Hindawi
Scientific Programming
Volume 2020, Article ID 8868686, 11 pages
https://doi.org/10.1155/2020/8868686

Research Article

QuPiD Attack: Machine Learning-Based Privacy Quantification Mechanism for PIR Protocols in Health-Related Web Search

Rafiullah Khan,[1,2] Arshad Ahmad,[3] Alhuseen Omar Alsayed,[4] Muhammad Binsawad,[5] Muhammad Arshad Islam,[6] and Mohib Ullah[1,2]

[1]*Institute of Computer Science and Information Technology, The University of Agriculture, Peshawar, Pakistan*
[2]*Capital University of Science and Technology, Islamabad, Pakistan*
[3]*Department of Computer Science, University of Swabi, Anbar, Pakistan*
[4]*Deanship of Scientific Research, King Abdulaziz University Jeddah, Jeddah, Saudi Arabia*
[5]*Faculty of Computer Information Systems, King Abdulaziz University Jeddah, Jeddah, Saudi Arabia*
[6]*National University of Computer and Emerging Sciences, Islamabad, Pakistan*

Correspondence should be addressed to Arshad Ahmad; yaarshad@gmail.com

Received 16 March 2020; Accepted 22 April 2020; Published 14 July 2020

Academic Editor: Rodziah Binti Atan

With the advancement in ICT, web search engines have become a preferred source to find health-related information published over the Internet. Google alone receives more than one billion health-related queries on a daily basis. However, in order to provide the results most relevant to the user, WSEs maintain the users' profiles. These profiles may contain private and sensitive information such as the user's health condition, disease status, and others. Health-related queries contain privacy-sensitive information that may infringe user's privacy, as the identity of a user is exposed and may be misused by the WSE and third parties. This raises serious concerns since the identity of a user is exposed and may be misused by third parties. One well-known solution to preserve privacy involves issuing the queries via peer-to-peer private information retrieval protocol, such as useless user profile (UUP), thereby hiding the user's identity from the WSE. This paper investigates the level of protection offered by UUP. For this purpose, we present QuPiD (query profile distance) attack: a machine learning-based attack that evaluates the effectiveness of UUP in privacy protection. QuPiD attack determines the distance between the user's profile (web search history) and upcoming query using our proposed novel feature vector. The experiments were conducted using ten classification algorithms belonging to the tree-based, rule-based, lazy learner, metaheuristic, and Bayesian families for the sake of comparison. Furthermore, two subsets of an America Online dataset (noisy and clean datasets) were used for experimentation. The results show that the proposed QuPiD attack associates more than 70% queries to the correct user with a precision of over 72% for the clean dataset, while for the noisy dataset, the proposed QuPiD attack associates more than 40% queries to the correct user with 70% precision.

1. Introduction

Currently, web search engines (WSEs) have become the preferred way to find health care-related content on the World Wide Web. A recent survey reports that more than 80% of patients use WSE to seek health-related information before consulting the physician [1], while according to the report published by Pew Research Center, 35% of American adults consulted WSE to diagnose medical conditions [2]. However, while using the web search services, the user usually posts their physical condition and health information as a query [3]. Web search engines claim that they collect and maintain user queries as user profile for various activities such as result ranking [4], market research [3], personalization [5], targeted advertisements [6], and others. On the brighter side, maintaining users profile can actually improve the quality of results and user experience, while on the darker side, this indiscriminate collection of users' queries may cause critical privacy breaches as users' queries may contain sensitive and personal information [7]. This issue of users' privacy breach received significant attention in 2005 when the US Department of Justice compelled Google to submit records of users' queries [8]. Later, America Online (AOL) released (pseudonymized) 20

million queries of more than 650,000 users submitted in three months of time [9], from which the identities of some users had been inferred through personal information enclosed in their queries [10].

Patient's health information is considered to be a sensitive issue since ancient times, and it is also reflected in the Hippocratic Oath [11] that physician will keep the patient's information secrete [12]. However, in online and public health facility services, user privacy is just becoming behavior tracking [12]. Consider a scenario when a user posts a series of private queries related to his/her health condition such as "HIV" or "diabetes." WSE may sell this information to the advertisement agencies or other companies for business purposes, which ultimately breaches the user's privacy [3]. Such kind of privacy disclosure happened in 2006 when the New York Times managed to deduce and infer personal information from the search history from the pseudonymized log published by AOL. One of them was a 62-year-old widow who conducted hundreds of searches related to her health condition such as "hand tremors," dry mouth," and "nicotine effects on the body" which were linked back to her [13].

To address this issue of privacy infringement, several methods have been proposed. These methods include user profile obfuscation [14], query scrambling [15], anonymizing networks [16], and private information retrieval (PIR) protocols [17–20]. In a user profile obfuscation, a user profile is contaminated with fake queries to mislead the WSE. In the query scrambling technique, the user query is replaced by a set of blurred and benign synonyms and later posted to WSE. Techniques based on anonymizing network forward the user query through a series of routers to make it difficult for WSE to trace the origin of the query. These methods hide the IP address while the user is still traceable through cookies and device fingerprints [21]. In PIR protocols, a group of users submits queries on behalf of each other to hide their identity.

Despite the fact that the aforementioned methods improve the user privacy, yet some previous studies [22–25] using a machine learning algorithm and user profile (i.e., user history or logged user queries) show that an adversary is able to break profile obfuscation and anonymizing network methods. However, it is not clear if an adversary is able to break PIR protocols using machine learning techniques. Therefore, in this research, we propose a machine learning-based attack in order to evaluate the effectiveness of popular PIR protocol, i.e., useless user profile (UUP) [17, 18].

A higher-level goal of this work is to analyze the effectiveness of PIR protocol in preserving users' privacy against an adverse WSE (from here on, we will call the PIR protocols as UUP, for simplicity of presentation without loss of generality). In UUP, a group of users exchanges their queries with each other in such a way that the identity of the query originator node remains hidden from other group mates. In the next step, all group members submit the received queries to the WSE and results are broadcasted in the group. On the WSE side, the user's query is received in plain text but with a different identity, and thus WSE cannot identify the originator of the queries. We set out to investigate whether it is possible (and to what extent) for an adverse WSE—equipped with users' web search profile (histories)—to link the queries coming out of UUP exit user to the original users and thus undermine the privacy provided by UUP.

To better understand the limits of UUP on user's privacy, we present in this paper a study of UUP focusing on active users. This study is conducted with QuPiD attack, a machine learning-based attack that determines the distance between the user's profile and query. We conducted our experiments with randomly selected active 100 users from publicly available AOL dataset and treated them as users of UUP. The AOL dataset is composed of over 20 million queries submitted during the period of March 1, 2006, to May 31, 2006, by 6.5 million users. The data of the first two months are used as training data while the last month data are used as testing data. We measured the efficiency of attack using some known machine learning matrices: precision, recall, F-measure, and true-positive rate. The results showed that our proposed QuPiD attack associates more than 70% queries to the correct user with more than 72% precision. Based on the results, we can conclude that most of the users are vulnerable to privacy infringement despite using UUP. The contributions of this work are as follows:

(1) Proposed QuPiD attack: a machine learning-based attack for privacy evaluation of PIR protocols

(2) A proposed new vector for query classification

(3) Recommendation of a suitable machine learning algorithm for query classification

The remainder of the paper is organized as follows: In Section 2, we describe the proposed QuPiD attack. Experimentation setup, preprocessing of the dataset, feature vector construction, and classification algorithms are discussed in Section 3. Section 4 presents the experimental results. Section 5 presents the conclusions and outlines directions for future work.

2. Adverse Model and QuPiD Attack

Users are more concerned about the privacy risks of querying WSEs. In this work, we investigated the robustness of popular PIR protocol, i.e., UUP. As mentioned earlier, WSE receives a user's query with a different identity due to the shuffling process. Therefore, the entries of queries will never appear with their true originator in the weblog. However, the weakness of this protocol is the timing of query submission by all group members. After the query shuffling step, every group member submits the received query to WSE almost at the same time. Due to which their entries appeared close to each other in the weblog. Figure 1 illustrates an example of query entries in the weblog. In Figure 1, exhibit 1 shows the users' queries before the shuffling process while exhibit 2 shows the queries after the shuffling process. After shuffling, the queries are submitted to WSE (Figure 1, exhibit 3).

In the proposed adverse model, WSE is assumed to be an entity whose goal is to work against the privacy-preserving solution and identify the user of interest (UoI) queries for profiling purposes. It is assumed that WSE is equipped with

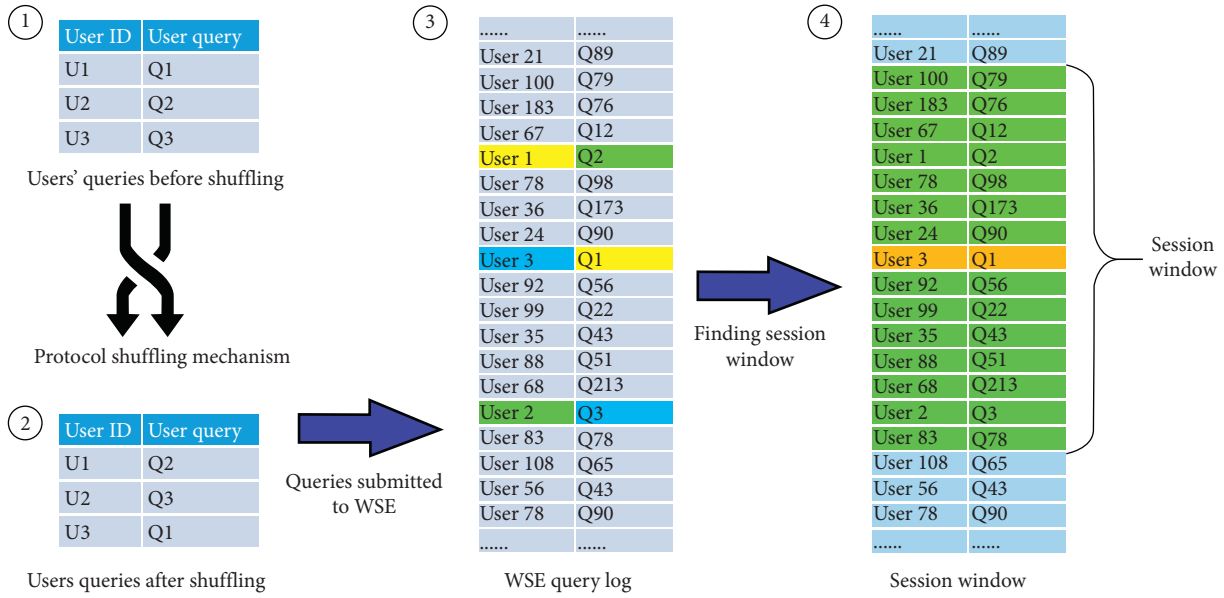

Figure 1: Query entry in the weblog and session window.

the user's search history (i.e., user profile) PU. The user profile contains queries submitted by the user in the past without using any UUP protocol shown in equation (1) (where $P_q i$ shows the queries in the UoI profile).

$$\text{PU} = \left[P_q 1,\ P_q 2,\ P_q 3,\ \ldots\ P_q n\right]. \tag{1}$$

The user profile PU is used as training data for building the classification model. As the dataset used for experimentation is spread across three months' duration, the first two months' data are used as a training set, while the UUP protocol is simulated with the third month data to create an anonymized log (as shown in Figure 1, exhibit 3). The anonymized log is used as the test set. For testing, all session windows of the UoI are drawn out from the query logs. Here, the session window is a block of records (query entries in the log) in an anonymized log that contains the entry of UoI, but with another user [26, 27]. In other words, the session window is composed of the selected number of queries' entries in the WSE query log, which appeared immediately before and after the query of UoI. As shown in Figure 1 (exhibit 4), our UoI is "User 3" and the session window size is 15 records (7 records before UoI and 7 after UoI). For this research, we have used the window size of 251 records. Each session window (S_{win}) is composed of 125 queries appearing before and 125 queries appearing after the query of UoI (as per the recommendation of [27]). A generic session window S_{win} is shown in equation (2) (where q_i represents a query in the session window). The collection of all session windows GS_{win} is shown in equation (3).

$$S_{\text{win}} = [q1, q2, q3, \ldots, q125, q\text{UoI}, q126, \ldots, q251], \tag{2}$$

$$\text{GS}_{\text{win}} = \left[S_{\text{win}}1, S_{\text{win}}2, S_{\text{win}}3, \ldots, S_{\text{win}}n\right]. \tag{3}$$

As shown in the query log, the target user who uses any PIR protocol will remain hidden since his/her query is exchanged with a query of another user in the group. Therefore, a session window is used to reduce the testing data. Both PU (training set) and GS_{win} (testing set) are used as input to the algorithm of the adverse model. The working of the adverse model is presented in Algorithm 1 and depicted in Figure 2. The working of the algorithm is as follows:

$$\text{PU}_v = \left[P_q 1v, P_q 2v, P_q 3v, \ldots, P_q nv\right], \tag{4}$$

$$S_{\text{win}v} = [q1v, q2v, q3v, \ldots, q125v, q\text{UoI}v, q126v, \ldots, q251v]. \tag{5}$$

For experimentation purposes, two subsets of 100 users were created from the AOL dataset constituting a three-month web query log of AOL users. Each subset was divided into two portions, i.e., training and testing data. Training data are composed of the first two months of the log, while the testing data are composed of the last month of the log. The details of the user selection criteria and dataset formation are discussed in Section 4.

3. Methodology

3.1. AOL Dataset. We used the real-world web search query log released by AOL in 2006 for the evaluation of our proposed adverse model. The AOL dataset consists of over 20 million queries submitted during the period of March 1, 2006, to May 31, 2006, by 6.5 million users. Although the AOL dataset is old and has a lot of deficiencies as compared to the current situation, we are forced to use this dataset due to a lack of availability of the benchmark dataset. The attributes of the query log are user ID, query, date and time of the query, the rank of the content clicked, and the clicked URL. For experimentation purposes, the data of the first two months were used as user profile (PU) or training data while the third month's data were the new queries to be classified

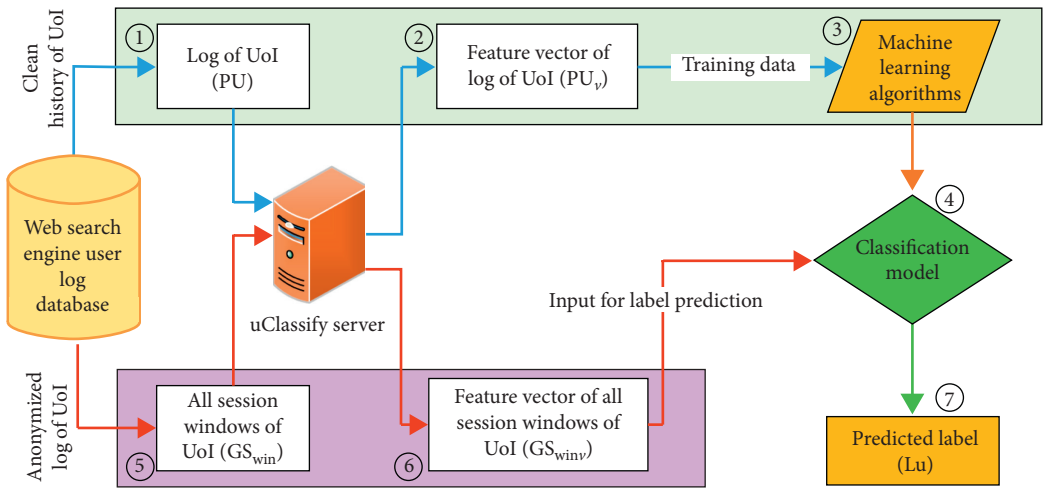

Figure 2: Operation of the adverse model.

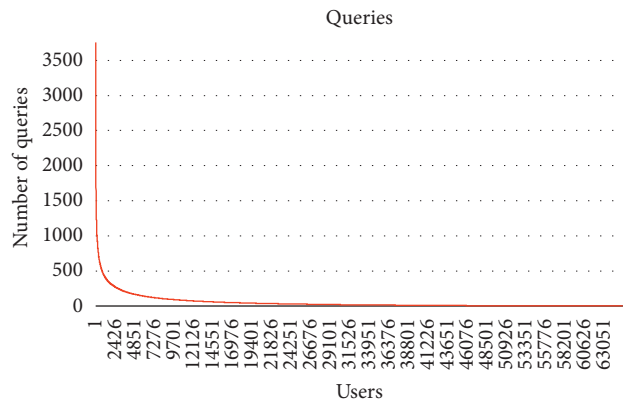

Figure 3: Distribution of the number of queries issued per user in the selected dataset.

(i.e., testing data). The distribution of the number of queries issued per user in the selected dataset is shown in Figure 3. For experimentation, we chose 100 users with high query frequency instead of concentrating on all users. The user selection criteria are discussed in Section 3.3, while the summary of the dataset is provided in Table 1.

3.2. Feature Vector Extraction. The dataset is composed of five attributes: user ID, query, date and time of the query, the rank of the content clicked, and the clicked URL. Since our adverse model works with the user ID, submitted query, and query score in ten major topics, we neglect the remaining features. To obtain query scores in ten major classes, we used uClassify service that provides classifiers for topics, age, gender, sentiments, language detection, and many others. In this paper, the topic classifier is employed that provides the numeric value of 10 categories against each query. The topic classifier uses a subset of topics from the Open Directory Project (ODP) directory in which topics are placed in a hierarchy. The classes are Arts, Business, Computers, Games, Health, Home, Recreation, Science, Society, and Sports. The classifier provides the percentage of each query in each category. For example, for query "olive oil," the score for each topic is shown in Table 2.

In some cases, uClassify was unable to find the score of the dominant topic of the submitted query. For example, uClassify is unable to find the dominant class for the query "glenliviet 18." Therefore, in that case, uClassify just divided an equal score in each class, i.e., 10% for each class. We refer to this kind of query as a "confused query" (shown in Table 2). In the dataset of selected 100 users, uClassify marked 28% of the queries as confusing queries. Therefore, we conducted our experiments using two datasets. One dataset was comprised of both confused and unconfused queries, while the other dataset was comprised of only unconfused queries to find the impact of confusing queries over the results of a classifier. From this point onwards, the dataset with confused queries will be referred to as a noisy dataset while the dataset with only unconfused queries will be referred to as the clean dataset. The details of both datasets are given in Table 3.

3.3. User Selection and Subset Construction. Instead of conducting experiments using all users, we focused on a few users who were considered to be active. Active users are those users who submitted more than 300 queries for at least 61 days during the entire period. From the analysis of the dataset, we found only 21,407 (3.29%) users to be active users. From those active users, we randomly selected 100 users as UoI. The cumulative distribution of queries in both noisy and clean dataset is shown in Figure 4. To see the effects of the size of the training data, we divide both noisy and clean datasets into five groups based on the average of query frequency. The selected 100 users are divided into 5 groups in both datasets. The average number of total, training, and testing instances in all groups for both datasets is given in Table 4.

3.4. Anonymized Log Creation. As mentioned earlier, the AOL data spans across three months. For experimentation purposes, we have considered the first two months' data as

Input: User Profile (PU); all session windows belong to the user (GS_{win}).
Output: Expected User Label (Lu)

```
(1)  procedure QUERY ASSOCIATION (PU, GS_win)
(2)    for P_q i ∈ PU do
(3)      PU_v ⟵ get feature vector for (P_q i)
(4)    P_Model ⟵ Classification Algorithm (PU_v)
(5)    for S_win i ∈ GS_win do
(6)      for q_k ∈ S_win j do
(7)        q_k v ⟵ get feature Vector for (q_k)
(8)        Lu ⟵ P_Model (q_k v)
(9)    return Lu
```

(1) Firstly, the user profile (PU) feature vector is acquired for training purposes. The user profile with the feature vector (PU_v) is shown in equation (4). The feature vector is acquired from the uClassify (http://www.uclassify.com) service, a machine learning web service that provides numerous different classifiers for text classification. We have selected the "Topics" classifier that gives the score of each phrase or query in 10 major classes including Sports, Society, Science, Recreation, Home, Health, Games, Computers, Business, and Arts.
(2) In the second step, a classification model P_{Model} is built using PU_v and supervised machine learning algorithms. To test the response of the data with different classification techniques, 10 classification algorithms are selected from tree-based, rule-based, lazy learner, metaheuristic, and Bayesian families.
(3) After the classification model (P_{Model}), the third step is to acquire the feature vector S_{winv} shown in equation (5) for the queries of session window S_{win} from uClassify for testing data.
(4) In the last step, each query of S_{winv} is provided to the classification model for the expected label Lu. The label Lu shows whether the incoming query belongs to UoI or not.

Algorithm 1: Associating incoming query to the user using the prior profile.

Table 1: AOL dataset properties.

Total queries	36,389,567
Total users	657,426
Unique queries	10,154,742
Attributes	5 (AnonID, query, query time, item rank, click URL)
Time duration	01 March, 2006–31 May, 2006

Table 2: The score of queries from uClassify.

Query	Arts	Business	Computers	Games	Health	Home	Recreation	Science	Society	Sports
Olive oil	0.0386	0.0974	0.0280	0.0396	0.0569	0.4659	0.0652	0.1028	0.0874	0.0182
Glenliviet 18	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

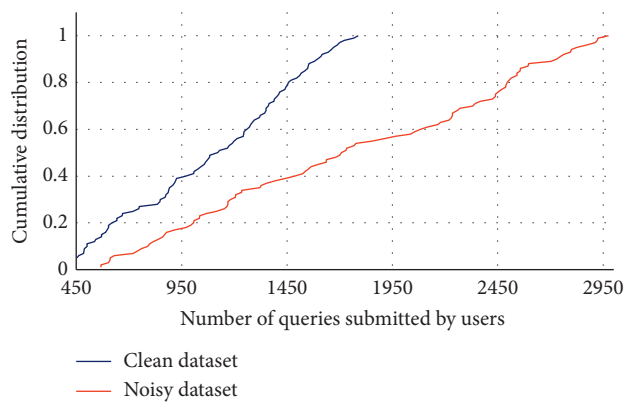

Figure 4: Distribution of the number of queries submitted by users in the clean and noisy datasets.

the clean history of UoI available to the search engine and last month's data as new queries to be classified. The selected PIR protocol, i.e., (UUP) is simulated with the third month's query log to create the anonymized log of UoI. The parameters considered for simulations are group size and the number of queries submitted by the respective users. According to the literature, UUP is tested with a group size of 3, 4, 5, and 10 users [17, 18]. Another study indicated that a bigger group size offers more privacy [27]. We, therefore, considered a group size of 20 users. The number of queries submitted by the target user is dependent on the actual query frequency of the selected user in the third month queries log.

3.5. Classification Algorithms. In several previous studies, Peddinti et al. [23, 24] and Petit [21] used Random Forest,

AD Tree, Zero R, Regression, and SVM algorithms for the classification of the data queries. In both studies, the classification model was biclass, i.e., the query is machine or user generated. Moreover, the model was built based on two attributes like query and assigned label. In our work, however, the classification model is multiclass, i.e., in the testing data, the model will decide which query belongs to which user and the model is based on twelve attributes (discussed in Section 3.2). We selected ten off-the-shelf (with default settings) different families' classification algorithms. We chose J48 [28] and Logistic Model Tree (LMT) [29] from the tree-based family, Decision Table [30], JRip [31], and OneR [32] from rule-based family, IBK [33] and KStar [34] from lazy learner family, Bagging [35] and LogitBoost [36] from metaheuristic family, and Bayes Net [37] from Bayesian family. Rep Tree [38] and Regression are used as base classifiers for Bagging and LogitBoost algorithms.

3.6. Performance Evaluation Metrics. Three metrics, precision, recall, and F-measure, are usually used to evaluate the performance of a classifier. Precision represents how many of the identified samples are correct and recall describes how many of the total samples are correctly identified. Both precision and recall are mathematically represented in the following equations:

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}, \tag{6}$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}, \tag{7}$$

where true positive represents the actual positives that are correctly identified cases by the classifier and false positive is the proportion of all negatives that still yield positive test outcomes, while false negative represents the proportion of positive which yields negative test outcomes with the test. The trade-off between precision and recall is represented by a unified metric called F-measure. The value of F-measure is in the range from 0 to 1, where 0 shows none of the samples is classified correctly, while 1 shows perfect classification. Mathematically, F-measure is represented as

$$\text{F} - \text{measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \tag{8}$$

4. Results and Discussion

The primary aim of this study is to propose and evaluate a privacy quantification model for PIR protocols. Experiments are performed with two datasets: noisy and clean (Section 3.3), each set composed of 100 users having variable query frequencies distributed over five groups. For each UoI, we measured precision, recall, and true-positive percentage of correctly classified queries from an anonymized log.

Tables 5 and 6 illustrate the true-positive percentage of the queries of *UoI* in both datasets. According to Table 5, all algorithms correctly identified more than 89% queries of 2 users in the noisy dataset except OneR and LogitBoost. OneR correctly identified 80% to 90% queries of 4 users. Overall, IBK correctly identified more than 50% queries of 36 users followed by Bagging and KStar with 30 and 28 users, respectively, in the noisy dataset. Similarly, in the clean dataset, LMT and IBK were able to correctly identify more than 89% queries of 14 users followed by J48 and Bagging with 12 users each. Overall, IBK correctly identified more than 50% queries of all 100 users followed by KStar and Bagging with 96 and 92 users in the clean dataset. The detailed performance of all algorithms (in terms of true-positive rate) of the clean dataset is given in Table 6. In both datasets, the performance of lazy learner family algorithms (i.e., IBK and KStar) is better when compared to other selected algorithms.

As mentioned earlier, both datasets are further divided into 5 groups of 20 users (Table 4) in order to observe the impact of the size of training on the accuracy of results. Table 7 shows the comparison of the performance of all algorithms with a variation of the training dataset size in the noisy dataset. The performance of each algorithm is measured in precision and recall. IBK and KStar associated more than 40% queries to the correct user with the precision of above 60% in all cases, while Bagging, J48, Decision Table, and Bayes Net associated more than 25% queries to the correct user with the precision of above 60% in all cases. From the perspective of the size of the training dataset, it is slightly difficult to draw a conclusion about its effect on accuracy. Almost every algorithm shows irregular behavior with a variation in the training dataset size. For the first three groups, the performance of IBK, J48, KStar, and LMT is observed more accurately. However, unexpectedly, the rate of recall drops for the last two groups. The results of precision and recall of noisy data are plotted in Figure 5.

In the clean dataset, however, a clear pattern of improvement in the recall is visible. According to Table 8, the performance of all algorithms is improving as the size of the training dataset increases. IBK and KStar associated more than 62% queries to the correct users with the precision of above 70% in all cases, while Bagging, J48, Decision Table, and LMT associated more than 51.68% to 82.84% queries to the correct user with the precision of above 60% in all cases. Among other algorithms, Bayes Net was able to associate more than 70% of the queries in some cases. Although the increase in recall with the increase in training data is not linear, an improvement pattern is clearly visible in the clean dataset. The results of precision and recall of clean data are plotted in Figure 6.

Overall, IBK and Bagging associated 45.1% and 43% queries to the correct user with above 70% precision for the noisy dataset, while J48, KStar, and LMT associated 42.2%, 41.7%, and 40.6% queries to the correct user with the precision of 70.9%, 73.5%, and 70.2%. Similarly, in the clean dataset, IBK and Bagging associated 79.5% and 75.7% queries to the correct user with 79.6% and 75.9% precision, while J48, KStar, and LMT associated 73.9%, 74.4%, and 72% queries to the correct user with the precision of 73.9%,

Table 3: Properties of noisy and clean datasets.

Properties	Noisy dataset	Clean dataset
Training instances	116101	71817
Testing instances	59809	36998
Total instances	175911	108815
Max queries by the single user	2975	1788
Min queries by the single user	567	365
Distinct queries	69164	49662

Table 4: Average dataset instances (queries).

Dataset	Group	Total data	Training data	Testing data
Noisy	Group 1	777.55	513.183	264.37
	Group 2	1215.15	801.99	413.15
	Group 3	1752.45	1156.62	595.833
	Group 4	2332.1	1539.18	792.91
	Group 5	2718.3	1794.08	924.22
Clean	Group 1	509.55	336.30	173.25
	Group 2	820.95	541.83	279.12
	Group 3	1132	747.12	384.88
	Group 4	1367	902.22	464.78
	Group 5	1611.25	1063.43	547.83

Table 5: Percentage of users in a group based on true-positive values of the noisy dataset.

True-positive percentage bands	Tree-based		Rule-based			Lazy learner		Metaheuristic		Bayesian
	J48	LMT	DT	JRip	OneR	IBK	KStar	Bagging	LogitBoost	Bayes Net
100%-90%	2	2	2	2	0	2	2	2	0	2
90%-80%	2	2	2	2	4	2	0	2	2	2
80%-70%	4	2	4	2	0	4	4	4	2	4
70%-60%	4	8	4	2	6	4	6	4	0	2
60%-50%	14	2	4	6	0	24	16	18	2	14
50%-40%	26	28	24	4	10	18	22	20	6	20
Below 40%	48	56	60	82	80	46	50	50	88	56

Table 6: Percentage of users in a group based on true-positive values of the clean dataset.

True-positive percentage bands	Tree-based		Rule-based			Lazy learner		Metaheuristic		Bayesian
	J48	LMT	DT	JRip	OneR	IBK	KStar	Bagging	LogitBoost	Bayes Net
100%-90%	12	14	10	10	4	14	8	12	0	4
90%-80%	18	12	14	8	4	32	26	24	4	12
80%-70%	26	22	22	6	8	24	26	22	2	18
70%-60%	20	26	16	8	6	22	20	18	0	18
60%-50%	12	16	10	10	18	8	16	16	6	14
50%-40%	12	10	20	16	18	0	4	8	10	18
Below 40%	0	0	8	42	42	0	0	0	78	16

76.1%, and 72.6%. The top three algorithms in terms of F-measure (trade-off between precision and recall) for the noisy dataset are IBK, Bagging, and J48 with the score of 0.514, 0.487, and 0.477, respectively, while for the clean dataset, the top three algorithms are IBK, Bagging, and KStar with the score of 0.793, 0.753, and 0.745, respectively. Hence, IBK is determined to be a more appropriate algorithm for the feature vector "categories." The results of the average F-measure of the noisy and the clean dataset are plotted in Figure 7.

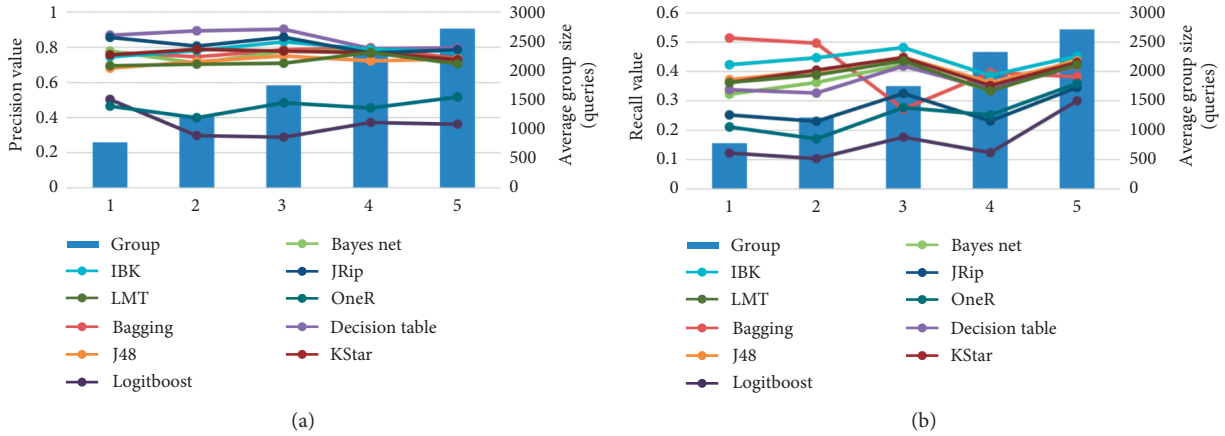

Figure 5: Noisy dataset's groupwise precision and recall in different groups. (a) Noisy dataset precision. (b) Noisy dataset recall.

Table 7: Precision and recall of noisy dataset in different groups.

Group			Group 1	Group 2	Group 3	Group 4	Group 5
Tree-based	J48	Precision	0.68	0.71	0.75	0.72	0.72
		Recall	0.37	0.40	0.44	0.36	0.43
	LMT	Precision	0.69	0.70	0.70	0.75	0.72
		Recall	0.36	0.38	0.43	0.33	0.42
Rule-based	Decision Table	Precision	0.86	0.89	0.90	0.79	0.79
		Recall	0.33	0.32	0.41	0.34	0.41
	JRip	Precision	0.85	0.80	0.85	0.77	0.78
		Recall	0.25	0.23	0.32	0.23	0.34
	OneR	Precision	0.46	0.39	0.48	0.46	0.51
		Recall	0.21	0.17	0.27	0.25	0.35
Lazy learner	IBK	Precision	0.74	0.78	0.83	0.78	0.77
		Recall	0.42	0.44	0.48	0.38	0.45
	KStar	Precision	0.75	0.78	0.77	0.76	0.72
		Recall	0.36	0.40	0.44	0.35	0.72
Metaheuristic	Bagging	Precision	0.77	0.74	0.78	0.79	0.73
		Recall	0.37	0.41	0.45	0.36	0.44
	LogitBoost	Precision	0.50	0.29	0.28	0.37	0.36
		Recall	0.12	0.10	0.17	0.12	0.30
Bayesian	Bayes Net	Precision	0.77	0.71	0.77	0.78	0.69
		Recall	0.32	0.36	0.42	0.33	0.44

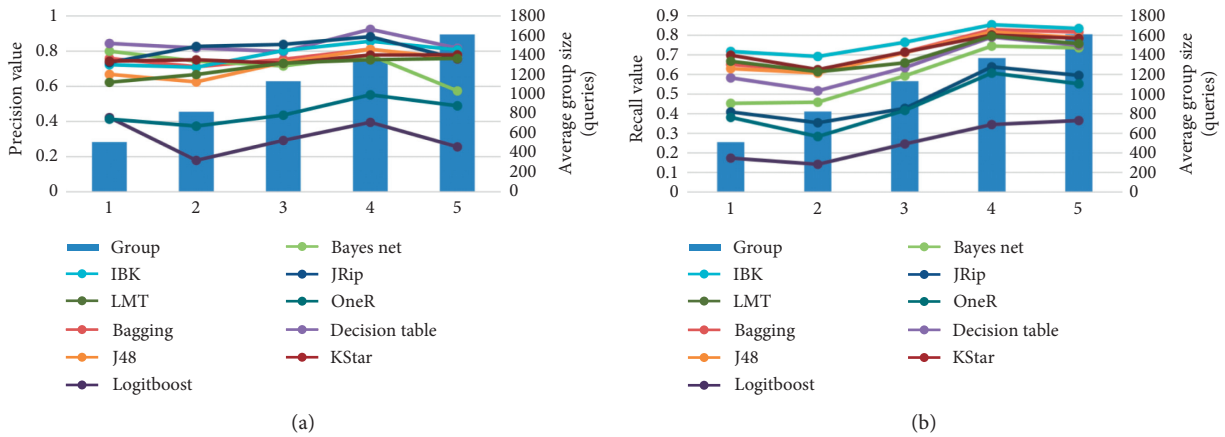

Figure 6: Clean dataset's groupwise precision and recall in different groups. (a) Clean dataset precision. (b) Clean dataset recall.

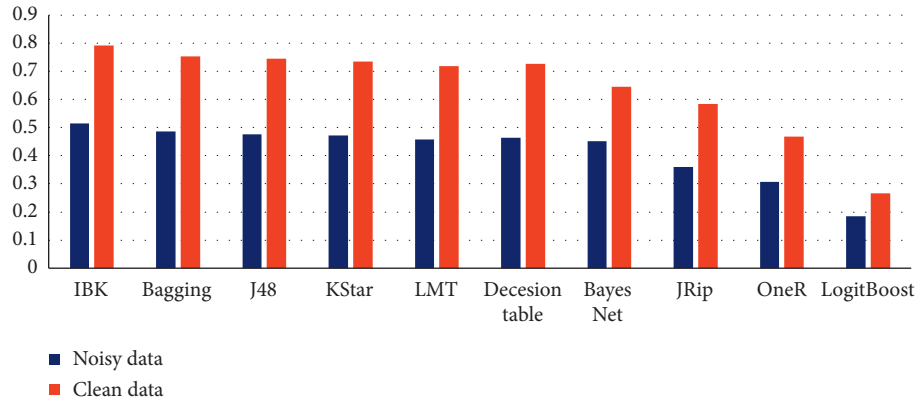

Figure 7: Average F-measure of all selected classification algorithms for noisy and clean datasets.

Table 8: Precision and recall of clean dataset in different groups.

Group			Group 1	Group 2	Group 3	Group 4	Group 5
Tree-based	J48	Precision	0.66	0.62	0.73	0.80	0.76
		Recall	0.62	0.60	0.71	0.81	0.78
	LMT	Precision	0.62	0.66	0.73	0.75	0.75
		Recall	0.66	0.61	0.65	0.79	0.75
Rule-based	Decision Table	Precision	0.84	0.81	0.79	0.92	0.81
		Recall	0.58	0.51	0.63	0.79	0.74
	JRip	Precision	0.73	0.82	0.83	0.88	0.75
		Recall	0.40	0.35	0.42	0.63	0.59
	OneR	Precision	0.41	0.37	0.43	0.55	0.48
		Recall	0.38	0.28	0.41	0.60	0.55
Lazy learner	IBK	Precision	0.72	0.70	0.80	0.85	0.80
		Recall	0.71	0.69	0.76	0.85	0.83
	KStar	Precision	0.74	0.75	0.73	0.77	0.77
		Recall	0.69	0.62	0.71	0.80	0.78
Metaheuristic	Bagging	Precision	0.75	0.71	0.75	0.81	0.75
		Recall	0.65	0.61	0.71	0.82	0.81
	LogitBoost	Precision	0.42	0.17	0.29	0.39	0.20
		Recall	0.19	0.14	0.23	0.34	0.38
Bayesian	Bayes Net	Precision	0.79	0.74	0.71	0.77	0.57
		Recall	0.45	0.45	0.59	0.74	0.73

5. Conclusions

Health information has been regarded as sensitive private information since ancient times. However, WSE collects this information for selling and targeted advertisements, which can infringe user's privacy. This paper presents QuPiD attack: a machine learning-based attack that quantifies the level of protection provided by popular PIR protocol UUP. The QuPiD attack uses a classification algorithm and the history of the user to classify an incoming query. We used two subsets (noisy and clean datasets) of real-world web data to test the proposed model. We showed that our proposed attack succeeds in correctly associating incoming queries to their real originator at a high ratio. For the selection of the best classification algorithm, we conducted our experiments with ten classification algorithms from different families. J48 and LMT from the tree-based family, Decision Table, JRip, and OneR from rule-based family, IBK and KStar from lazy learner family, Bagging and LogitBoost from metaheuristic family, and Bayes Net from Bayesian family were selected. The results showed that IBK is the most appropriate algorithm if the "categories" feature vector is used.

During the analysis of the noisy dataset, almost every algorithm showed irregular behavior with the variation in the training dataset size. However, analyzing the clean dataset, we found that when increasing the size of the training data while building the classification model, the testing data in terms of recall are improving. We, therefore, conclude that noise is one of the factors responsible for unsteady behavior. Our analysis shows that PIR protocols are vulnerable to machine learning attacks, even with the first-degree classification tags of queries. This situation is alarming for currently available PIR protocols. Any web search engine or even web service armed with a profile of the user can expose a targeted user. In the future, we are interested to assess the proposed attack from different

perspectives, such as the impact of group size, the number of queries in a session, user profile size, and others. Moreover, we are excited to explore the unsteady behavior of classification algorithms.

Data Availability

The data used to support the findings of the study are available at http://www.radiounderground.net/aol-data/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Deanship of Scientific Research, King Abdulaziz University (KAU), Jeddah, Saudi Arabia.

References

[1] M. W. Ng, R. Smith, N. Wickramesinghe, P. J. Smart, and N. Lawrentschuk, "Health on the net: do website searches return reliable health information on hemorrhoids and their treatment?" *International Surgery*, vol. 102, no. 5-6, pp. 216–221, 2017.

[2] S. Fox and M. Duggan, "Health online 2013," *Health*, pp. 1–55, Pew Research Center, Washington, DC, USA, 2013.

[3] R. Khan, M. A. Islam, M. Ullah, M. Aleem, and M. A. Iqbal, "Privacy exposure measure: a privacy-preserving technique for health-related web search," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 6, pp. 1196–1204, 2019.

[4] P. Thomas, B. Billerbeck, N. Craswell, and R. W. White, "Investigating searchers' mental models to inform search explanations," *ACM Transactions on Information Systems (TOIS)*, vol. 38, no. 1, pp. 1–25, 2020.

[5] H. Yoganarasimhan, "Search personalization using machine learning," *Management Science*, vol. 66, no. 3, pp. 1045–1070, 2020.

[6] F. Long, K. Jerath, and M. Sarvary, "Leveraging information from sponsored advertising at online retail marketplaces," *Kenan Institute of Private Enterprise Research Paper*, no. 20-03, https://ssrn.com/abstract=3516104, 2019.

[7] S. B. Mokhtar, A. Boutet, P. Felber, M. Pasin, R. Pires, and V. Schiavoni, "X-search: revisiting private web search using intel sgx," in *Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference*, pp. 198–208, Las Vegas, NV, USA, December 2017.

[8] K. Hafner and M. Richtel, *Google Resists US Subpoena of Search Data*, New York Times, New York, NY, USA, 2006.

[9] G. Pass, A. Chowdhury, and C. Torgeson, "A picture of search. Infoscale' 06, Hong Kong," in *Proceedings of the 1st International Conference on Scalable Information Systems*, ACM, New York, NY, USA, 2006.

[10] I. Lundberg, A. Narayanan, K. Levy, and M. J. Salganik, "Privacy, ethics, and data access: a case study of the fragile families challenge," 2018, https://arxiv.org/abs/1809.00103.

[11] L. Edelstein, "The hippocratic oath: text, translation and interpretation," in *Ancient Medicine: Selected Papers of Ludwig Edelstein*, pp. 3–63, Johns Hopkins Press, Baltimore, MD, USA, 1943.

[12] T. Libert, "Privacy implications of health information seeking on the web," *Communications of the ACM*, vol. 58, no. 3, pp. 68–77, 2015.

[13] M. Barbaro, T. Zeller, and S. Hansell, *A Face is Exposed for AOL Searcher No. 4417749*, New York Times, New York, NY, USA, 2006.

[14] V. Toubiana, L. Subramanian, and H. Nissenbaum, "Trackmenot: enhancing the privacy of web search," 2011, https://arxiv.org/abs/1109.4677.

[15] A. Arampatzis, G. Drosatos, and P. S. Efraimidis, "Versatile query scrambling for private web search," *Information Retrieval Journal*, vol. 18, no. 4, pp. 331–358, 2015.

[16] R. Dingledine, N. Mathewson, and P. Syverson, *Tor: the Second-Generation Onion Router*, Naval Research Lab, Washington, DC, USA, 2004.

[17] C. Romero-Tris, J. Castellà-Roca, and A. Viejo, "Distributed system for private web search with untrusted partners," *Computer Networks*, vol. 67, pp. 26–42, 2014.

[18] C. Romero-Tris, A. Viejo, and J. Castellà-Roca, "Multi-party methods for privacy-preserving web search: survey and contributions," in *Advanced Research in Data Privacy*, pp. 367–387, Springer, Berlin, Germany, 2015.

[19] K. Stokes and M. Bras-Amorós, "Optimal configurations for peer-to-peer user-private information retrieval," *Computers & Mathematics with Applications*, vol. 59, no. 4, pp. 1568–1577, 2010.

[20] M. Ullah, M. A. Islam, R. Khan, M. Aleem, and M. A. Iqbal, "ObSecure logging (OSLo): a framework to protect and evaluate the web search privacy in health care domain," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 6, pp. 1181–1190, 2019.

[21] A. Petit, *Introducing Privacy in Current Web Search Engines*, Université de Lyon, Lyon, France, 2017.

[22] A. Gervais, R. Shokri, A. Singla, S. Capkun, and V. Lenders, "Quantifying web-search privacy," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pp. 966–977, Scottsdale, AZ, USA, November 2014.

[23] S. T. Peddinti and N. Saxena, "On the privacy of web search based on query obfuscation: a case study of TrackMeNot," in *Proceedings of the International Symposium on Privacy Enhancing Technologies Symposium*, pp. 19–37, Berlin, Germany, July 2010.

[24] S. T. Peddinti and N. Saxena, "On the effectiveness of anonymizing networks for web search privacy," in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, pp. 483–489, Hong Kong, China, March 2011.

[25] A. Petit, T. Cerqueus, A. Boutet et al., "SimAttack: private web search under fire," *Journal of Internet Services and Applications*, vol. 7, no. 2, 2016.

[26] R. Khan and M. A. Islam, "Quantification of PIR protocols privacy," in *Proceedings of the 2017 International Conference on Communication, Computing and Digital Systems (C-CODE)*, pp. 90–95, Islamabad, Pakistan, March 2017.

[27] R. Khan, M. Ullah, and M. A. Islam, "Revealing pir protocols protected users," in *Proceedings of the 2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, pp. 535–541, Dublin, Ireland, August 2016.

[28] J. R. Quinlan, *C4. 5: Programs for Machine Learning*, Elsevier, Amsterdam, Netherlands, 2014.

[29] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," *Machine Learning*, vol. 59, no. 1-2, pp. 161–205, 2005.

[30] R. Kohavi, "The power of decision tables," in *Proceedings of the European Conference on Machine Learning*, pp. 174–189, Heraclion, Greece, April 1995.

[31] W. W. Cohen, "Fast effective rule induction," in *Machine Learning Proceedings 1995*, pp. 115–123, Elsevier, Amsterdam, Netherlands, 1995.

[32] R. C. Holte, "Very simple classification rules perform well on most commonly used datasets," *Machine Learning*, vol. 11, no. 1, pp. 63–90, 1993.

[33] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37–66, 1991.

[34] J. G. Cleary and L. E. Trigg, "K∗: an instance-based learner using an entropic distance measure," in *Machine Learning Proceedings 1995*, pp. 108–114, Elsevier, Amsterdam, Netherlands, 1995.
[35] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
[36] E. Frank, M. Hall, G. Holmes et al., "Weka-a machine learning workbench for data mining," in *Data Mining and Knowledge Discovery Handbook*, pp. 1269–1277, Springer, Berlin, Germany, 2009.
[37] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Machine Learning*, vol. 29, no. 2/3, pp. 131–163, 1997.
[38] N. Midha and V. Singh, "Classification of E-commerce products using RepTree and K-means hybrid approach," in *Big Data Analytics*, pp. 265–273, Springer, Berlin, Germany, 2018.