You Can Yak but You Can't Hide: Localizing Anonymous Social Network Users

Minhui Xue[†‡], Cameron Ballard[‡], Kelvin Liu[‡], Carson Nemelka[‡], Yanqiu Wu[‡],
Keith Ross[‡♮][*], and Haifeng Qian[†]
[†]East China Normal University (ECNU), Shanghai, China
[‡]New York University Shanghai (NYU Shanghai), Shanghai, China
[♮]New York University (NYU), New York, USA

ABSTRACT

The recent growth of anonymous social network services – such as 4chan, Whisper, and Yik Yak – has brought online anonymity into the spotlight. For these services to function properly, the integrity of user anonymity must be preserved. If an attacker can determine the physical location from where an anonymous message was sent, then the attacker can potentially use side information (for example, knowledge of who lives at the location) to de-anonymize the sender of the message.

In this paper, we investigate whether the popular anonymous social media application Yik Yak is susceptible to localization attacks, thereby putting user anonymity at risk. The problem is challenging because Yik Yak application does not provide information about distances between user and message origins or any other message location information. We provide a comprehensive data collection and supervised machine learning methodology that does not require any reverse engineering of the Yik Yak protocol, is fully automated, and can be remotely run from anywhere. We show that we can accurately predict the locations of messages up to a small average error of 106 meters. We also devise an experiment where each message emanates from one of nine dorm colleges on the University of California Santa Cruz campus. We are able to determine the correct dorm college that generated each message 100% of the time.

Keywords

Localization Attack; Machine Learning Inference; Anonymous Social Networks; Yik Yak

1. INTRODUCTION

With approximately two million active users, Yik Yak combines smartphone location-based services with anonymity to create a unique social network experience. Yik Yak allows a user to post an anonymous short message, called a "yak," which can be seen by anyone else who has the smartphone application and is nearby. This location-based feature allows Yik Yak to foster anonymous discussions that are relevant to specific geographic communities, such as college campuses. The anonymity allows users to express whatever they want without having to fear other people's reactions. This in principle allows for fully open discussion, and facilitates the sharing of controversial ideas specific to the local community.

Yik Yak enjoys enormous popularity on US college and university campuses [16]. Yik Yak is also controversial often in the press [12]. There have been yaks calling people out by name and spreading nasty rumors about them. There have been messages discussing others' sexuality crudely, insulting professors including ridiculing their personal appearance. Students from several different campuses have even made prank threats of mass violence on the Yik Yak application [1, 4, 6].

When using Yik Yak, Alice does not know from where on campus the message originated – unlike Whisper, the Yik Yak application does not even report to Alice the distances between her and the message origins. Alice only knows that the message originated from somewhere on the campus or nearby the campus. However, if Alice could determine from where a message originates, she could then potentially de-anonymize the author of the message. For example, suppose Alice can somehow determine that a message originated from a specific dorm room. If Alice also has access to a campus directory, then by knowing the dorm room she can reduce the number of possible authors to a small number (to one for a single room). Even if the yaks can only be localized to large buildings, the yaks can potentially be de-anonymized. For example, suppose a yak criticizes a professor in a specific class. If the professor can determine the dormitory from which the yak originated, then she can take the intersection of the list of students who live in that dorm with the list of students in her class and narrow down the sender of the yak, again possibly to one person.

In this paper, we show how Yik Yak is susceptible to localization attacks, whereby an attacker can determine the approximate origin of all messages posted in a region such as a college campus. As described above, such a localization attack can put user anonymity at risk. The problem is challenging because (i) Yik Yak does not report any distance information about the yaks. (If distance information were available, then trilateration could be used to localize yaks [25].); (ii) the algorithm Yik Yak employs for deciding which messages to display to Alice is unknown; (iii) for recent releases of Yik Yak, it appears difficult to reverse engineer the protocol or employ man-in-the-middle attacks to read messages off the wire. Our localization attack has two major components: (i) remote data collection, where yaks are collected from numerous virtual-probe locations on and nearby the campus; (ii) data analysis, where the presence or not presence of the yaks at the various virtual probes is used to infer the approximate origins of the yaks. For the data collection, we provide a methodology that does not require any reverse

[*] Corresponding author. Email: keithwross@nyu.edu

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

IMC '16, November 14–16, Santa Monica, CA, USA.
© 2016 ACM. ISBN 978-1-4503-4526-2/16/11…$15.00
DOI: http://dx.doi.org/10.1145/2987443.2987449

RIGHTSLINK

engineering of the Yik Yak protocol, and can be run remotely from the target region. For the data analysis, we consider two methodologies: a supervised machine learning approach and an unsupervised heuristic approach. We then apply the methodologies to Yik Yak at two US campuses:

- On the University of Montana campus in Missoula, Montana, we first use a honeycomb layout with 2,880 virtual probes. We find that when Alice checks for nearby yaks, Yik Yak *does not* display messages in a circular region centered at Alice, as one might expect; instead, it displays messages in a square-like region shaped like the Yik Yak logo, approximately centered at Alice. Furthermore, the dimensions of the square-like shape may change from campus to campus.
- On the University of California Santa Cruz, we first generate a labeled data set in order to maliciously learn the locations of the messages. Specifically, we post 50 messages scattered throughout the University of California Santa Cruz (UCSC) campus, and then collect data from 160 virtual probes in a sparse layout. Using this labeled data, we use supervised machine learning to predict the locations of the messages, and then compare the predictions to the ground-truth location values. We also consider an unsupervised heuristic for predicting the locations. We show that we can accurately predict the locations of messages up to a small average error of 106 meters. We also devise an experiment where each message emanates from one of nine dorm colleges. In this experiment, we were able to determine the correct dorm college that generated each message 100% of the time. Our environment to run the experiments was located at NYU Shanghai, in Shanghai, China.

In this paper, we will also argue that natural obfuscation techniques, such as adding randomness to the message reporting algorithm, can also potentially be machine learned and hence susceptible to localization attacks. However, we will discuss one natural defense that is not susceptible to malicious machine learning.

2. ETHICS AND PRIVACY

In this paper, in order to illustrate this methodology, we have collected data by virtually placing the Yik Yak mobile application at two universities. The data we collect is publicly available data, and in fact is also collected by UCSC students using Yik Yak every day. Moreover, we only attempt to localize messages that we generate; no attempt was made to localize messages made by people outside of our research group. Furthermore, we make no attempt to de-anonymize any Yik Yak users. Finally, we informed the Yik Yak engineering team of this localization attack. We exchanged emails with the Yik Yak VP for engineering, who said he would work on resolving the problem with his team. We believe this study performs an important public service, as it shows that even anonymous social network services are susceptible to localization attacks. Our goal is to inform users and designers of such services, so that more comprehensive privacy solutions can be taken in the future.

3. METHODOLOGY

In this section, we describe a general methodology that allows the attacker to place and collect data from virtual probes, which can be remotely located anywhere in the world. The attacker does not have to be physically present in the region where he is trying to localize message origins. The basic steps in the methodology are listed as follows:

Data Collection Environment. HTTPS proxy as man-in-the-middle can be used to intercept traffic between the client and the server [24]. However, the man-in-the-middle attack fails to work on modern Yik Yak clients because recent releases of Yik Yak have implemented certificate pinning without accepting a spoofed certificate. Our data collection methodology, shown in Figure 1, does not require the attacker to reverse engineer the Yik Yak communication protocol or use a man-in-the-middle attack. Specially, we create a generic framework by combining multiple independent off-the-shelf tools, including a smartphone emulator, a task automation tool, and an optical character recognition (OCR) tool. The emulation environment runs Yik Yak on a personal computer (PC) and allows us to set the (*i.e.*, fake) GPS location to that of any target location. Each faked GPS location is referred to as a *virtual probe* location. We use the Sikuli test automation tool[1] to automatically set the fake locations, tap on various buttons in the Yik Yak app, and collect screenshots of displayed messages. We then process the screenshots, using OCR software (ABBYY FineReader,[2] to extract the messages.

Virtual Probe Layout. Our methodology requires that we use multiple virtual probes to collect data. There is a tradeoff between the number of probe locations and the effort required to perform the attack. Because we take the probe readings sequentially, and the reading and processing at each probe can take 60 seconds (often we need to scroll down and take multiple screenshots at a single probe), it could take roughly a week to collect the data from a single PC. To reduce the collecting effort with a honeycomb layout as shown in Figure 2, we design a virtual probe layout which is sparse but nevertheless provides enough information to perform meaningful inferences using machine learning. Particularly, as shown in Figure 3, we place four lines of probes, one vertical in the northern part of the extended region, one vertical line in the southern part, one horizontal line in the western part, and one horizontal line in the eastern part. For example, suppose each line of probes is 1,500 meters long and each pair of probes is separated by 50 meters, then approximately 120 probes are needed to collect data. This represents a significant reduction in the number of probes, thereby reducing the data collection time to 6 - 7 hours and also reducing the number of message/people-nearby requests sent to the server. However, a more fine-grained probe layout could potentially reduce the localization errors.

Collection of Labeled Data. Supervised machine learning requires labeled data. To create the labeled data, the methodology requires the generation of a small number of messages. We use fake GPS to place the messages at random locations, and then use the automation tool to take Yik Yak message readings across the entire probe layout, thereby generating the labeled data.

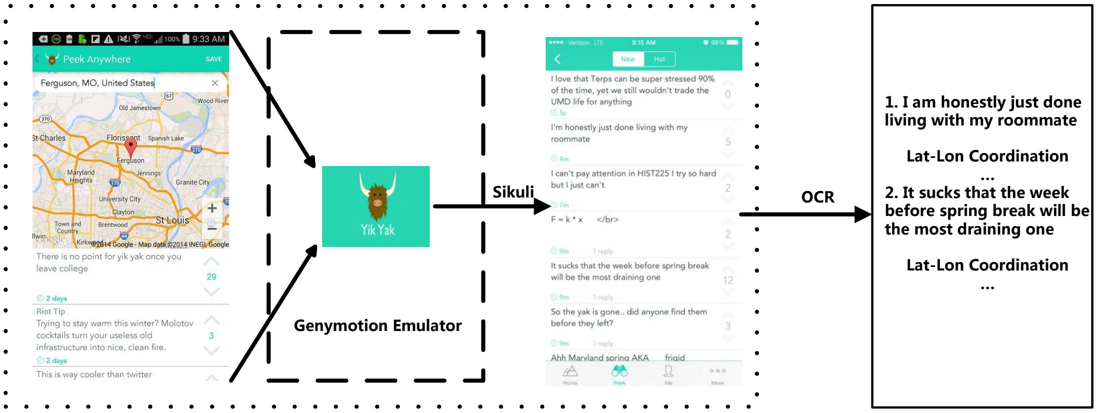
Supervised Machine Learning. We then use a portion of the labeled data to train a machine learning algorithm. Once having learned the weights, we predict the locations of the messages for the remaining portion of the labeled data. We also consider using a simple centroid heuristic for predicting the locations of the messages.

Calculating Geographic Distances on Earth. In order to determine the error rates for our inference methods, we will need to convert longitude and latitude values to meters by applying the Haversine formula. The Haversine formula gives great-circle distances between two points on a sphere from their longitudes and latitudes and is well-conditioned when measuring distances between points that are located very close together [21].

In the remainder of this section, we elaborate on collection of labeled data and machine learning application steps.

[1] http://www.sikuli.org/

[2] http://finereader.abbyy.com/



There is no point for yik yak once you leave college
29
2 days

Riot Tip
Trying to stay warm this winter? Molotov cocktails turn your useless old infrastructure into nice, clean fire.
3
2 days

This is way cooler than twitter

I love that Terps can be super stressed 90% of the time, yet we still wouldn't trade the UMD life for anything
0
5s

I'm honestly just done living with my roommate
5
4m

I can't pay attention in HIST225 I try so hard but I just can't
2
7m

F = k * x </br>
2
9m 1 reply

It sucks that the week before spring break will be the most draining one
12
9m 1 reply

So the yak is gone.. did anyone find them before they left?
3
9m 1 reply

Ahh Maryland spring AKA frigid

1. I am honestly just done living with my roommate

Lat-Lon Coordination

...

2. It sucks that the week before spring break will be the most draining one

Lat-Lon Coordination

...

Figure 1: Methodology: The Android emulator runs Yik Yak application; Sikuli interacts with the Yik Yak to set the fake GPS location and takes screenshots of Yik Yak bulletin-board; FineReader (OCR tool) then extracts all the messages shown on each screenshot.

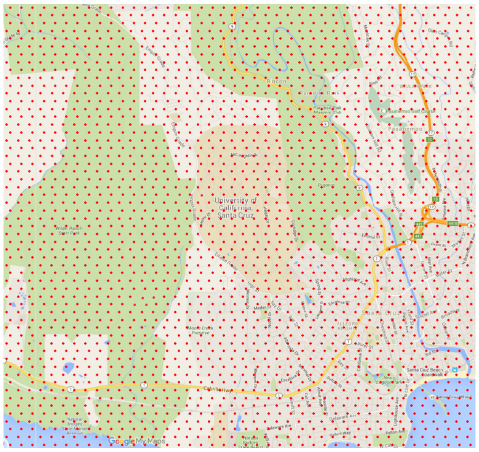


Figure 2: A honeycomb layout

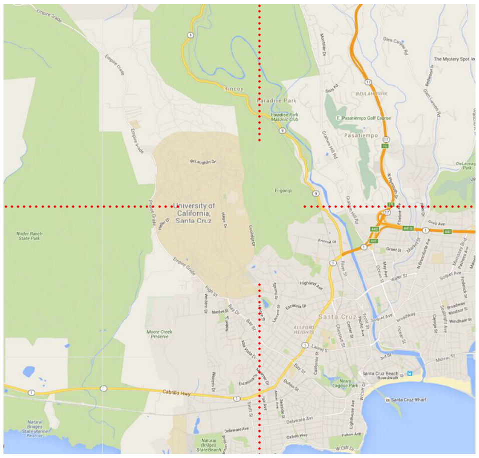


Figure 3: A sparse probe layout (four lines of probes)

3.1 Collection of Labeled Data

We will use supervised machine learning to infer the locations where anonymous messages were generated and the locations of users. Suppose there are n probes in our layout. A labeled data example takes the form $(\mathbf{x}, y)$, where $\mathbf{x} = (x_1, \ldots, x_n)$, y is the known longitude and latitude of a message/user, and x_i is the probe reading at the ith probe for the specific message. The feature x_i can take one of two values: 1: If the message is present in the ith probe reading; 0: If message is not present. In order to create this labeled data, that attacker can go through the following steps:

- Generate m messages and locate the m messages at random locations in the target region.
- Run the automation tool so that the Yik Yak service is run at each probe in the layout. A given screenshot may contain one or more of the generated messages.
- After applying OCR to each screenshot, determine which of the m messages are present at each of the n probe locations.

Thus, we only traverse the probe layout once to collect the presence information for all m messages. Let $y^{(i)}$ denote the latitude and longitude locations of the ith randomly placed message. Let $\mathbf{x}^{(i)}$ denote the corresponding probe readings in the probe layout. Then $\left(y^{(i)}, \mathbf{x}^{(i)}\right)$, $i = 1, \ldots, m$ are labeled data.

Recall that our probe layout has four lines, as shown in Figure 3. For the Yik Yak service, for any given message and any line of probes, all of the inner probes up to some threshold will observe the message, and all of the outer probes beyond the threshold will not observe the message. Therefore we can reduce the number of features from n to just four by defining x_E to be the number of probes that observe the message from east. If there are p probes in each line, then x_E can take values in $\{0, 1, \ldots, p\}$. Similarly we can define x_W, x_N, x_S from respective directions, each taking values in $\{0, 1, \ldots, p\}$. In this case, the labeled data collapses to $\left(y^{(i)}, x_E^{(i)}, x_W^{(i)}, x_N^{(i)}, x_S^{(i)}\right)$, for all $i = 1, \ldots, m$. We mention that if the service were to apply an obfuscation technique (such as randomly choosing messages to display for each click), then we would not collapse the features and instead use the extended feature vector.

3.2 Learning the Locations

After having collected the label data, we use supervised machine

RIGHTSLINK

learning regression to predict the locations of non-labeled messages. Many penalization techniques have been proposed, support vector regression [23], ridge regression [9], and the Lasso (Least Absolute Shrinkage and Selection Operator) [22]. For example, ridge regression minimizes the residual sum of squares subject to a bound on the L2-norm of the coefficients and the Lasso is imposing an L1-penalty on the regression coefficients. Owing to the nature of the L1-penalty, the Lasso does both continuous shrinkage and automatic variable selection simultaneously. Fu [8] compared the prediction performance of the Lasso, ridge, and bridge regression [7] and found that none of them uniformly dominates the other two. For simplicity, owing to its sparse representation, we finally select the Lasso regularized linear regression model as our predictive model. Each data point is then labeled with a latitude and a longitude $y = \left(y^{(i)}_{Lat}, y^{(i)}_{Lon}\right)$, for all $i = 1, \ldots, m$.

We also consider a heuristic that does not require any training data, so no artificial messages need to be introduced into the system. The heuristic is as follows. For any given target message for which we would like to determine the location, we take the probe readings and obtain the measurements x_E, x_W, x_N, x_S. Let x'_E and x'_W denote the corresponding longitudes for x_E and x_W. Similarly, let x'_N and x'_S denote the corresponding latitudes for x_N and x_S. Then we simply use $(x'_E + x'_W)/2$ to predict the longitude of the message, and $(x'_N + x'_S)/2$ to predict the latitude of the message. We refer to this location inference heuristic as *Centroid Prediction*. We will see that it provides very good results, although not quite as accurate as supervised machine learning.

4. REAL-WORLD EXPERIMENTS

In this section, we show how the methodology can predict the locations from where yaks originate with a high-degree of accuracy. We carry out three experiments. In the first experiment, we use a honeycomb layout with 2,880 probes covering the University of Montana campus in Missoula, Montana. The goal is to gain some basic insight into Yik Yak's service. In the second experiment, we use our experimental setup to post 50 messages scattered throughout the University of California Santa Cruz (UCSC) campus, and then collect data about these messages by using 160 virtual probes in a sparse layout. We use both the machine learning methodology and the centroid heuristic to predict the locations of the 50 messages, and then compare the predictions with the ground-truth location values. In the third experiment, again on the UCSC campus, we post the yaks from each of the dorm colleges on the UCSC campus, and then attempt to predict from which dorm college each yak was posted. We choose these campuses as they are large, contiguous, and mostly isolated from non-university activity. Our environment to run the experiments was located at NYU Shanghai, in Shanghai, China.

4.1 Preliminary Experiment

We conduct our first experiment on the University of Montana, in August 2015. In this experiment, we post five yaks at random locations on the University of Montana campus. We cover the city of Missoula with a honeycomb layout, using 2,880 probes, with adjacent probes separated by 200 meters. We partition the virtual probes across six computers, and have the six computers probing in parallel. At each location, our environment took screenshots of all the yaks seen. Each probe reading took anywhere from 30 seconds to 2 minutes to record all the yaks made available by Yik Yak.

Figure 4 shows the results for all the probes in the honeycomb for one of our five yaks. Green (resp. red) indicates presence (resp. no presence) of the yak at the probe. *Strikingly, for each of the messages, the shape of the set of green probes is square like* [15]. Oddly, there are also small ears attached to the square, vaguely resembling the Yik Yak logo! (There are some points within the shape that are red, which are due to the low recognition rates of the OCR software we were using at the time.). The other four yaks give rise to the same shape, but shifted so that the shape is roughly centered on the message.

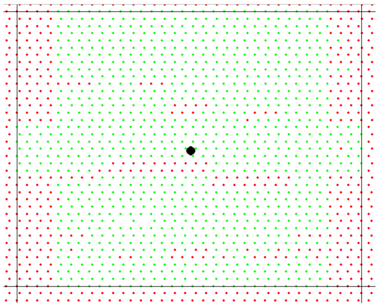


Figure 4: Probe results for one yak at the University of Montana. Green (resp. red) indicates presence (resp. no presence) of the yak at the probe.

From this preliminary experiment, we make the following observations. First, Yik Yak does not use a static geographic region for limiting all messages to the University of Montana (or to any other region); if it did, then the shape of the figure would not shift for different yaks. Instead, Yik Yak indeed employs user-centric proximity, showing the user messages that are in proximity to that specific user. Second, the proximity region has a definite shape, but surprisingly it is not a circle.

4.1.1 *Learning the Locations Using Lasso*

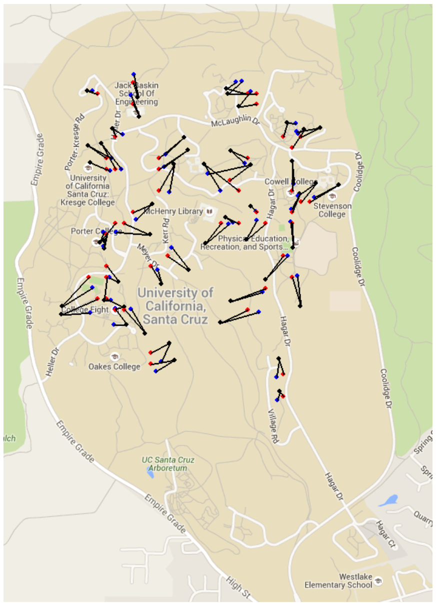


Figure 5: Ground-truth, machine learning prediction, and centroid prediction. A black dot represents the true location, the corresponding blue dot is our machine learning inference, and the corresponding red dot is the centroid prediction.

We now apply to Yik Yak the sparse-layout and machine-learning methodology, as described in Section 3. In this second experiment, we make predictions (again from Shanghai) for the University of California Santa Cruz (UCSC) in mid-December 2015. We use the sparse east, south, west, north linear layout, with probes spaced by 100 m, as described in Section 3. Each line has 20 probes. To reduce the impact of probing error, we use two parallel lines of probes for each direction, giving a total of 160 probes. If two parallel probes give inconsistent results, we use the inner-most probe that has seen the message for defining the corresponding feature value. (Due to an improved OCR tool, inconsistencies were very rare, and the additional lines of probes were probably unnecessary.)

To generate labeled data, we post 50 messages from random locations on the UCSC campus. As described in Section 3.2, we use Lasso regression over our trained data set to make predictions. We use the mean absolute errors (MAE) and the coefficient of determination R^2 [3] to measure the performance of the predictions. MAE measures how close predictions are to the ground-truth values. We apply leave-one-out cross validation to determine the parameters for Lasso and the weights for each feature. In order to make the weight regularization work properly, each feature is scaled within the range $[0, 1]$.

Figure 5 shows the locations of the 50 messages (black dots) and the corresponding machine-learning predicted locations (blue dots). We can see the predictions are always in the vicinity of the actual message locations. However, the machine-learning predictions have errors. As shown in Table 1, the MAE is 105.9 meters and the coefficient of determination R^2 is 0.966, which indicates that the given four features can explain 96.6% of the outcome variance. These estimates are remarkably accurate – accurate enough to predict the dorm college from which the yak was made, as we'll soon discuss.

The MAE error can potentially be reduced by collecting more labeled data and/or reducing the spacing between the probes. (With the current 100 m spacing, we would minimally expect an average of 50 m errors in both the east-west and north-south directions.) Table 1 also shows the root of mean squared error (RMSE); compared with MAE, RMSE penalizes more the large errors.

4.1.2 Centroid Heuristic

Recall from Section 3.2 that our centroid heuristic does not require training; instead it simply takes the averages of the longitude and latitude of the inner most probes having the presence of the target message. Figure 5 also shows the centroid predictions for our 50 yaks. A black dot represents the true location while the corresponding red dot represents our centroid prediction. As shown in Table 1, the MAE is 118.2 meters.

Table 1: Estimation accuracy (absolute distance errors in units of meters)

	Lasso Regression Inference	Centroid Prediction
MAE	105.9	118.2
RMSE	117.8	125.8
Minimum	14.4	30.3
Maximum	217.8	220.4
SD	51.2	43.6

4.2 Lasso Regression versus the Centroid Heuristic

Figure 5 and Table 1 show that the machine learning only slightly out-performs the centroid heuristic. Machine learning can potentially do better with a larger number of training examples. On the other hand, the centroid heuristic does not require any training. The errors for both approaches can potentially be reduced by reducing the spacing between the probes.

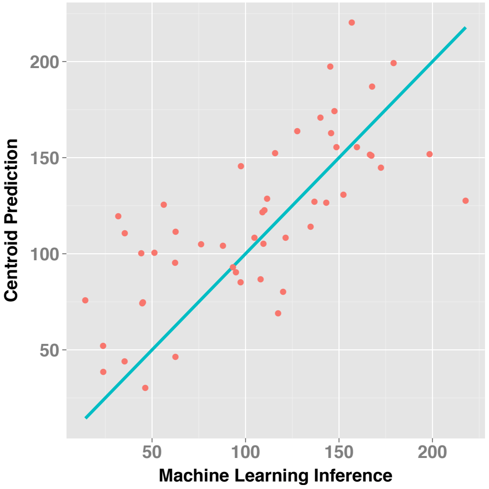


Figure 6: Scatter plot of fifty yaks scattered throughout the UCSC campus

Figure 6 shows a scatter diagram where, for each of the 50 messages, the machine learning errors are plotted along the x-axis and the centroid-heuristic predictions are plotted along the y-axis. We see from this figure that there is a strong correlation between the machine-learning error and the centroid error: when the machine-learning error is small, the centroid error is typically small; and when the machine-learning error is large, the centroid error is also typically large. We also see from Figure 6 and Figure 7 that for small errors (under 100 m with machine learning), the machine-learning predictions typically out-perform the centroid predictions; however, for large errors, neither seems to be significantly better than the other.

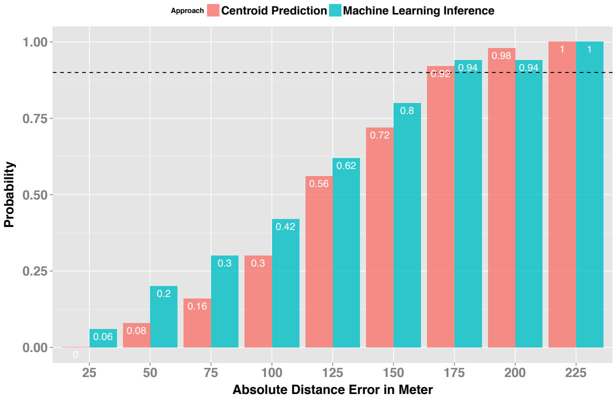


Figure 7: Cumulative distribution function of the absolute distance errors across approaches. Dashed line represents the 90th percentile of the absolute distance errors.

4.3 Dormitory Determination

UCSC dorms are organized into colleges, with each college located in a different region of the campus. In this third experiment, we assume that the majority of the yaks are emanating from campus dorms in these colleges. Our goal is to predict the specific college from which the yak originated. Specifically, we post nine yaks from the center of each of nine colleges on UCSC campus. For each such yak, we collect presence information from the 120 probes and predict the location of the yak using the centroid heuristic. We then predict the college that sent the yak as the college that is the closest to the predicted location.

Figure 8 shows the actual and predicted locations for each of our 9 yaks emanated from the center of 9 colleges of UCSC. A black dot represents the true location while the corresponding red dot represents our centroid prediction. As we can see from Figure 8, in each case the corresponding predicted college is equal to the actual college that made the post. Thus, in this experiment, the methodology is 100% correct in predicting the college that created the yak. As an example usage case, suppose a professor is teaching a class with, say, 20 students, and one of these students makes a derogatory yak about the professor. The professor can use the methodology in this paper to determine from which college the derogatory post came. If the professor can also obtain access to student housing information, he can then determine the students in his class that live in the identified college. If only one student from his class lives in the identified college, he can determine with a high-level of certainty which student made the derogatory remark on Yik Yak.

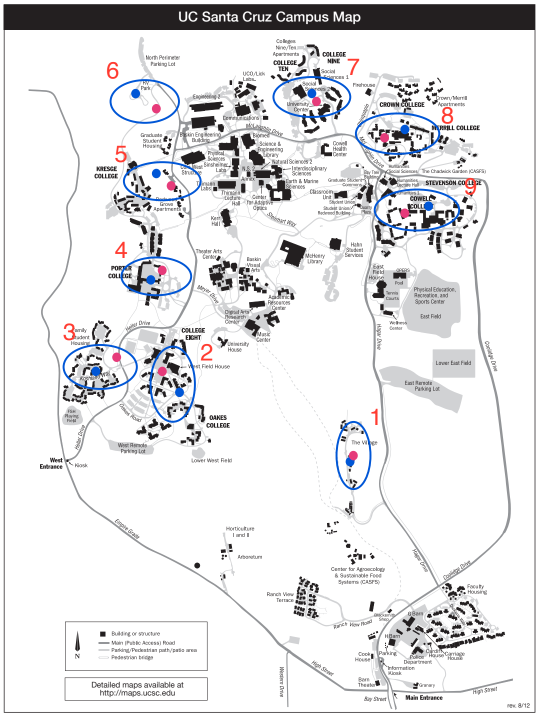


Figure 8: Campus building determination

5. PRIVACY ENHANCEMENT

Location-based services today, including Yik Yak, employ user-centric proximity, showing the user all the messages that are in proximity to that specific user. This property is at the heart of the localization attack – when shifting to a different physical location, the app displays a different set of people messages. To defend against this attack, Yik Yak can potentially try to employ some obfuscation. For example, when Alice makes a request, for each message in a region centered at Alice, Yik Yak draws a binary random variable X_i. If $X_i = 1$, the service displays that message; if $X_i = 0$ the service does not display the message. Such an approach would preserve the integrity of the Yik Yak service, showing messages that are nearby in all (albeit, a subset of them). We conjecture, however, that this and related obfuscation techniques can be potentially learned by machine learning if sufficient data is collected. This is an avenue for future research.

A much more robust defense would be for Yik Yak to use fixed and static display regions, where each region might cover a college campus, a small city, or a district in a large city. Static regions can also be defined outside of cities and campuses, with their sizes roughly inversely proportional to population densities. Having defined the regions, when Alice uses the service from within a region, she is shown all the messages in the region, but no other messages. With this redesign of the location-based service, the attack described in Section 3 would be thwarted – for any given message posted within a region (for example, a campus), all the probes located in the fixed region would report it, and all the probes outside the region would not report it. Thus, every message, no matter where it is posted from within the campus, would have the same feature vector, and the machine-learning attack would break down.

Additionally, the Yik Yak service can keep its location-based service as is, but has an eye out for the localization attack. For example, the service can impose tight restrictions on the number of periodic requests or the amount of traffic coming from each device. The service can block queries from an IP subnet if starts to see a large number of queries from that subnet. A workaround for this defense is to run the attack environment from a large number of different subnets, or to run the attack through proxies located on different subnets. Yik Yak can also detect forged GPS locations, either by enforcing strong location authentication using trusted software/hardware modules on mobile devices [13, 20], or by relying on wireless infrastructures, such as WiFi APs [11], cellular base stations [19], and femtocells [2].

Finally, users can also take precautions to prevent de-anonymization when using Yik Yak. Alice could refrain from making posts from her dorm room, but instead make posts from campus locations that she infrequently visits. Although such a precaution could go long way in keeping her anonymous, it would certainly take away from the enjoyment of the service.

6. RELATED WORK

For people-nearby and dating services, there is previous work on determining the locations of users for services that report either exact distances [14, 17, 18] or band distances [5, 10]. To the best of our knowledge, the closest work to ours is [25], which provides a methodology for localizing Whisper users within 0.2 miles based on the triangulation attack. We emphasize here that Whisper reports the distances between users and message origins, facilitating triangulation attacks. Localizing Yik Yak users is more challenging than localizing Whisper users, because Yik Yak neither reports exact distances nor distance bands about the yaks. Instead, it simply indicates which users are nearby (or which messages have been generated from nearby locations) without showing any specific distance; furthermore, the algorithm Yik Yak employs for deciding which messages to display is completely unknown. The current paper is an extension of a poster paper [15].

7. CONCLUSION

We showed that the popular anonymous social media application Yik Yak is susceptible to localization attacks, thereby putting user anonymity at risk. We provided a comprehensive data collection and supervised learning methodology that does not require any reverse engineering of the Yik Yak protocol, is fully automated, and can be remotely run from anywhere. We applied the measurement and machine learning methodologies to Yik Yak at two US campuses. We accurately predicted the locations of messages up to a small average error of 106 meters. We also devised an experiment where each message emanates from one of nine dorm colleges on the University of California Santa Cruz campus. In this experiment, we were able to determine the correct dorm college that generated each message 100% of the time. Finally, we described how Yik Yak can be modified to enhance the anonymity of its service.

Although our experiments and defenses have strictly focused on Yik Yak, our results are applicable to a plethora of mobile applications that rely on geolocation for "messages-generated-nearby" services or people-nearby services, such as Badoo, Grindr, WeChat. We find that all these apps are facing the potential localization attacks. Even if app developers are able to attempt to obfuscate by adding some randomness, geo-location can still be maliciously machine learned by the methodology proposed in this paper. To the best of our knowledge, this is the first paper that considers the problem of de-anonymizing users in the popular Yik Yak application. We hope our study will inform developers working on the next generation anonymous mobile applications.

Acknowledgments

We thank the anonymous reviewers for their valuable feedback. This work was supported in part by the National Science Foundation under grant CNS-1318659.

8. REFERENCES

[1] T. Abdollah. Yik Yak Isn't So Anonymous, Turns Data Over To Police, The Huffington Post, November 12, 2015.

[2] J. Brassil, P. K. Manadhata, and R. Netravali. Traffic signature-based mobile device location authentication. *Mobile Computing, IEEE Transactions on*, 13(9):2156–2169, 2014.

[3] A. C. Cameron and F. A. Windmeijer. R-squared measures for count data regression models with applications to health-care utilization. *Journal of Business & Economic Statistics*, 14(2):209–220, 1996.

[4] W. D. Cohan. Putting the Heat on Yik Yak After a Killing on Campus, The New York Times, January 6, 2016.

[5] Y. Ding, S. T. Peddinti, and K. W. Ross. Stalking beijing from timbuktu: A generic measurement approach for exploiting location-based social discovery. In *Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, pages 75–80. ACM, 2014.

[6] J. Dougherty. OSU Student Arrested For Making Threats On Social Media. News9 Oklahoma's Own, April 22, 2015.

[7] L. E. Frank and J. H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993.

[8] W. J. Fu. Penalized regressions: the bridge versus the lasso. *Journal of computational and graphical statistics*, 7(3):397–416, 1998.

[9] A. Hoerl and R. Kennard. Ridge regression, in Encyclopedia of Statistical Sciences, vol. 8, 1988.

[10] M. Li, H. Zhu, Z. Gao, S. Chen, L. Yu, S. Hu, and K. Ren. All your location are belong to us: Breaking mobile social networks for automated user location tracking. In *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*, pages 43–52. ACM, 2014.

[11] W. Luo and U. Hengartner. Proving your location without giving up your privacy. In *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, pages 7–12. ACM, 2010.

[12] J. Mahler. Who Spewed That Abuse? Anonymous Yik Yak App Isn't Telling. The New York Times, March 8, 2015.

[13] C. Marforio, N. Karapanos, C. Soriente, K. Kostiainen, and S. Capkun. Smartphones as practical and secure location verification tokens for payments. In *NDSS*, 2014.

[14] S. Mascetti, L. Bertolaja, and C. Bettini. A practical location privacy attack in proximity services. In *Mobile Data Management (MDM), 2013 IEEE 14th International Conference on*, volume 1, pages 87–96. IEEE, 2013.

[15] C. L. Nemelka, C. L. Ballard, K. Liu, M. Xue, and K. W. Ross. You can yak but you can't hide. In *Proceedings of the 2015 ACM on Conference on Online Social Networks*, pages 99–99. ACM, 2015.

[16] H. J. Parkinson. Yik Yak: the anonymous app taking US college campuses by storm, The Guardian, October 21, 2014.

[17] I. Polakis, G. Argyros, T. Petsios, S. Sivakorn, and A. D. Keromytis. Where's wally? precise user discovery attacks in location proximity services. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 817–828. ACM, 2015.

[18] G. Qin, C. Patsakis, and M. Bouroche. Playing hide and seek with mobile dating applications. In *ICT Systems Security and Privacy Protection*, pages 185–196. Springer, 2014.

[19] S. Saroiu and A. Wolman. Enabling new mobile applications with location proofs. In *Proceedings of the 10th workshop on Mobile Computing Systems and Applications*, page 3. ACM, 2009.

[20] S. Saroiu and A. Wolman. I am a sensor, and I approve this message. In *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*, pages 37–42. ACM, 2010.

[21] B. Shumaker and R. Sinnott. Virtues of the haversine. *Sky and telescope*, 68:158–159, 1984.

[22] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[23] V. Vapnik. *The nature of statistical learning theory*. Springer Science & Business Media, 2013.

[24] G. Wang, B. Wang, T. Wang, A. Nika, B. Liu, H. Zheng, and B. Y. Zhao. Defending against sybil devices in crowdsourced mapping services gang wang. In *Proceedings of The 14th ACM International Conference on Mobile Systems, Applications, and Services*. ACM, 2016.

[25] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao. Whispers in the dark: analysis of an anonymous social network. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 137–150. ACM, 2014.

RIGHTSLINK