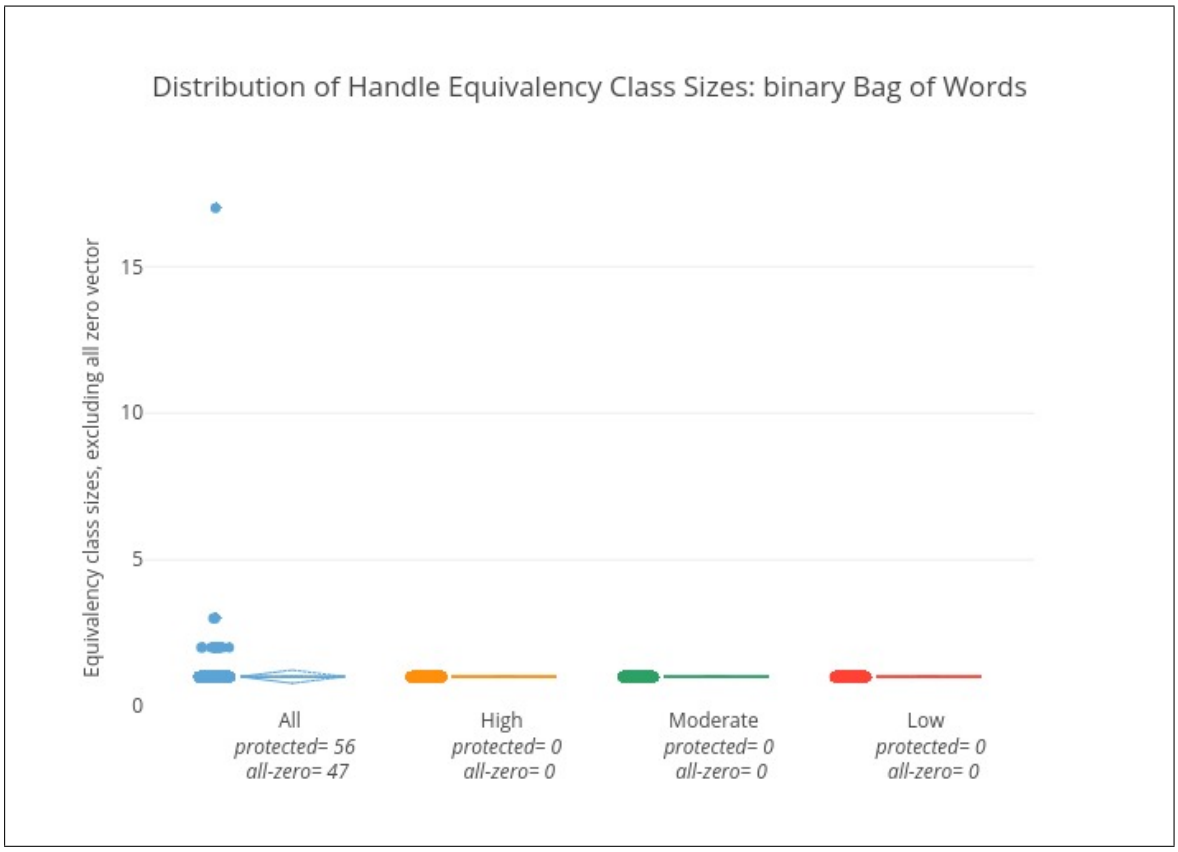

Figure 5.4: Binary Bag of Words

projection are explained by similarly small tweet streams containing few features after preprocessing. The tweet streams in our baseline *Bag of Words* projection exhibit low privacy, as there are very few handles that share the same vector representation of their tweet streams. There are 56 handles in equivalence classes of size greater than 1 for the full handles set that contain more than 1 handle for the full handles set that contain more than 1 handle, and no handles in equivalence classes of size greater than 1 for the full handles set that contain more than 1 handle for the handle engagement levels. This projection exhibits no variation in size between the equivalence classes.

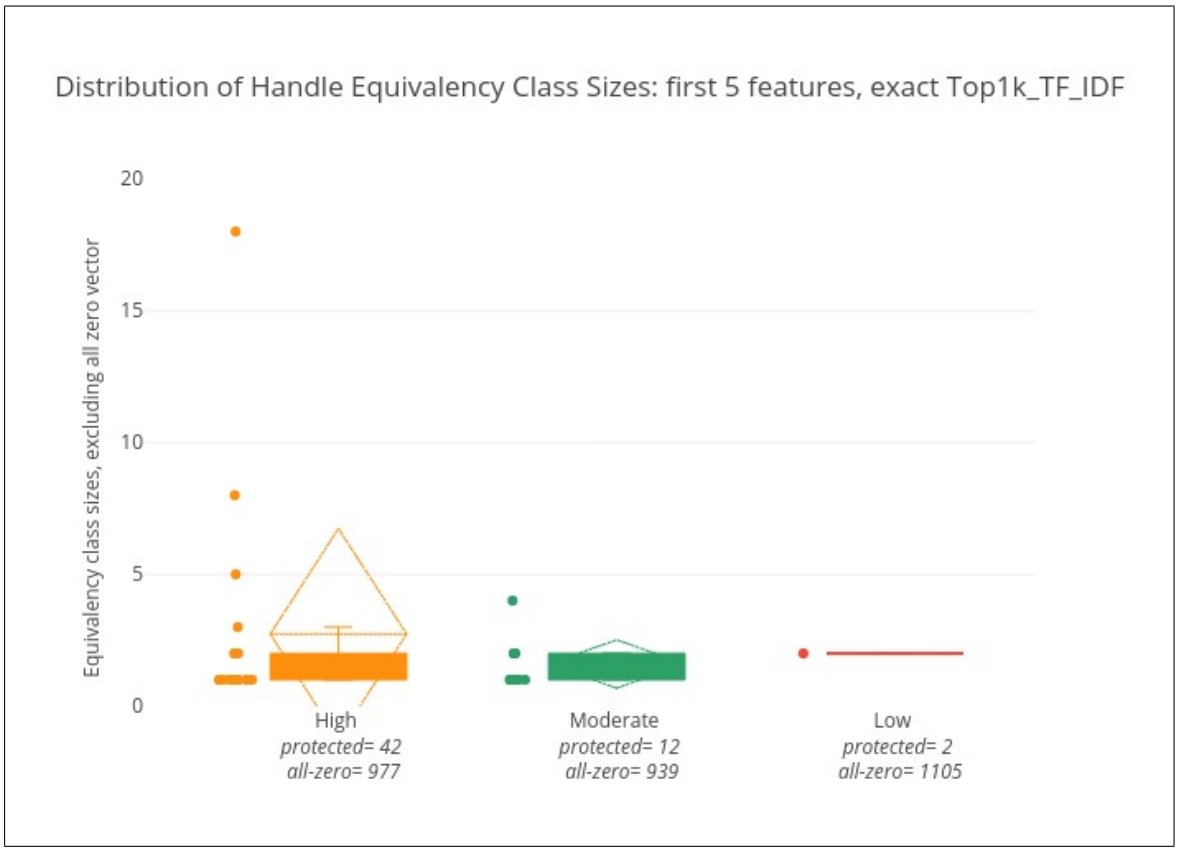

Figure 5.5: Exact Top 1k TF-IDF, Top 5 features

In Figure 5.5 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of the top 5 most important features of our importance projection, where equivalence class membership is computed by exact comparison of feature vectors. There are 42 handles in equivalence classes of size greater than 1 for the high tweet frequency handle bucket, 12 such handles for the moderate tweet frequency handle bucket, and 2 such handles the low tweet frequency handle bucket. The low average degree of privacy of handles represented by our *Top 5 TF-IDF* projection can be explained by the feature space being used to represent them is

comprised of the top five most distinguishing features across the set of tweet streams in our dataset. Not only do a large number of handles lack a single occurrence of these distinguishing features in their tweet streams, as can be seen by the high counts of all-zero vectors for each engagement level (remember that are approximately 1,000 handles for each engagement level), but those who do seldom possess the same set of these distinguishing feature as other handles. This projection exhibits relatively high variation in size between the equivalence classes for high engagement handles.

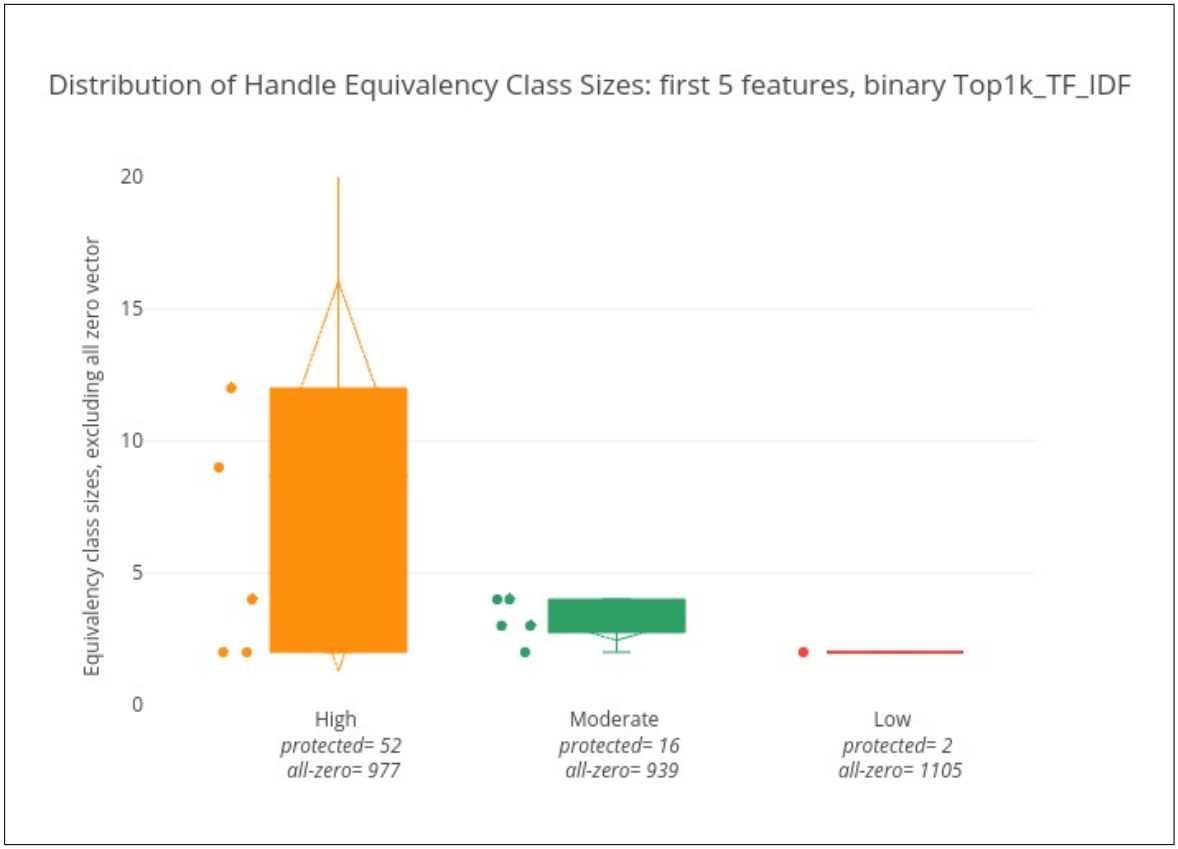

Figure 5.6: Binary Top 1k TF-IDF, Top 5 features

In Figure 5.6 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of the top 5 most important features of our importance projection, where equivalence class membership is computed by binary comparison of feature vectors. There are 52 handles in equivalence classes of size greater than 1 for

the high tweet frequency handle bucket, and 16 such handles for the moderate tweet frequency handle bucket, and 2 such handles for the low tweet frequency handle bucket. The low impact that a binary comparison had on the privacy of handles represented with this projection, as compared to the exact comparison results in Figure 5.5, is explained by most protected handles having only one occurrence of the features used to represent them with this projection, such that only a equivalence class sizes increase a small amount when equivalence classes are evaluated by binary, instead of exact comparison of feature vectors. This projection exhibits relatively high variation in size between the equivalence classes for high engagement handles.

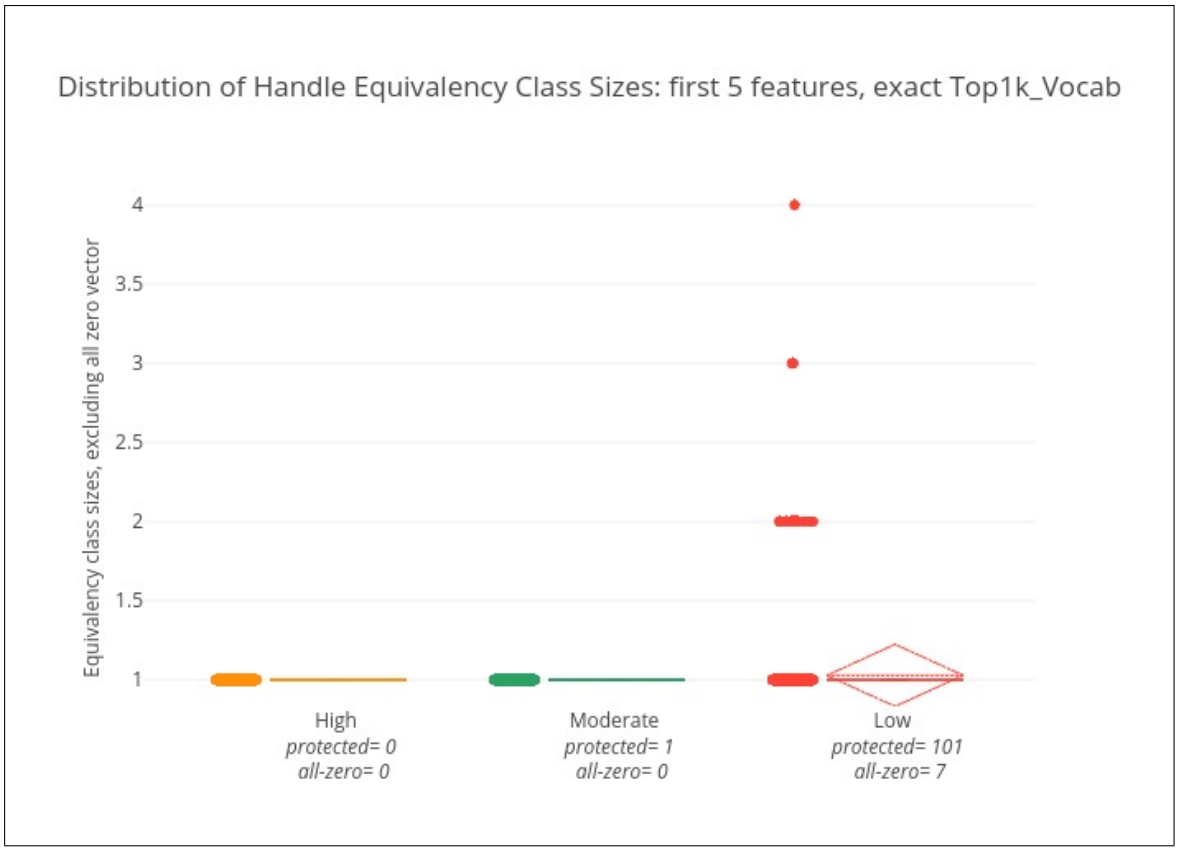

Figure 5.7: Exact Top 1k Vocab, Top 5 features

In Figure 5.7 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of the top 5 most important features of our frequency

projection, where equivalence class membership is computed by exact comparison of feature vectors. Equivalence class sizes for handles with a non-all-zero vector projection is shown on the y-axis. There are 52 protected handles for the high tweet frequency handle bucket, 16 protected handles for the moderate tweet frequency handle bucket, and 2 protected handles for the low tweet frequency handle bucket. Because low engagement handles have fewer tweets in their tweet streams, features are likely to occur at a low frequency. We accordingly observe low engagement handles belonging to equivalence classes of size greater than 1, as these handles correspond to tweet streams represented by feature values with low variance, as compared to those of the higher engagement levels. This projection exhibits variation in size between the equivalence classes for low engagement handles.

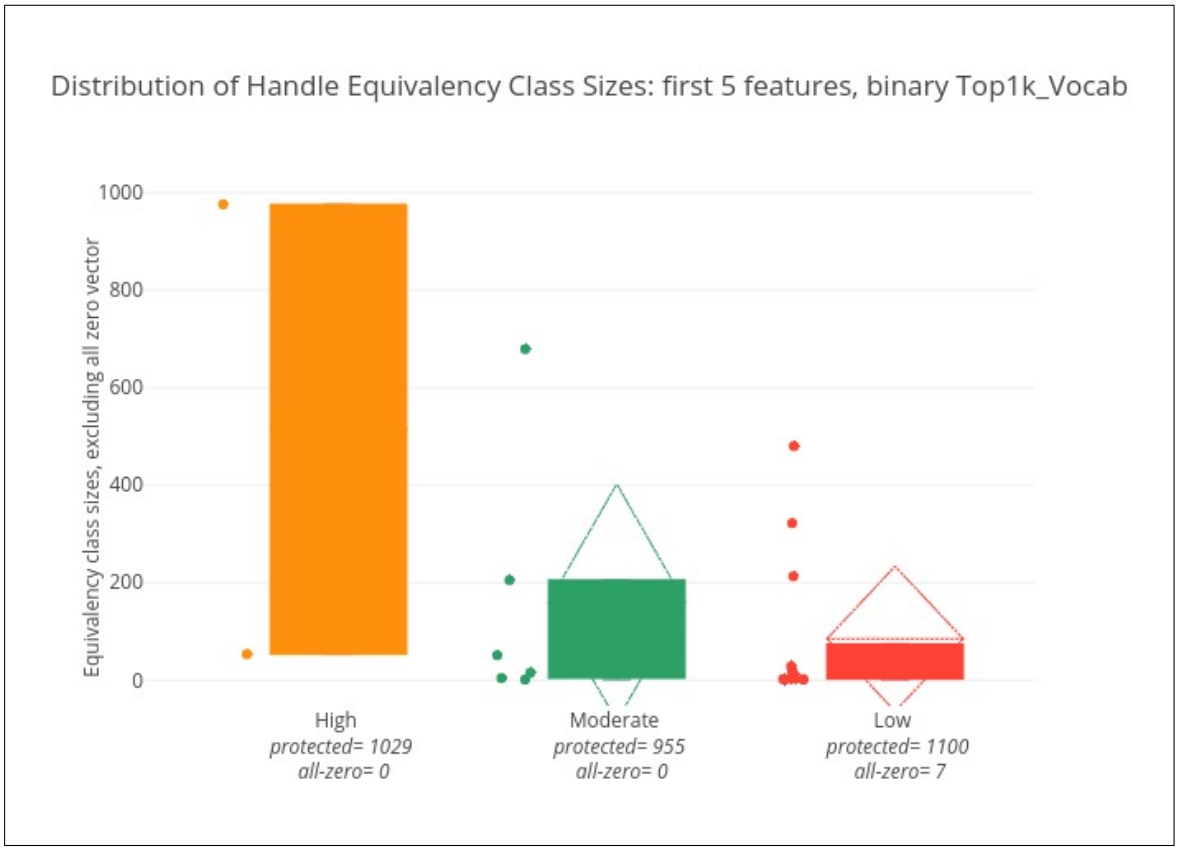

Figure 5.8: Binary Top 1k Vocab, Top 5 features

In Figure 5.8 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of the top 5 most important features of our frequency projection, where equivalence class membership is computed by binary comparison of feature vectors. There are 1029 protected handles for the high tweet frequency handle bucket, 955 protected handles for the moderate tweet frequency handle bucket, and 1100 protected handles for the low tweet frequency handle bucket. We observe that more engagement corresponds to higher average k for binary comparisons of tweet streams represented by this Importance projection with a reduced feature space. This projection exhibits relatively high variation in size between the equivalence classes across all engagement levels.

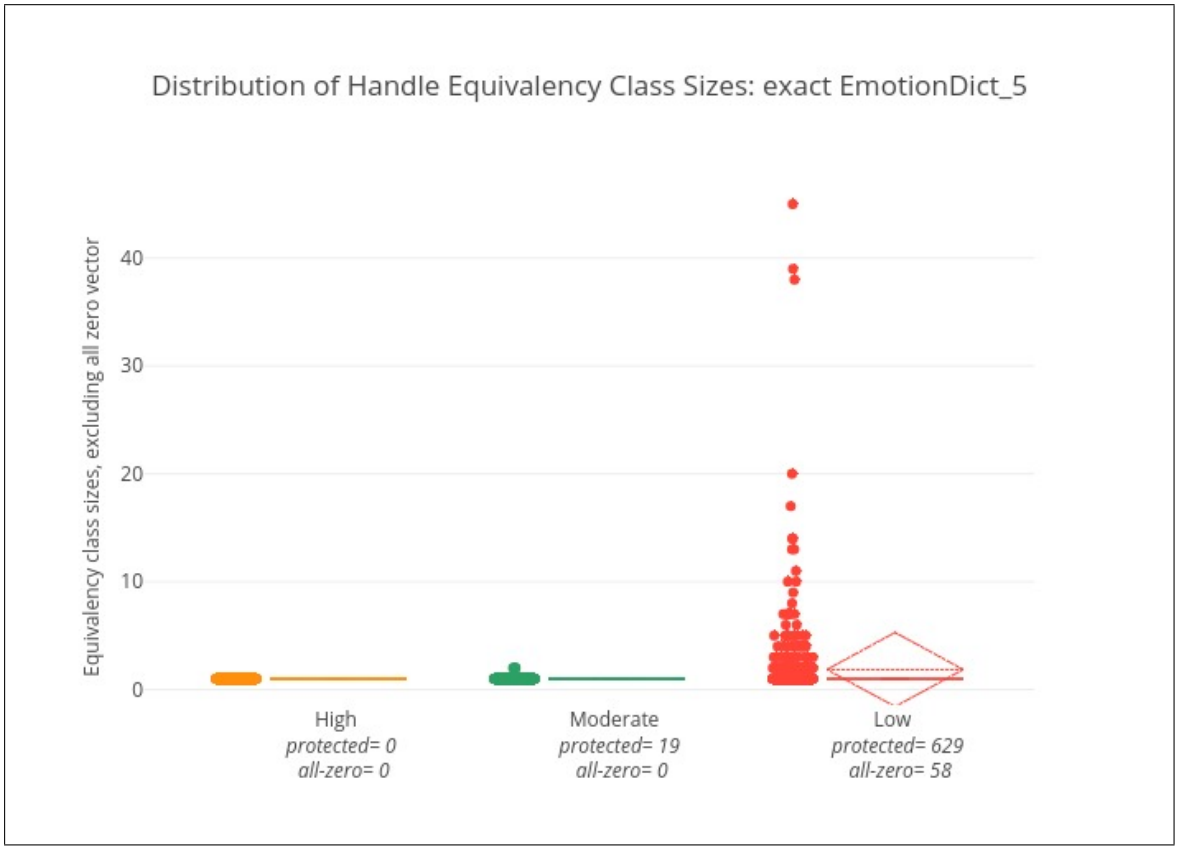

Figure 5.9: Exact Emotion Dictionary of 5 Emotions

In Figure 5.9 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of (*sadness_ emotionality*, *joy_ love_ emotionality*, *anger_ disgust_ emotionality*, *surprise_ emotionality*, and *fear_ emotinality*), the categories of emotion represented by features in our full Emotion projection. For this figure, equivalence class membership is computed by exact comparison of feature vectors. There are 0 protected high engagement handles, 19 protected moderate engagement handles, and 629 protected low engagement handles. As was also the case in Figure 5.7, because low engagement handles have fewer tweets in their tweet streams, features are likely to occur at a low frequency. We accordingly observe low engagement handles belonging to equivalence classes of size greater than 1, as these handles correspond to tweet streams represented by feature values with low variance, as compared to those of the higher engagement levels. This projection exhibits relatively variation in size between the equivalence classes for low engagement handles.

In Figure 5.10 we compare the level of privacy for users who tweet similar amounts for a reduced feature space of (*sadness_ emotionality*, *joy_ love_ emotionality*, *anger_ disgust_ emotionality*, *surprise_ emotionality*, and *fear_ emotinality*), the categories of emotion emotion encompassed by features in our Emotion projection. For this figure, equivalence class membership is computed by exact comparison of feature vectors. There are 1029 protected high engagement handles, 955 protected moderate engagement handles, and 1048 protected low engagement handles bucket. The observed gains in privacy, achieved when comparing feature vectors on binary terms instead of exact, particularly for the high and moderate engagement levels, can be explained by variance in feature occurrence no longer contributing to the uniqueness of the tweet streams corresponding to handles. This projection exhibits relatively high variation in size between the equivalence classes for high engagement handles, some for moderate engagement handles, but little for low engagement handles.

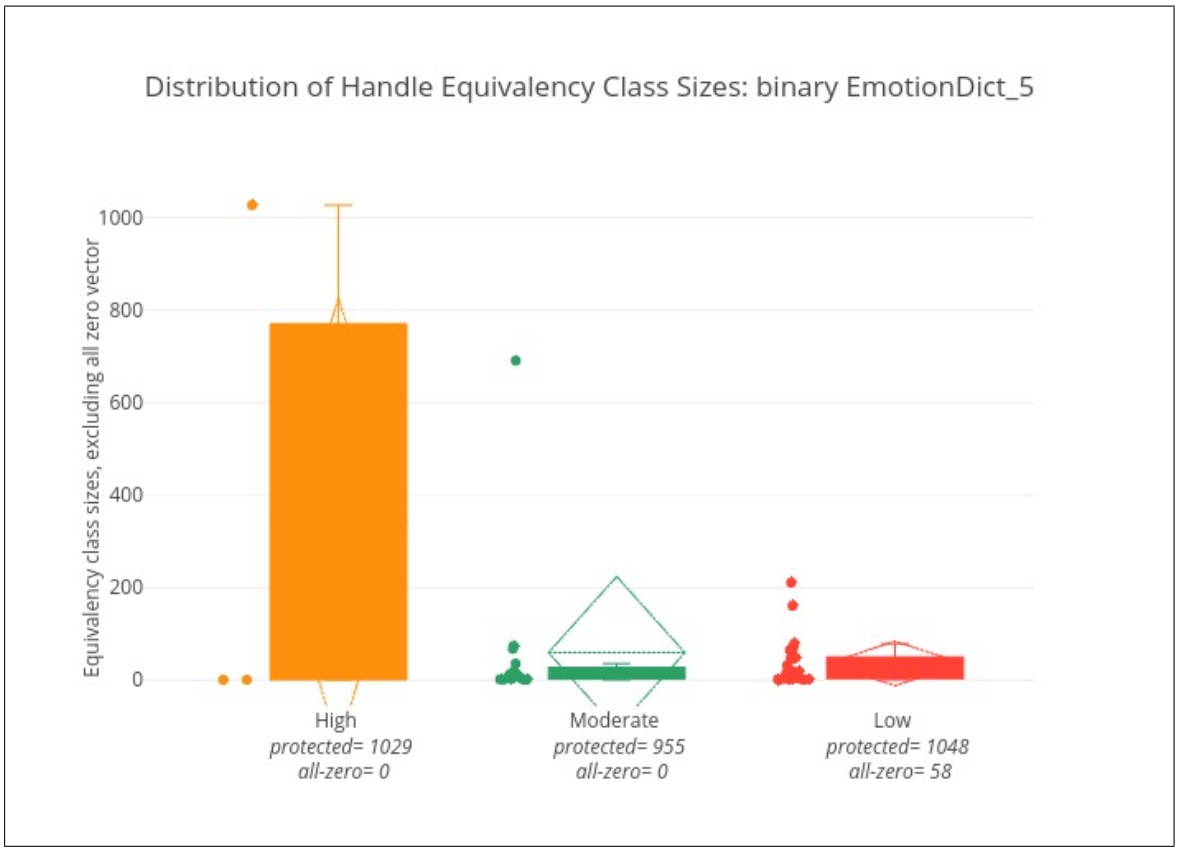

Figure 5.10: Binary Emotion Dictionary of 5 Emotions

5.4.3 Results - Zero-Valued Feature Incidence

Because of the potential sparseness of tweet streams represented as feature vectors across our four feature spaces, we computed the incidence of zero-valued features for all tweet stream vectors for each projection. It is worth noting that because two tweet stream vectors cannot belong to the same equivalence class for both binary and exact comparison of their feature vectors if they do not share the same set of zero-valued features, it is not possible for two handles to belong to the same equivalence class unless they have the same number of zero-valued features. Accordingly, large

groupings of tweet stream vectors at the same zero-valued feature incidence levels indicate greater likelihood of handles belonging to the same equivalence class for a given projection. For the following graphs, we compute the incidence of zero-valued features across the handles in our dataset, as well as for each of our engagement level categories. Because low engagement handles have fewer words in their tweet streams, we expected a greater incidence of zero-valued features for this category of handles.

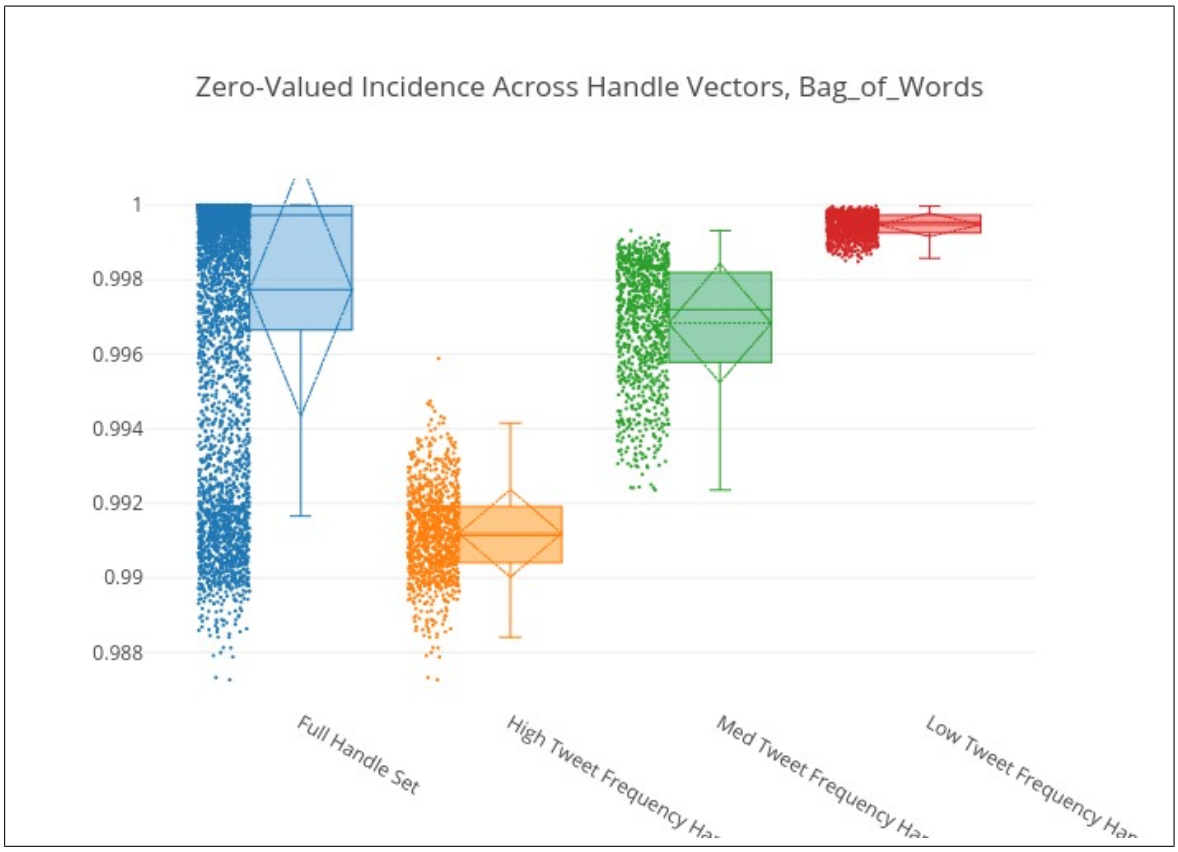

Figure 5.11: Zero-Valued Feature Incidence - Bag of Words

In Figure 5.11 we show the proportion of zero-valued features to size of feature space for each user represented by our Bag of Words projection. The y-axis shows the distribution of zero-valued featured incidence for each user. For each engagement level, the distribution of per-handle zero-value feature incidence is shown by a box-and-whisker plot. Additionally, each per-handle zero-value feature incidence value is represented by the set of dots adjacent to each box-and-whisker plot, with the zero-value feature incidence of each handle's feature vector represented by the dot's relation to the y-axis.

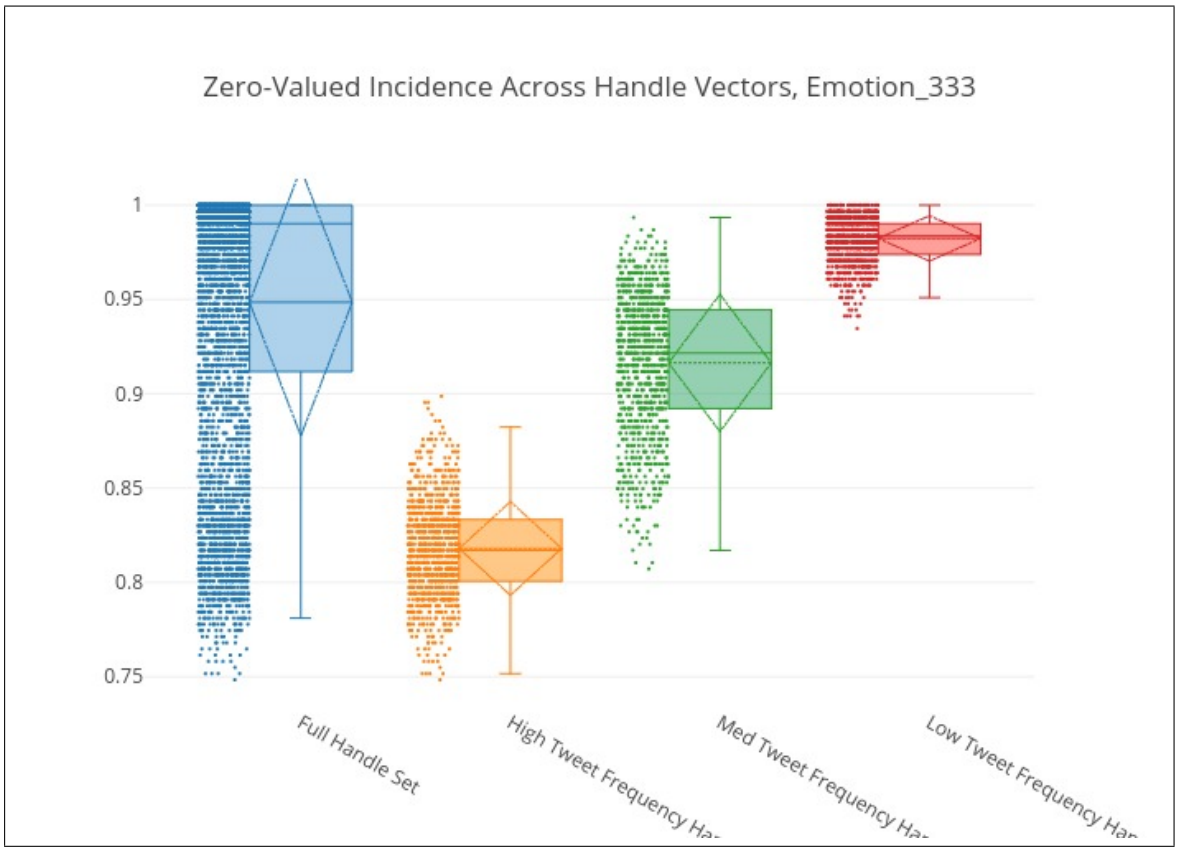

Figure 5.12: Zero-Valued Feature Incidence - Emotion 333

In Figure 5.12 we show the proportion of zero-valued features to size of feature space for each user represented by our Emotion projection. The y-axis shows the distribution of zero-valued featured incidence for each user. For each engagement level, the distribution of per-handle zero-value feature incidence is shown by a box-and-whisker plot. Additionally, each per-handle zero-value feature incidence value is represented by the set of dots adjacent to each box-and-whisker plot, with the zero-value feature incidence of each handle's feature vector represented by the dot's relation to the y-axis.

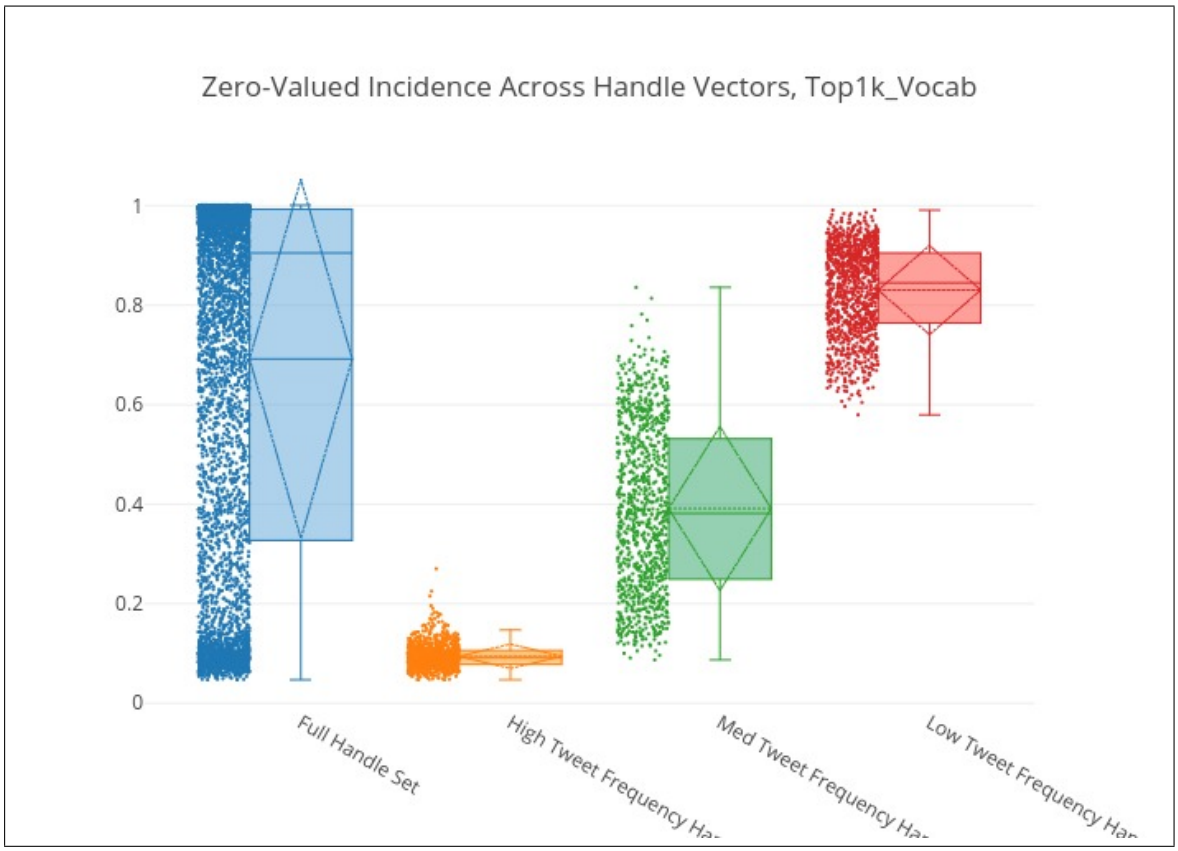

Figure 5.13: Zero-Valued Feature Incidence - Top1k Vocab

In Figure 5.13 we show the proportion of zero-valued features to size of feature space for each user represented by our Frequency projection. The y-axis shows the distribution of zero-valued featured incidence for each user. For each engagement level, the distribution of per-handle zero-value feature incidence is shown by a box-and-whisker plot. Additionally, each per-handle zero-value feature incidence value is represented by the set of dots adjacent to each box-and-whisker plot, with the zero-value feature incidence of each handle's feature vector represented by the dot's relation to the y-axis.

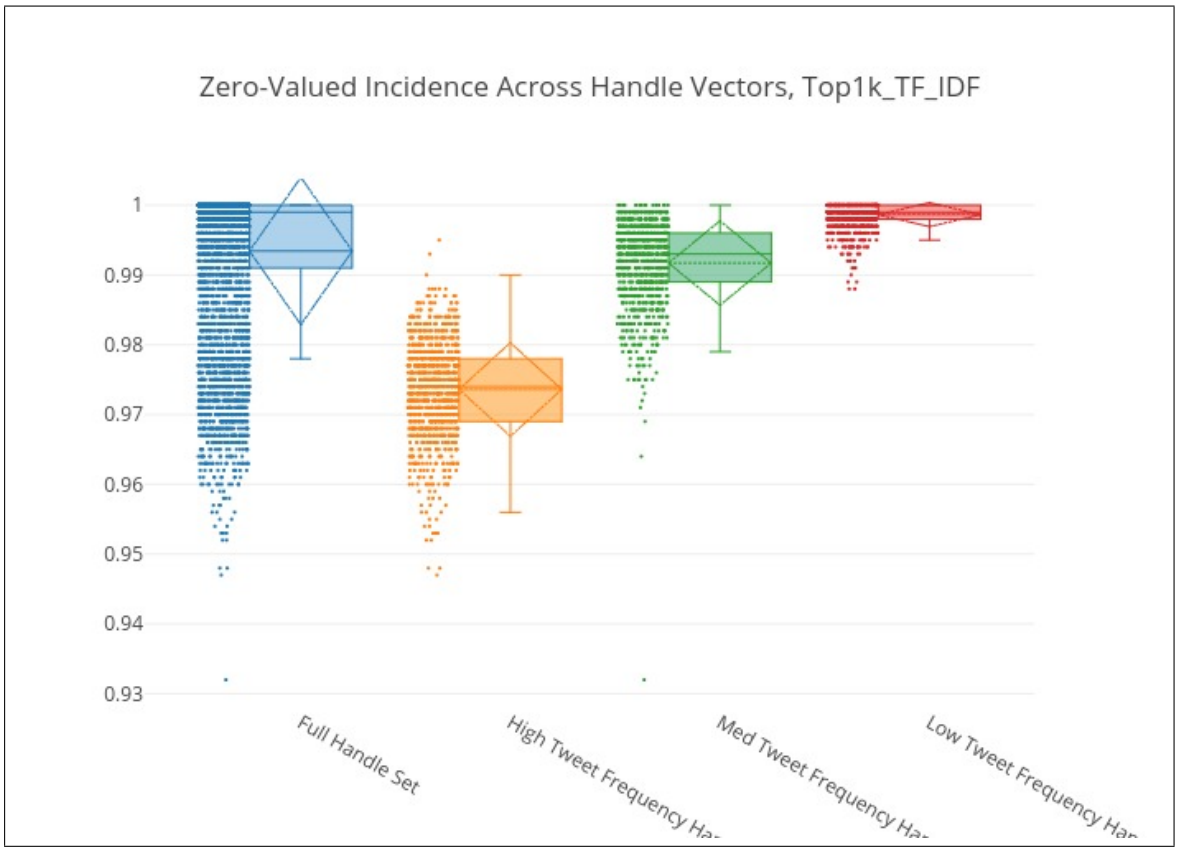

Figure 5.14: Zero-Valued Feature Incidence - TF-IDF

In Figure 5.14 we show the proportion of zero-valued features to size of feature space for each user represented by our Importance projection. The y-axis shows the distribution of zero-valued featured incidence for each user. For each engagement level, the distribution of per-handle zero-value feature incidence is shown by a box-and-whisker plot. Additionally, each per-handle zero-value feature incidence value is represented by the set of dots adjacent to each box-and-whisker plot, with the zero-value feature incidence of each handle's feature vector represented by the dot's relation to the y-axis.

From Figures 4.11-14, we note the following observations. Tweet streams represented by the baseline *Bag of Words* projection have the highest incidence of zero-valued features. This makes sense because tweet streams represented by this projec-

tion are projected as feature vectors for a feature space of size 767,937 - the number of unique words, plus emoji and orthographic features. We also observe high zero-valued incidence for tweet streams represented by *Top1k TF-DF*, a projection that represents tweet streams according to the top 1k most distinguishing words. Although many of the tweet streams represented by *Emotion 333* also exhibit high zero-valued feature incidence, the greater variance of zero-valued feature incidence observed relative to the previous two projections can be explained: (1) high-occurring features in the *Emotion 333* feature space and (2) the smaller size of the *Emotion 333* feature space means that there are fewer features to be valued as zero. As expected, the zero-incidence for *Top1k Vocab* exhibits the least zero-incidence, as the *Top1k Vocab* is comprised of the top 1k most frequently occurring words across the tweet streams in our dataset.

5.5 Computation of Utility

We evaluate the utility of our different transformations on the performance of 3 categories of unsupervised data mining tasks (clustering, frequent itemset mining, and anomaly detection) on a set of feature vectors. We use the *K Means* technique as our clustering task, the *BitDrill* algorithm for our frequent itemset mining task, and the *Local Outlier Factor* method as our anomaly detection task.

To evaluate the effect on utility that a projection has for our *K Means* clustering task, we first compute the number of handles with the same cluster label in both the projection being evaluated and the baseline, for each cluster. To normalize for number of clusters selected for the clustering task, we then divide each of these handle match counts by the size of cluster corresponding to each handle match count, sum the results and divide by the number of clusters. This results in a value in the range of 0 and 1,

with a 0 meaning no matches between the evaluated projection and the baseline, and a 1 meaning every handle sharing the same cluster between the evaluated projection and the baseline.

To evaluate the effect on utility that a projection has for our anomaly detection task, *Local Outlier Factor*, we first compute the local outlier factors, a value that indicates the abnormality of a sample, for each of our handles. We then compute the number of tweet streams that (1) have the same abnormality rank for a given projection and our baseline and (2) do not have the same abnormality rank for a given projection and our baseline. We then compute the Kendall's Tau correlation coefficient from these two values, such that a Kendall Tau coefficient of 1 would indicate a perfect match between the abnormality rankings of the baseline and the projection being evaluated, while a value of -1 would indicate a complete mismatch between the abnormality rankings.

We compute our frequent itemset mining task across the 4 million tweets in our dataset. To evaluate the effect on utility that a projection has for our anomaly detection task, Frequent Itemset Mining, we first compute 2 proportions: (i) the number of projection frequent itemsets present in the frequent itemsets of the baseline, divided by the total number of projection frequent itemsets (accuracy) and (ii) the number of baseline frequent itemsets present in the frequent itemsets of the projection, divided by the total number of baseline frequent itemsets (recall). Finally, we sum these 2 proportions and divide by two to obtain a value between the range of 0 and 1, with a 0 meaning the evaluated projection exhibits high utility loss relative to the baseline, and a 1 meaning the evaluated projection exhibits no utility loss relative to the baseline.

This equation is the average of two proportions: (i) proportion of frequent items sets in $S_{\hat{\omega}}$ from projection $\hat{\omega}$ present in *Bag of Words* frequent itemsets S_{ω} (preci-

sion) and (ii) proportion of *Bag of Words* frequent itemsets in S_ω present in frequent itemsets of projection $\hat{\omega}$ (recall). This computation results in a value between the range of 0 and 1, with a 0 meaning the evaluated projection exhibits high utility loss relative to the baseline, and a 1 meaning the evaluated projection exhibits no utility loss relative to the baseline.

5.5.1 Sensitivity Analysis

In order to obtain approximately-optimal parameter settings for *K Means*, a set of combinations for the *num_ clusters* parameters both *K Means* was evaluated for the baseline projection. We evaluated the 'optimality' of a given parameter combination in terms of 3 aspects of the clusters produced for each parameter-value combination: (1) silhouette_score - a measure of how similar an object is to its own cluster (i.e. cohesion) compared to other clusters (i.e. separation) and (2) the subjective similarity of feature vectors within each cluster, determined by the evaluation of the intersection of the feature space for the transformation and top words for users in a given cluster. Using this approach, we determined that the optimal parameters values, evaluated in terms of the baseline, were *num_ clusters*=8 for exact *K Means* and num_clusters=6 for binary *K Means*, with silhouette scores of 0.6547 and 0.4000, respectively - the highest silhouette scores observed in each case for a num_clusters value of above 3.

We tuned the Frequent Itemset Mining *BitDrill* algorithm by evaluating the itemsets returned across different *min_ support* parameter values. We sought a *min_ support* value that resulted in a list of frequent itemsets with (i) sets of size larger than 2 (ii) a number of itemsets around 100. We decided upon a *min_ support* = 1% that resulted in 114 itemsets for our baseline. We tried two approaches for addressing the potential loss of high-frequency features being lost in a non-baseline projection. The first involved setting the *min_ support* threshold for the frequent

itemset mining task proportional to the number of tweets that contained at least one feature in the feature space of the projection used. This approach resulted in 113 itemsets for our baseline (for a *min_ support* of 1% across all 4 million tweets), 24 for *Emotion 333* (for a *min_ support* of 1% across the 393959 tweets with at least one feature in the *Emotion 333* feature space), 1 for *Top 1k TF-IDF* (for a *min_ support* of 1% across the 228876 tweets with at least one feature in the *Top 1k TF-IDF* projection), and 138 sets for *Top 1k Vocab* (for a *min_ support* of 1% across the 3678455 tweets with at least one feature in the *Top 1k Vocab* projection). Because this first approach resulted in too few itemsets for a fair comparison of *Emotion 333* and *Top 1k TF-IDF* to the baseline according to our frequent itemset mining utility metric, we then computed the frequent itemsets and frequent itemset mining utility metrics for these two projections across a range of *min_ support* values, in order to find a value that would result in a number of frequent itemsets closer to that of our baseline (i.e. approximately 100 itemsets). For our *Emotion 333* projection, this *min_ support* value was .01% (across all 4 million tweets) and resulted in 44 frequent itemsets for the tweet streams represented by this projection. For our *Top 1k TF-IDF* projection, this *min_ support* value was .02% (across all 4 million tweets) and resulted in 92 frequent itemsets for the tweet streams represented by this projection. We used the same approach to compute utility metrics for the reduced feature space projections that achieve k-anonymous vector privacy measure: *Emotion Dict 5*, *Top 5 TF-IDF*, and *Top 5 Vocab*. This resulted in 59, 6, and 31 frequent itemsets respectively. We use the minimum support values arrived at by this second approach to compute the results presented in this section's Figures.

We tuned the *Local Outlier Factor* method *min_ number_ neighbors* across each of our for our baseline projection of tweet streams by evaluating different values for the *min_ number_ neighbors* (to consider a handle represented by a feature vector as

a non-outlier) parameter. This evaluation was undertaken by examining (1) *silhouette_score*, (2) the subjective quality of 'outlier' labels for feature vectors, determined by the evaluation of the intersection of the feature space for the transformation and top words for users in a given cluster, and (3) the number of feature vectors labeled as outliers. We decided on a *min_number_neighbors* of 20, as this resulted in highest quality 'outlier' labels across the range of *min_number_neighbors* values we evaluated.

5.5.2 Results

We provide utility metric results across our utility tasks in Table 1. Exact indicates the use of the counts of features as their value for a set of projection vectors. Binary means that all non-zero counts for each feature were converted to 1 before the utility task was performed.

Table 1. Utility Retention from Baseline

Utility Method \ Projection	Emotion_333	Top1k_TF_IDF	Top1k_Vocab
exact K Means	0.132	0.125	0.124
binary K Means	0.162	0.241	0.180
exact LOF	0.013	-0.006	0.011
binary LOF	0.014	-0.003	0.004
Method1 Freq Itemset Mining	0.000	0.000	0.814
Method2 Freq Itemset Mining	0.000	0.000	-

Table 1. Utility Retention from Baseline - Projection Set 2

Utility Method \ Projection	Emotion_Dict_5	Top5_TF_IDF	Top5_Vocab
binary K Means	0.1641	0.1657	0.1639
binary LOF	0.0061	0.0056	0.0210
Method2 Freq Itemset Mining	0.000	0.00	0.125

For our clustering tasks, only a small proportion of handles receive the same label when represented by our three projections as they did when represented according to our baseline. Our interpretation of this result is that these projections do not retain much data utility as compared to the baseline. A qualitative examination, however, of the tweet streams corresponding to the handles ascribed the same cluster labels across *K Means* clusters for our projections reveals that clusters of meaningful categories of handles are nonetheless being constructed. For our anomaly detection task, there was little similarity in the rankings of handles according to abnormality, when the representation of handles for our three projections was compared against that of the baseline. For our frequent itemset mining tasks, only our Frequency projection exhibits utility retention.

We also compute both utility retention metrics and k anonymous vector privacy measures for our three data mining tasks across a range of reduced feature space sizes for our projections, to evaluate tradeoffs between data mining utility and higher privacy. The results are shown in the following three figures. For these figures, the x-axis shows the utility retained from the baseline projection (i.e. our utility metric for abnormality detection). The y-axis displays the proportion of handles with privacy, where and individual is considered to be private if belonging to an equivalence class of k greater than 1. The blue line corresponds to our Importance projection, green our Frequency projection, and red our Emotion projection.

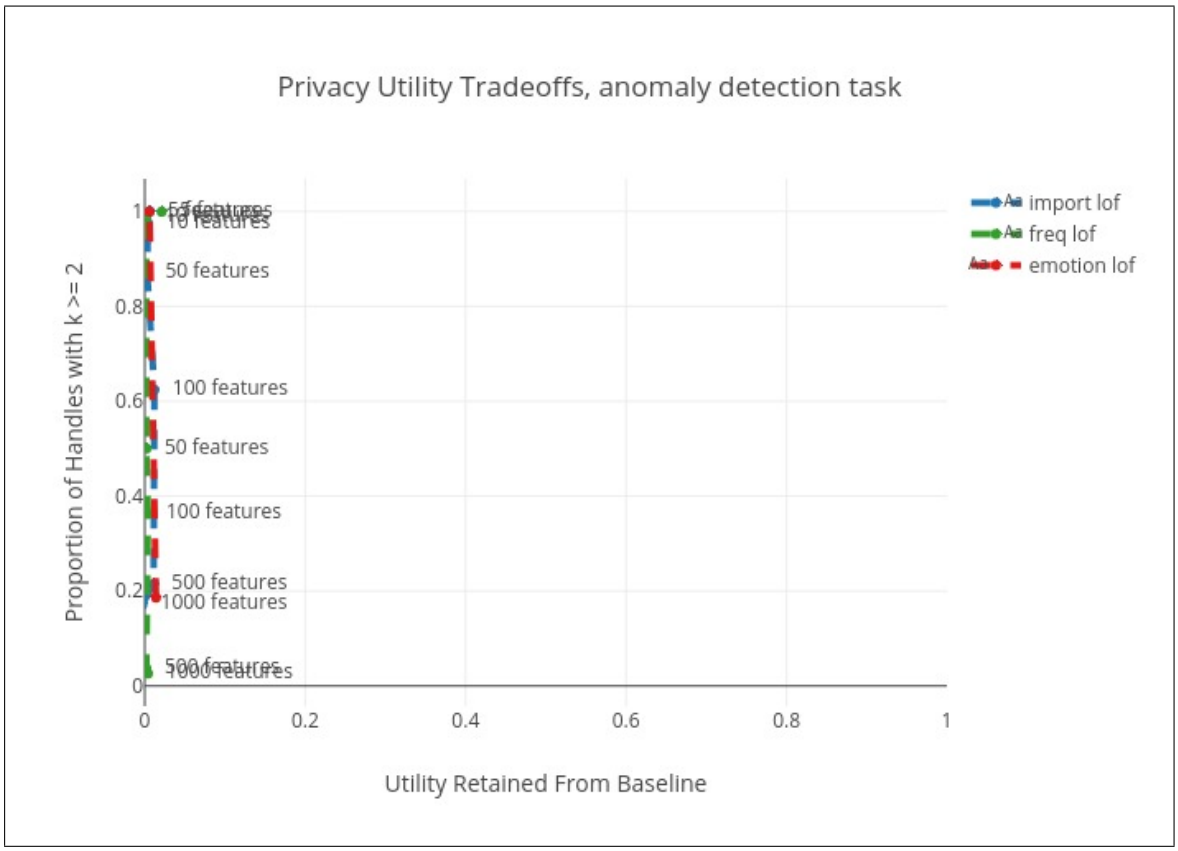

Figure 5.15: Privacy Utility Tradeoff - Anomaly Detection Task

Figure 5.15 shows the utility retained for our abnormality detection task (using the Local Outlier Factor technique) across a range of reduced feature space sizes for each of our three projections. It is not surprising that anomaly detection utility does not exist for our three projections, because these projections represent tweet streams according to features selected to capture semantic meaning. Accordingly, it can be said that we set our anomaly detection task up for failure.

Figure 5.16 shows the utility retained for our cluster task across a range of reduced feature space sizes for each of our three projections. Although not remarkable, utility

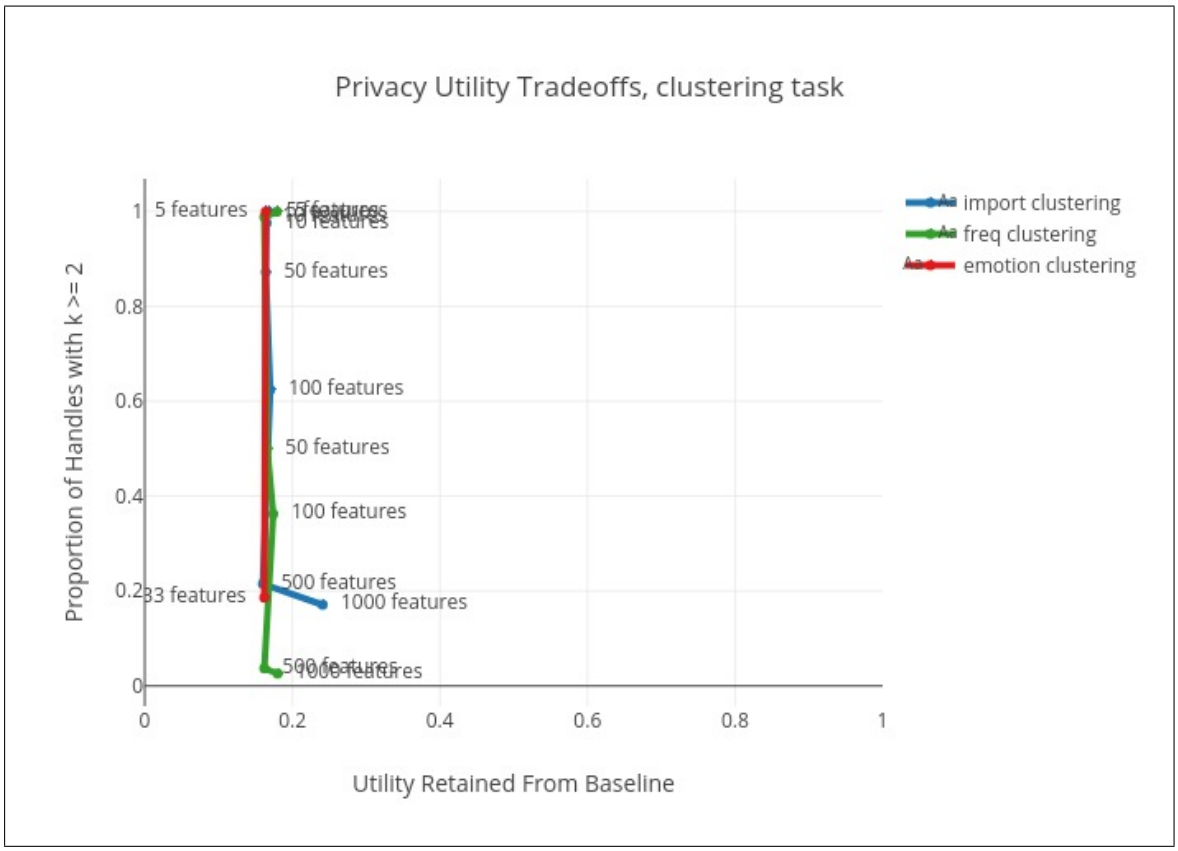

Figure 5.16: Privacy Utility Tradeoff - Clustering Task

retention is primarily 20% for all projections, across all feature space sizes. Although the Frequency projection of feature space size 1,000 retains comparatively more utility, less than 20% of handles represented in such a manner retain their privacy.

Figure 5.17 shows the utility retained for our frequent itemset mining task across a range of reduced feature space sizes for each of our three projections. Only our Frequency projection retains any utility for the set of feature space sizes evaluated. This is explained by the Frequency projection retaining the most frequently occurring features when representing tweet streams as feature vectors, features that can be

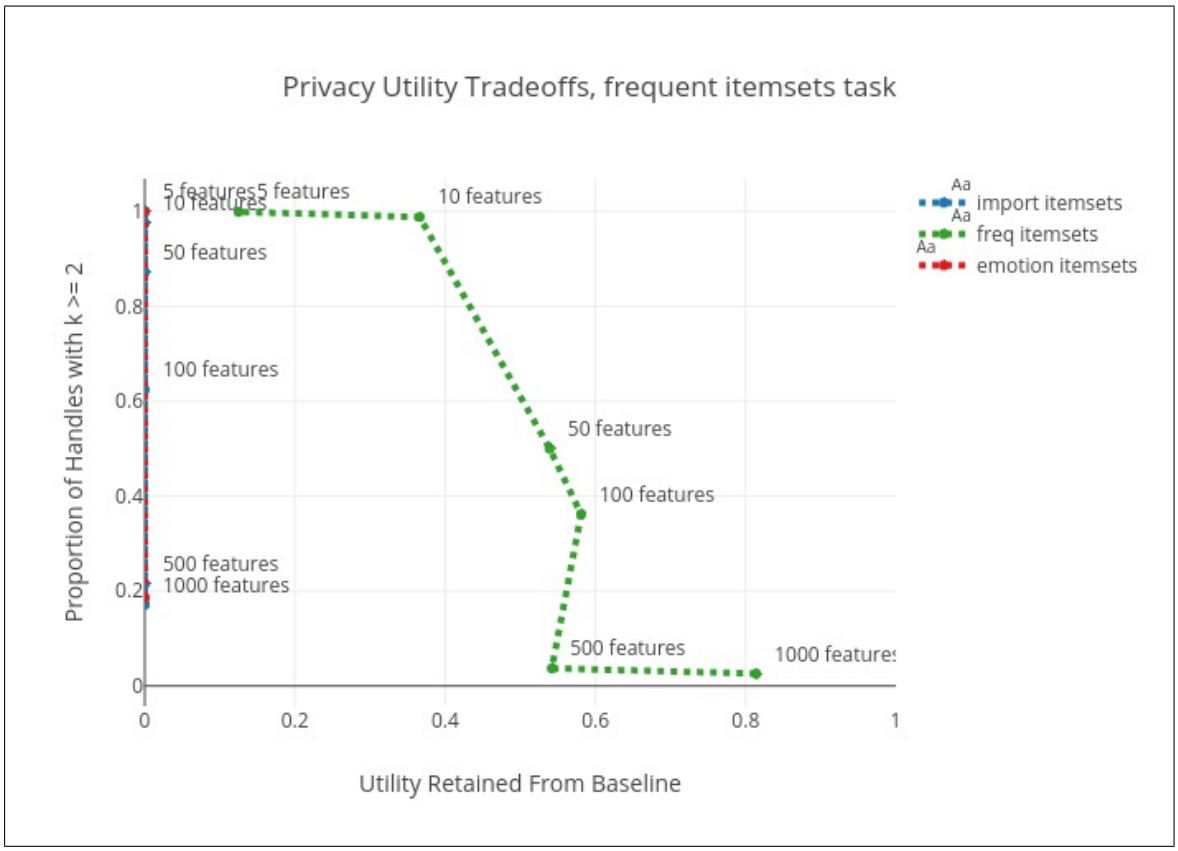

Figure 5.17: Privacy Utility Tradeoff - Frequent Itemset Mining Task

expected to exist in frequent itemsets. The Frequency projection with a feature space of size 1,000 retains a lot of utility, but only 3% of handles retain their privacy at this feature space size. For a feature space size of 10, however, 40% of utility is retained with almost complete privacy.

5.6 Computation of Distortion

We also evaluate the distortion introduced by our projections. We consider distortion to be the difference between a given projection and our baseline Bag of Words trans-

formation. Our interest in the distortion introduced by our transformations stems from the observation that the higher the distortion of a projection, the fewer the number potentially distinguishing features attributable to a handle. Distortion is calculated as follows: compute the number of non-zero valued features values across all tweet streams corresponding to handles in H lost when mapping each handle's tweet streams to $\hat{W}$ instead of W. This number is then divided by the total number of non-zero valued features in all tweet streams corresponding to handles in H, mapped to W to obtain fraction of non-zero valued features lost in the projection $\hat{W}$ of tweet streams from the baseline W. For $\omega_i \subseteq \omega$ & $\hat{\omega}_i \subseteq \hat{\omega}$, distortion $\mathbb{D}$ is defined as the normalized sum of the difference between ω_i and $\hat{\omega}_i$:

$$\mathbb{D} = \sum_{i=1}^{|H|} \frac{|binary(\omega_i) - binary(\hat{\omega}_i)|}{|W|}$$

Accordingly, a distortion rate of 1 corresponds to a loss of all non-zero valued features across the set of tweet streams in a dataset, when representing said tweet streams according to a given projection instead of the *Bag of Words* representation. A distortion rate of 0, on the other hand corresponds to the retention of all non-zero valued features across the set of tweet streams in a dataset, when representing said tweet streams according to a given projection instead of the *Bag of Words* representation.

Table 5.2 Projection Distortion Rates

Top1k_TF_IDF	Top1k_Vocab	Emotion_333	Top5_TF_IDF	Top5_Vocab	Emotion_Dict_5
0.9966	0.6054	0.9993	0.9997	0.9597	0.9995

We provide the distortion rates introduced by our three projections when using a large feature space, as well as when the feature space used to represent the tweet streams of handles is reduced in Table 5.2. Only our *Frequency* projection with a large feature space exhibits a distortion rate that is not approximately 1. This is because

the *Frequency* projection represents tweet streams according to the most frequently occurring features, such that the feature retained for this projection are those most likely to have non-zero values for each handle represented in this manner.

5.6.1 Results

Our *Top 1k TF IDF* and *Emotion 333* projections exhibit high distortions rates because they represent handles according to features that occur at a relatively low frequency across the tweet streams in our data set. Although the *Emotion 333* projection results in a slightly higher level of distortion than *Top1k TF IDF*, a projection that represents users with highly distinguishing feature (i.e. features that seldom occur across the set of tweet streams in our data set, but occur often across a few handles), this can be explained by the smaller size of the feature space (333 features) used for the *Emotion 333*. The *Top1k Vocab* projection results in comparatively less distortion because tweets are represented according to the top one thousand most frequently occurring features across the tweet streams in our dataset. This means that tweet streams represented with this projection will retain the features most likely to be present.

We find that transformations of tweet streams according to feature spaces of specific categories of features (substantive content discussed by an individual, the proportions of emotionality represented across and individual's tweets, and the frequency at which an individual engages on Twitter) result in low user privacy, unless small feature spaces are used in these projections. Because the representation of tweet streams according to such small feature spaces results in the dropping of such a large number of features, we would expect a high loss of data utility as a result. For exact *K Means*, as an example, the 8 clusters seem to encompass the following categories of handles (as evaluated by an examination of the top features in the tweet streams for handles

with the same cluster labels): (i) promotional handles (exhibiting many features such as '*win*', '*exclamation*', '*giveaway*', '*chance*', and '*enter*') (ii) handles that frequently engage with the topic of parenting (exhibiting many features such as '*charity*', '*child*', '*support*', '*disney*', '*love*', and '*parenting*') (iii) handles with few tweets in their tweet streams (the largest cluster, with a size of 3876) and (iv) positive handles (exhibiting many features such as '*heart_ emoji*', '*happy*', '*well*', '*great*', '*smile_face*', '*beautiful*', and '*awesome*') and (v) handles frequently engage in substantive content domains (exhibiting many features such as '*america*', '*community*', '*country*', '*trumps*,' '*trump*', and '*healthcare*'). In this sense, it can be said that even after our projections, it seems to be the case that handles can be distinguished according to meaningful categories.

Chapter 6

Conclusion

6.1 Contributions

The contributions of this thesis are as follows: (i) we contribute a novel analysis of social media distinguishability (ii) propose a framework for analyzing the privacy-utility tradeoff for social media text posts in the context of traditional data mining tasks and (ii) we conduct an empirical analysis on 5626 Twitter users of 4 million tweets and show that users are only private if represented by a small number of features and that a high degree of data utility is lost across our projections of tweet streams for anomaly detection, frequent itemset mining, and clustering tasks. Our analysis has shown that it is difficult to obtain k anonymity privacy measure of 1 for tweet streams representing Twitter users without a high loss of data utility. We also observed that feature space size had little or no impact on utility for all projections, besides frequent itemset mining for tweets represented by the Frequency projection. Finally, because the relationship between utility and privacy is very task dependent, a general solution should not be expected. Instead, projection selection should be task dependent when data utility and individual privacy is a concern.

6.2 Future Work

The approach to evaluating the tradeoffs between privacy and utility of different representations of the public tweet streams of Twitter users describe in this thesis could

be extended in future work in a few directions. One such direction would be a better understanding of the types of transformations (including mathematical transformations) that help maintain any level of utility for specific data mining tasks. Because of the sparseness of our projections using full feature spaces, applying mathematical transformations that compress the tweet streams in our data set has the potential to increase the privacy of represented handles with little impact on data utility. Given the high rate at which Twitter users retweet, we are also interested in how the frequency a user publishes tweets from others in their tweet stream impacts the privacy. In this work, We have learned that common semantic projections are not sufficient for maintaining privacy and utility - future work will need to consider more sophisticated projections.

Bibliography

[1] Why the cambridge analytica scandal is a watershed moment for social media, Mar 2018. URL http://knowledge.wharton.upenn.edu/article/fallout-cambridge-analytica/.

[2] Jonathan Albright. Cambridge analytica: the geotargeting and emotional data mining scripts, Oct 2017. URL https://medium.com/tow-center/cambridge-analytica-the-geotargeting-and-emotional-data-mining-scripts-bcc3c428d77f.

[3] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM, 2000.

[4] Keith Collins and Gabriel J. X. How researchers learned to use facebook 'likes' to sway your thinking, Mar 2018. URL https://www.nytimes.com/2018/03/20/technology/facebook-cambridge-behavior-model.html?action=click&contentCollection=Europe&module=RelatedCoverageÂőion=Marginalia&pgtype=article.

[5] Peter Sheridan Dodds, Eric M. Clark, Suma Desu, Morgan R. Frank, Andrew J. Reagan, Jake Ryland Williams, Lewis Mitchell, Kameron Decker Harris, Isabel M. Kloumann, James P. Bagrow, Karine Megerdoomian, Matthew T. McMahon, Brian F. Tivnan, and Christopher M. Danforth. Human language

reveals a universal positivity bias. *Proceedings of the National Academy of Sciences*, 112(8):2389–2394, 2015. ISSN 0027-8424. doi: 10.1073/pnas.1411678112. URL http://www.pnas.org/content/112/8/2389.

[6] R. N. Horspool and G. V. Cormack. Constructing word-based text compression algorithms. In *Data Compression Conference, 1992.*, pages 62–71, March 1992. doi: 10.1109/DCC.1992.227475.

[7] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998.

[8] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):881–892, 2002.

[9] Tiancheng Li and Ninghui Li. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 517–526, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9. doi: 10.1145/1557019.1557079. URL http://doi.acm.org/10.1145/1557019.1557079.

[10] Alexis C. Madrigal. What took facebook so long?, Mar 2018. URL https://www.theatlantic.com/technology/archive/2018/03/facebook-cambridge-analytica/555866/.

[11] Andrew McCallum, Kamal Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998.

[12] A Ian McLeod. Kendall rank correlation and mann-kendall trend test. *R Package Kendall*, 2005.

[13] Nasser M Nasrabadi. Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4):049901, 2007.

[14] E. Ozkural, B. Ucar, and C. Aykanat. Parallel frequent item set mining with selective item replication. *IEEE Transactions on Parallel and Distributed Systems*, 22(10):1632–1640, Oct 2011. ISSN 1045-9219. doi: 10.1109/TPDS.2011.32.

[15] Ashequl Qadir and Ellen Riloff. Bootstrapped learning of emotion hashtags #hashtags4you, 2013. URL `https://www.semanticscholar.org/paper/Bootstrapped-Learning-of-Emotion-Hashtags-Qadir-Riloff/454a04f532bc32b2af286fc8df900a3c6d9ae040`.

[16] L. Singh, G. H. Yang, M. Sherr, A. Hian-Cheong, K. Tian, J. Zhu, and S. Zhang. Public information exposure detection: Helping users understand their web footprints. In *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 153–161, Aug 2015. doi: 10.1145/2808797.2809280.

[17] Twitter. Twitter's safety and security policy page. *Twitter - Safety and Security*, Feb 2013. doi: 10.2172/1093797. URL `https://help.twitter.com/en/safety-and-security/data-through-partnerships`.

[18] Ian Witten, Timothy Bell, Alistair Moffat, Craig Nevill-Manning, Cowlemon Tony, and Harold Thimbleby. Semantic and generative models for lossy text compression. 37, 07 2000. URL `https://www.researchgate.net/publication/2634573_Semantic_and_Generative_Models_for_Lossy_Text_Compression`.

[19] Changhua Yang, Kevin Hsin-Yih Lin, and Hsin-Hsi Chen. Building emotion lexicon from weblog corpora. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 133–136, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=1557769.1557809`.