Privacy Risk Predictions Based on Fundamental Understanding of Personal Data and an Evolving Threat Landscape

Haoran Niu K. Suzanne Barber
University of Texas at Austin
{haoranniu, sbarber}@utexas.edu

***Abstract*—It is difficult for individuals and organizations to protect personal information without a fundamental understanding of relative privacy risks. By analyzing over 5,000 empirical identity theft and fraud cases, this research identifies which types of personal data are exposed, how frequently exposures occur, and what the consequences of those exposures are. We construct an Identity Ecosystem graph—a foundational, graph-based model in which nodes represent personally identifiable information (PII) attributes and edges represent empirical disclosure relationships between them (e.g., the probability that one PII attribute is exposed due to the exposure of another). Leveraging this graph structure, we develop a privacy risk prediction framework that uses graph theory and graph neural networks to estimate the likelihood of further disclosures when certain PII attributes are compromised. The results show that our approach effectively answers the core question: Can the disclosure of a given identity attribute possibly lead to the disclosure of another attribute?**

***Index Terms*—privacy protection, risk prediction, link prediction algorithm, graph neural networks, graph convolutional networks, deep learning, identity graph**

I. Introduction

Different individuals and organizations have different sets of personally identifiable information (PII), and therefore have different perspectives on which PII attributes are more vulnerable, more valuable, and in greater need of protection. An individual's PII includes personal data in four different categories—What you Know (e.g., name, address, phone number, mother's maiden name), What you Have (e.g., driver's license, Social Security Card, employee ID, passport), What you Are (e.g., fingerprint, voice, facial image), and What you Do (e.g., patterns of life such as websites visited, GPS locations visited, phone logs) [1].

Protecting PII data can be costly and time-consuming. Research has uncovered various strategies to reduce risks of unintended data disclosure [2], including statistical disclosure limitation (SDL) techniques commonly used by national statistical agencies before releasing public-use data sets. Meanwhile, research about data self-destruction focuses on protecting data privacy for users who choose cloud services [3] [4]. With numerous methodologies for protecting privacy, this paper focuses on the first step of privacy protection: determining which set of data to protect. Protecting the most valuable and risky (i.e., likely to be exposed) set of PII promises to be a more effective and efficient method of protecting a person's personal, sensitive data. This is because individuals and institutions usually have limited time, energy, and financial resources allocated for privacy protection.

This research is based on the premise that if individuals and organizations have a more fundamental understanding and a more accurate evaluation of privacy risks resulting from disclosing PII, they will be in a better position to protect that information. Additionally, better risk analysis of personal data sharing will inform a wide range of information security and privacy applications.

This research determines which data requires the most protection by evaluating the consequences of exposing the respective data. Specifically, it analyzes the privacy risks incurred when an individual or an organization shares or loses a respective PII attribute. We primarily seek to answer the question: **Could the disclosure of a given PII attribute, such as date of birth, lead to the disclosure of another attribute, such as ATM PIN**? PII attributes have respective risk values associated with them [1] [5]. By evaluating the risk scores of the possible disclosed PII attributes, we can provide a quantified prediction of privacy risks based on empirical data of disclosures.

To provide these predictions of quantified privacy risks, we leverage graph theory. Many well-studied networks, such as transportation networks and social networks, are analyzed using graphs [6] [7]. Similar to those networks, there exist connections and relationships among PII attributes. This research represents each PII attribute as a node in the graph. The directional graph edges represent the disclosure/exposure relationship between two PII attributes—specifically, the probability that the disclosure of one PII attribute (e.g., date of birth) could lead to the exposure of another PII attribute (e.g., address). We name this graph of PII nodes with disclosure/exposure directional edges the Identity Ecosystem graph.

After constructing an Identity Ecosystem graph, we created and trained three different link prediction models. We also developed a risk score calculation model. Together, the Identity Ecosystem graphs, the link prediction algorithms, and the risk score calculation model form a comprehensive risk prediction framework.

Suppose an individual has lost PII attributes A1 and B2 and wants to determine which other PII attributes become highly risky as a result. The pipeline of the risk prediction framework is outlined below:

- The individual provides a risk score threshold ($[0, 100]$ scale) to indicate the level of risk of concern. The default risk score threshold is 0, meaning all potentially disclosed attributes are considered.
- The individual provides the PII attributes A1 and B2 that were stolen or lost.
- The PII attribute set is defined as all nodes in the Identity Ecosystem graph except for A1 and B2.
- The risk prediction model—composed of a link prediction module and a risk score calculation module—takes A1, B2, the risk score threshold, and the PII attribute set as input, and outputs the PII attributes that may be disclosed and have risk scores above the threshold.

Figure 1 uses an example of risk score threshold 75 to explain the pipeline. In general, the contributions of this paper include:

- We propose a way to construct Identity Ecosystem graphs; we also provide a way to customize the Identity Ecosystem graphs with different sizes and personal needs. (Section III).
- We create and train three different link prediction models (Section IV-B, IV-C, and IV-D).
- We conduct thorough testing experiments with different sizes of identity Ecosystem graphs to show the performance of the three link prediction models. (Section V).
- We construct a risk score calculation model (Section VI).

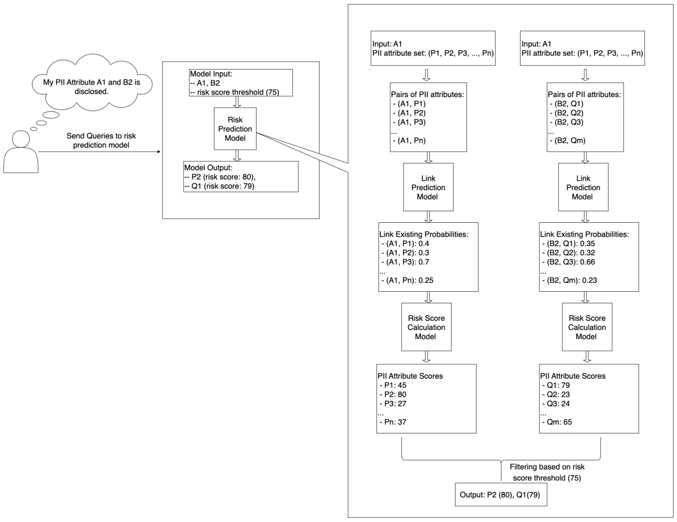


Fig. 1. Pipeline of the Risk Prediction Framework: from a Query to PII Attribute Risk Prediction Results.

II. Related Work

Privacy protection and risk assessment is an important research topic for many market sectors that collect, store and analyze personal identity information. Researchers have developed a lot of data mining techniques to find odd patterns in data and identity fraudulent transactions [8]. The applications of machine learning and data mining algorithms can help market sectors such as financial services, healthcare, transportation and more to prevent the loss of money and time.

For instance, paper [9] provides a method to assess privacy risks for medical big data. The researchers also apply Fuzzy C-means clustering algorithm to cluster the users into different groups, assign different permissions, and improve the users' access control accuracy.

Despite plenty of research addressing privacy risk assessment, there is limited research on finding the connections among different aspects of personal data. This paper not only fills the gap of finding the inter-connections between personal data but also uses graphs to model and predict the interactions of personal data to uncover risks in sharing this data.

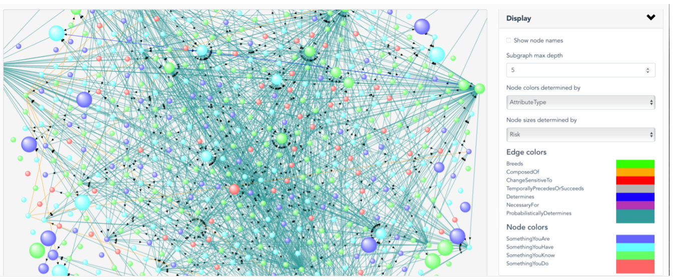
In relation to the graph-based models utilized in this project, the methods of link prediction play an important role. Link prediction is commonly used for detecting missing links or adding future connections. There are some simple methodologies based on node similarity such as Jaccard's coefficient [10] and Adamic/Adar measure [11]. These simple techniques are computational efficient but cannot perform well on a lot of graphs. Therefore, more and more researchers seek the help of machine learning algorithms [12] [13] [14].

The framework of this paper combines both simple similarity properties/scores and supervised learning algorithms. It can answer the questions about identity information risk prediction efficiently.

III. Experimental Setup – Constructing the Identity Ecosystem Graph Model

Now let's explore the Identity Ecosystem first – its content, its privacy risks analytics and its new prediction capabilities offered in this paper to help individuals to protect their personal information from unintended and harmful disclosure. The Center for Identity at The University of Texas at Austin (UT CID) first introduced an Identity Ecosystem graph [1]. The UTCID Identity Ecosystem graph uses nodes to represent PII attributes (i.e. personal data) and connects the nodes based on various types of relationships between PII attributes (Figure 2). It should be noted that while the paper [1] presents various types of relationships, the "probability of disclosure" relationship is the one leveraged in the privacy risk prediction research presented in this paper. Figure 1 displays the UTCID Identity Ecosystem graph with the nodes colored by Type (What you Know, What you Have, What you Are, What you Do) and nodes sized by value, where the value of a PII attribute node is calculated based on the degree to which that PII attribute was monetized in over 5000 (and counting) identity theft and fraud cases analyzed in the Identity Threat Assessment and Prediction project (ITAP) [1] [5]. For simplification purpose, we call the set of identity theft and fraud cases collected by the ITAP project "UTCID ITAP dataset".

In this paper, we reconstruct the UTCID Identity Ecosystem graph and only keep a minimal set of graph features in order to predict the PII disclosure risks efficiently. In the real-world news stories regarding identity theft and fraud cases, it is



Display

□ Show node names

Subgraph max depth: 5

Node colors determined by: AttributeType

Node sizes determined by: Risk

Fig. 2. UTCID Identity Ecosystem Graph Representing PII Attributes and Their Relationships

common that we might only know what the thieves used to steal the PII attribute and what the PII attributes that are lost. If we can use a minimal set of information to predict the PII disclosure risks accurately, then we do not need to waste money and time to collect other information. The construction of the new Identity Ecosystem graphs follows the principles below.

We still use nodes to represent PII attribute. Directed edges between PII attributes A and B, arrow $A \rightarrow B$ means an event disclosing node A leads to a disclosure of node B. Each edge is assigned with a weight. Overall, an arrow $A \rightarrow B$ with a weight w_i means an event disclosing node A may lead to a disclosure of node B and such a disclosure happened w_i times according to the empirical ITAP data of collected identity theft and fraud cases [12]. Figure 2 shows an example of visualization based on a simplified use case. The thickness of an edge visually shows a weight, frequency of the specific exposure (e.g. $A \rightarrow B$ with w_i). Figure 3 shows an example ecosystem graph with three identity attributes. The example graph below tells us:

- Disclosure of "name" attribute may lead to the disclosure of "bank account" with probability of 0.3 (calculation: 3/(3+7) = 0.3).
- Disclosure of "name" attribute may lead to the disclosure of "birth date" with probability of 0.7 (calculation: 7/(3+7) = 0.7).

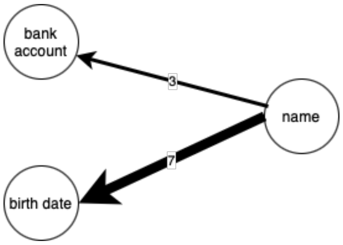


Fig. 3. An Example Identity Ecosystem Graph with Three Nodes.

Throughout the entire paper, we will use the Identity Ecosystem graphs which are constructed based on the rules above to analyze and predict PII attribute disclosure and consequently privacy risks.

To collect and structure the data necessary to execute the proposed risk prediction, two steps are required. First, we need to "preprocess" the UTCID ITAP dataset. For the inputs and outputs of the identity theft and fraud cases in the ITAP dataset, only identity-related information is retained. For each case, inputs are the data used by the actors to conduct the identity theft and fraud. Outputs include the consequent data that are acquired, stolen or otherwise exposed by the conclusion of the identity theft and fraud actions.

The second step is to construct the Identity Ecosystem graphs. We previously introduced the graph construction rules in the previous paragraphs with a small example graph (e.g. Figure 2, a graph with three nodes and two edges). Here, we will further explain the rules completely by using multiple input and output identity attributes.

Suppose there is a reported identity crime case. From the case, we can extract three input identity attributes – "bank account", "name" and "Social Security Number". The corresponding output identity attributes are "credit card" and "debit card". If we construct an Identity Ecosystem graph based on this single case, five nodes and six directed edges can be added to this graph. The five nodes are "bank account", "name", "Social Security Number", "credit card" and "debit card" respectively. The six edges are listed below:

- From "bank account" to "credit card".
- From "bank account" to "debit card".
- From "name" to "credit card".
- From "name" to "debit card".
- From "Social Security Number" to "credit card".
- From "Social Security Number" to "debit card".

Since this graph is created from the single identity criminal case, every edge in the graph has a weight 1 since each input-output attribute pair appears only once (given this scope of only one case).

Now let us consider a more complicated situation. Imagine we have three collected identity criminal cases (Case 1, 2 and 3). In the Case 1, the input identity attributes are "bank account", "name" and "Social Security Number" and the corresponding output attributes are "credit card" and "debit card". In the Case 2, the inputs are "bank account" and "Social Security Number" and the outputs are "birth date", "credit history" and "credit card". For the Case 3, the input is "Social Security Number" and the output is "bank account".

For the three cases described above, there are seven unique identity attributes: "bank account", "name", "Social Security Number", "credit card", "debit card", "birth date" and "credit history". Based on the three cases, the relationships among the seven attributes are shown below:

- From "bank account" to "credit card" (weight: 2).
- From "bank account" to "debit card" (weight: 1).
- From "bank account" to "birth date" (weight: 1).
- From "bank account" to "credit history" (weight: 1).
- From "name" to "credit card" (weight: 1).
- From "name" to "debit card" (weight: 1).
- From "Social Security Number" to "credit card" (weight: 2).
- From "Social Security Number" to "debit card" (weight: 1).

- From “Social Security Number” to “birth date” (weight: 1).
- From “Social Security Number” to “credit history” (weight: 1).
- From “Social Security Number” to “bank account” (weight: 1).

The weights reflect the occurrence frequencies for the input-output pairs among the identity theft and fraud cases. Figure 4 shows the graph visualization of this three-case example above.

The largest Identity Ecosystem graph we construct in this paper relies on 5,636 identity theft and fraud cases from the UTCID ITAP dataset. Following the procedures given above, the graph constructed from these 5,636 criminal cases contains 1,634 nodes and 18,522 edges. We denote this graph as G_{grand}.

We can build Identity Ecosystem graphs in different sizes. For instance, the UTCID ITAP dataset includes identity theft and fraud cases from a wide range of market sectors impacting a wide range of victim demographics involving different types of PII with differing values and losses. We are able to filter for different case parameters (e.g. victim age, market sector, losses) to focus our analysis. For example, if we filter for cases with losses greater than $10,000, we will generate a smaller Identity Ecosystem graph with 761 nodes and 6,413 edges. We use G_{big_loss} to represent this filtered graph.

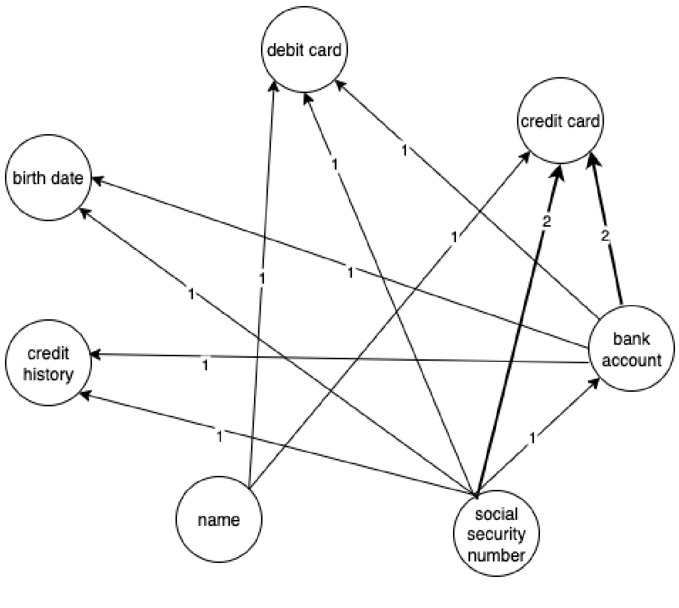


Fig. 4. An Example of Identity Ecosystem Graph Construction Using Three Cases.

IV. Link Prediction Algorithms

To answer a question like **Could the disclosure of a given identity attribute possibly leads to the disclosure of another identity attribute of concern**, we convert the question to the task of link prediction [15] and risk score calculation as Figure 1 shows. For link prediction task, we want to determine if a directed edge exists given an initial PII attribute (input) and a target PII attribute (output).

Suppose an Identity Ecosystem graph G_{TX} is constructed based on some identity theft and fraud cases that occurred in Texas. If an individual who lives in Texas was involved in a breach and the individual is informed their driver’s license number was disclosed, with the knowledge of G_{TX}, the link prediction algorithms can help this individual check to determine if there is a possible link (privacy risks) that may exist between their “driver’s license number” and “bank account” or a link between “driver license number” and “credit card”. In general, the link prediction algorithms can check if there are possible links between the initially exposed PII attributes to the remaining PII attributes in an Identity Ecosystem graph. An overview of link prediction algorithms is shown as Figure 9 [15] [16] in Appendix A.

Based on experimental results investigating ten benchmark homogeneous graphs (i.e. Ecoli [17], FB15K [18]), GNN-based link prediction algorithms show superiority over similarity-based algorithms [19]. Additional research shows that graph convolutional networks (GCN) and Word2Vec + Multi-layer Perceptron (MLP) have significant advantages regarding computation efficiency compared with Exponential Random graphs-based approaches for link prediction tasks while maintaining good prediction performance [20]. Moreover, since the nodes of an Identity Ecosystem graph are PII attributes and carry background information about individuals’ identity, a big concern for non-learning-based approaches is that it would be difficult to apply the context information with those methods. Given how large an Identity Ecosystem graph can be and how important it is to make accurate privacy risk predictions, creating an efficient and high-accuracy link prediction algorithm is necessary. We also need to take the link structure complexity into account for the link prediction algorithm. Therefore, we narrow the link prediction algorithm choices to GNN-based and general deep-learning based models which we highlight with red boxes in Figure 9. In the following subsections, we will discuss the features of the Identity Ecosystem graphs that we use to feed to the deep models. We will also explain the three link prediction models that we propose and create.

A. Semantic Processing for PII Attribute Nodes

The nodes of Identity Ecosystem graphs are composed of PII attributes. PII attributes, as we explained in the beginning of the paper, use natural languages to represent identity and privacy related information. Namely, the PII attributes are composed of words. In this paper, we only focus on the English words. We can find explanation for every word that exists in English with dictionaries. Hence, we define the “context information” of PII attributes as the word-by-word explanation of each attribute. We also call the process of converting the simple PII attributes to complex word-by-word explanation “semantic processing”. Our hypothesis is that: Similar to the inherent node properties of the Identity Ecosystem graphs (such as node degrees, node centrality and so on), the semantic information is also an important feature of nodes and can help us make predictions about whether or not a link between two nodes exists. We will use our proposed new model in Section 4.4 to verify our hypothesis.

The semantic processing toolkit we use is the Natural Language Processing Toolkit (NLTK) [21]. For each node in the Identity Ecosystem graphs, the processing procedures are: Get the words of the PII attribute; Iteratively process each word via NLTK $synsets$ and $definition$ functions to derive some explanation for each word, then, concatenate the explanation of all words of the PII attribute into one string of paragraph.

Furthermore, we use BERT uncased base model [22] to obtain the embedding of context information associated with each node. There are many embedding techniques. Since our focus is not about which embedding approach will give us better performance on PII attribute embedding, we will not test other embedding techniques in this paper. Our point is to utilize the context information of PII attributes and prove that the context information can improve the performance on link prediction algorithms. We define the embeddings of context information generated with the BERT encoder as the "semantic embeddings".

To explain the process of converting the PII attributes written in English words to the semantic embeddings, we will use a concrete example of the PII attribute "employee credential" (Figure 5). This example has two words: "employee" and "credential". As mentioned previously, the algorithm we create for processing each word employs NLTK synsets and definition functions. By applying this algorithm, the context information for "employee" is "a worker who is hired to perform a job". The context information for "credential" is "a document attesting to the truth of certain stated facts". Next, we concatenate the two strings. Therefore, we get the context information of "employee credential" being "a worker who is hired to perform a job a document attesting to the truth of certain stated facts". The context information is passed to the pretrained BERT uncased base model [22] and we can get the semantic embedding of "employee credential". Note that for context information associated with each PII attribute node, the corresponding semantic embedding might have different length. Since the median of the semantic embedding length for all nodes in G_{grand} is 124, we truncate the semantic embedding results for those with embedding length greater than 124 and apply zero paddings to those with embedding length is smaller than 124.

B. MLP Based Model for Link Prediction – featureMLP

For each node in a directed graph, we can calculate the basic node properties such as "in degree" and "out degree". These properties can be used to measure node similarity. Since it is possible to have a link between two nodes if they are similar [23] and there are powerful yet simple classifiers such as MLP [32] and support vector machine (SVM) [24], the first model we build is a MLP classifier with basic node properties according to paper [14]. Another reason for us to use MLP is to keep some consistency among the deep models we build for this paper. We call this MLP based link prediction model "featureMLP".

The overall model structure is shown as Figure 6. For the baseline models, the node features (the orange block in Figure 6) we use are:

- in degree, the number of incoming links for each node.
- out degree, the number of outgoing links for each node.
- betweenness centrality [25], a measure of how often a node lies in the shortest path of two nodes.
- closeness centrality [26], a measure of how close a node is to all other nodes in a graph.

FeatureMLP is computationally efficient and easy to implement. It uses the low-level node properties to make link predictions. Despite these advantages, we still need to explore the graph structural information to deal with the situation where the accuracy requirement is high. Therefore, we continue exploring and developing other link prediction models (Section IV-C and IV-D).

C. GCN Based Model for Link Prediction – featureGCN

Going forward from the first model (featureMLP), we want to find a way to leveraging the graph structural information. Therefore, we change the deep model from MLP to a 2-layer GCN [21].

The model structure is shown as Figure 7. We use two SAGEConv layers (based on GraphSage [27]) with Pytorch geometric package [28] to generate node embeddings that contain local graph structural information. We pair the node embeddings to get the embeddings for the graph edges. For each edge embedding, there are two node embeddings (ne_1, ne_2). We use element-wise multiplication to get the aggregation result for each edge embedding $(ne_1 * ne_2)$. The final aggregation result is denoted as $A1$ in Figure 7.

featureGCN leverages the low-level node properties and the graph structural information at the same time. Therefore, the prediction made by featureGCN is more reliable. However, as we mentioned earlier, the semantic information of the PII attribute also contain useful information. Therefore, a novel GCN model is needed to use the semantic information.

D. GCN with Semantic Embeddings for Link Prediction - SeeGCN

As discussed in Section IV-A, we can get the semantic embeddings for each node of the input graphs. Starting from the model featureGCN, we need to incorporate and leverage the semantic embeddings into our framework and show that the semantic embeddings do help to improve the link prediction accuracy. The new model structure is shown as Figure 8, and we name our new model SeeGCN.

For each pair of nodes (n_1, n_2) from the edges of the input graphs, we can get the pair of node semantic embeddings (se_1, se_2). We call the pair of node semantic embeddings "edge semantic embedding" of the input graphs. Next, we aggregate se_1 and se_2 by concatenation and fully connected layers. The final aggregation result is denoted as "$A2$" in

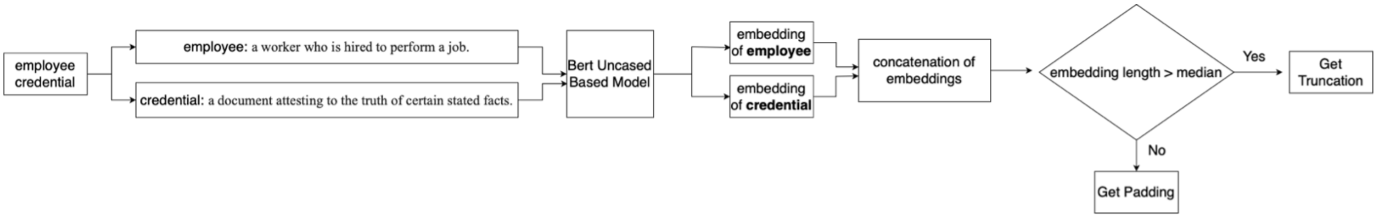

employee credential
employee: a worker who is hired to perform a job.
credential: a document attesting to the truth of certain stated facts.
Bert Uncased Based Model
embedding of employee
embedding of credential
concatenation of embeddings
embedding length > median
Yes
Get Truncation
No
Get Padding

Fig. 5. The Process of Converting PII Attributes from English Words to the Semantic Embeddings.

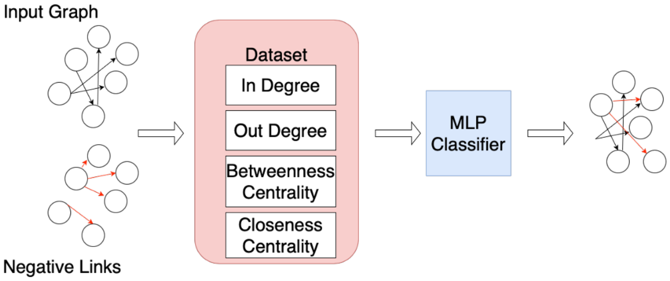


Fig. 6. Overview of the Model Structure - featureMLP.

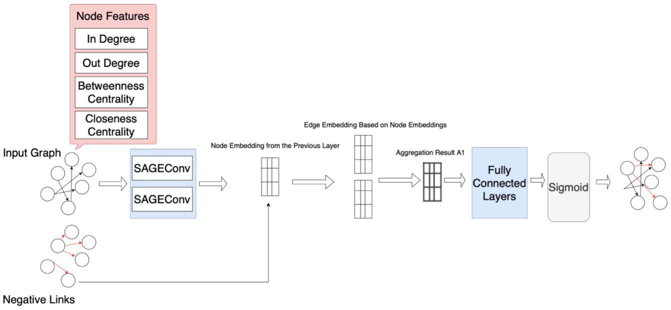


Fig. 7. Overview of the Model Structure - featureGCN.

Figure 8. The "$A1$" in Figure 8 is the same as the one in Figure 7. We concatenate $A1$ and $A2$ to generate the final aggregation result $A3$ in Figure 8.

This new link prediction model effectively integrates low-level node properties, graph structural information, and semantic information. Moreover, built from featureGCN, this model is not too complicated to implement.

V. Result Analysis

To compare the performance between proposed models and the baseline models, we test the models on the G_{grand} and G_{big_loss} graphs which are explained in Section III. We also randomly sample the collected identity theft and fraud cases (data from the UTCID ITAP dataset) and build graphs according to the random samples. We name the graphs $G_{(\#nodes,\#edges)}$, where $\#nodes$ indicates the number of nodes in the graph and $\#edges$ indicates the total edge number of the graph. Graphs are constructed by networkX [29] Python library and converted to Pytorch Geometric data with the function $from\_networkX$ in Pytorch Geometric (PyG) library [28]. The number of training epochs is set to 50 and the learning rate is set to 0.01. For each graph, we randomly split its edges into training and validation sets using the $RandomLinkSplit$ function from the $transformer$ package of PyG library. The split ratio is (training:validation = 9:1). The performance of the models is shown as Table I.

graph	**featureMLP**	**featureGCN**	**seeGCN**
G_{grand}	0.72	0.84	**0.85**
G_{big_loss}	0.54	0.82	**0.85**
$G_{(1109,9265)}$	**0.84**	0.67	0.83
$G_{(451,2054)}$	0.76	**0.83**	0.79
$G_{(527,2676)}$	0.80	**0.82**	0.81
$G_{(556,3142)}$	0.79	**0.81**	**0.81**
$G_{(1334,12351)}$	0.72	0.84	**0.86**
$G_{(1509,15492)}$	0.68	0.81	**0.84**
$G_{(1498,15072)}$	0.50	0.81	**0.84**
$G_{(1312,12109)}$	0.73	0.68	**0.86**

TABLE I
Model Validation Accuracy on Different Identity Ecosystem Graphs.

The table shows the performance of different models with the best validation accuracy achieved. All models achieve good accuracy (above 0.7), except that featureMLP got an accuracy of 0.54 on G_{big_loss}. It is because the node properties of G_{big_loss} is not sufficient to make link predictions. The highest validation accuracy for each testing graph is bolded.

Overall, it is clear that the link connections of Identity Ecosystem graphs can be properly predicted with our proposed models above. Moreover, the semantic embeddings of PII attributes can help improve the link prediction performance especially when the graphs are large.

VI. Risk Score Calculation

For the same PII attribute, people may value it differently. Some people may spend most of their time on social media. Some may only check social media occasionally. It is obvious that people from the former group will assign higher scores to usernames and passwords associated with their online accounts. After we get the possible disclosed nodes based on the link prediction results from Section IV, we can either assign risk scores manually based on our personal preference or use some evaluation metrics to help us predict the final scores.

We propose a risk score calculation technique here. First, suppose the query is to check the risk scores of nodes related to the disclosure of Attribute α_1. Suppose PII attribute set which is all nodes in the Identity Ecosystem graph except for

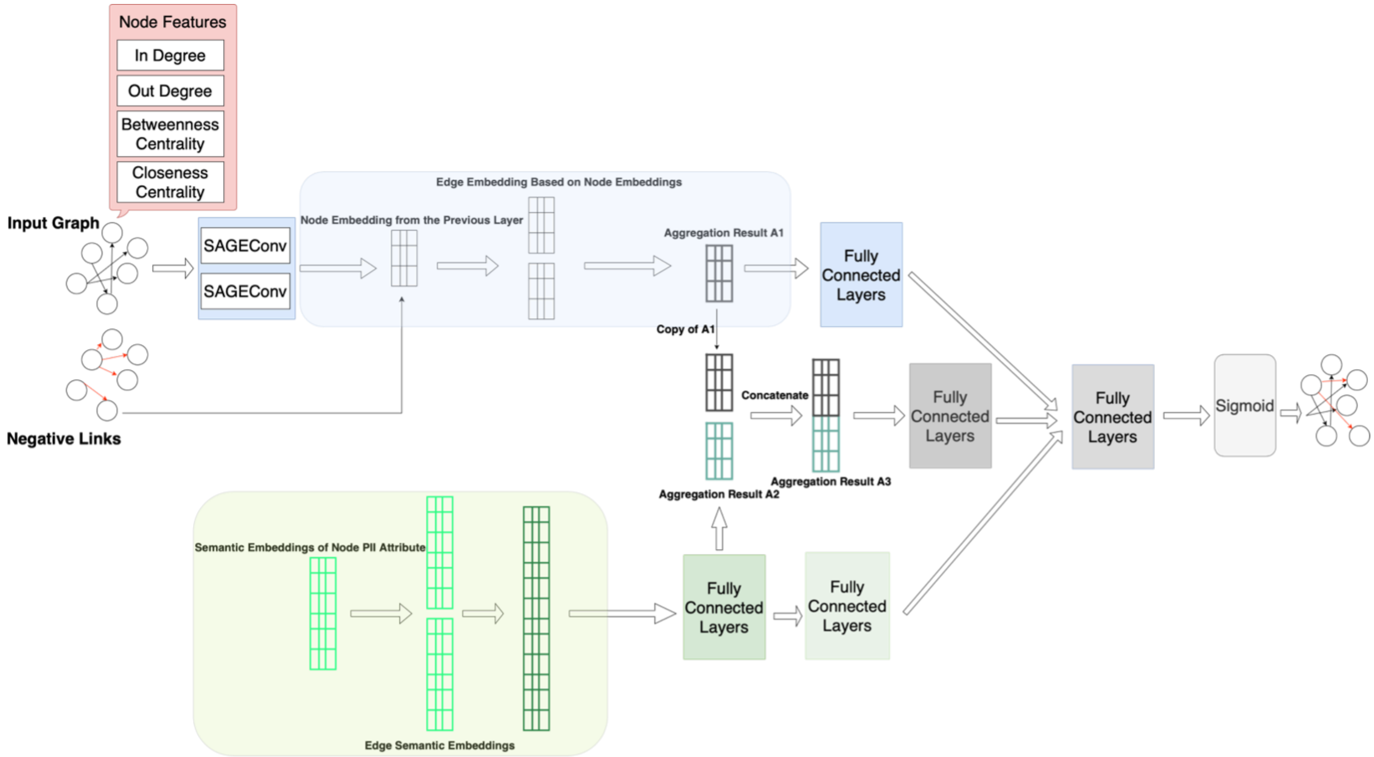


Fig. 8. Overview of the Model Structure - seeGCN.

α_1 is $\{n_1, n_2, n_3, ...\}$ where we denote each node in the set as n_i.

We can run the PageRank algorithm [30] on the graph of interest and obtain the PageRank coefficients, denoted as pr_i for each node n_i. Second, we can also calculate the reverse PageRank coefficient for each node by running the reverse PageRank algorithm on the graph [31], denoted as rpr_i for each node n_i. The score S_i is then defined as the sum of its forward and reverse PageRank coefficients: $S_i = pr_i + rpr_i$.

Next, for each pair of (α_1, n_i), the link prediction algorithm will output the link existence probability p_i. The final risk score of n_i in the event where A1 is disclosed is $RS_i = p_i * S_i$.

To convert the scores of a PII attribute node RS_i to a $[0, 100]$ scale, we can normalize and scale RS_i by the equation $RS_i = RS_i/maxS * 100$, where $maxS$ represents the maximum of RS_i among all nodes.

VII. Conclusion

The goal of this paper is to analyze and predict the privacy risks incurred when an individual shares their personal data. In this paper, we introduce MLP-based and GCN-based algorithms (featureMLP, featureGCN, seeGCN) to answer the question **Can the disclosure of a given identity attribute possibly lead to the disclosure of another attribute**? This research posits that an individual who better understands the risks of sharing respective personal data (identity attributes) will be better equipped to protect the respective data. Experimental analysis of the link prediction algorithms on different Identity Ecosystem graphs show that we could answer the proposed question well by using GCN-based algorithms and embedding the context information of PII attributes. The Identity Ecosystem graphs and the proposed privacy risk prediction framework provides both flexibility and customization.

Our future work is to search and find the optimal GCN structure based on different graph sizes, explore the direction of combining GNN and reinforcement learning, and research better methodologies to integrate the PII attribute semantic embeddings into the GCN-based models.

Appendix A
Link Prediction Methods Overview

Figure 9 shows an overview of link prediction algorithms.

References

[1] R. N. Zaeem, S. Budalakoti, K. S. Barber, M. Rasheed, and C. Bajaj, “Predicting and explaining identity risk, exposure and cost using the ecosystem of identity attributes,” in *2016 IEEE International Carnahan Conference on Security Technology (ICCST)*. IEEE, 2016, pp. 1–8.

[2] A. F. Karr and J. P. Reiter, “Using statistics to protect privacy,” *Privacy Big Data, and the Public Good: Frameworks for Engagement*, 2014.

[3] L. Zeng, Z. Shi, S. Xu, and D. Feng, “Safevanish: An improved data self-destruction for protecting data privacy,” in *2010 IEEE Second International Conference on Cloud Computing Technology and Science*. IEEE, 2010, pp. 521–528.

[4] R. Geambasu, T. Kohno, A. A. Levy, and H. M. Levy, “Vanish: Increasing data privacy with self-destructing data.” in *USENIX security symposium*, vol. 316, 2009, pp. 10–5555.

[5] J. Zaiss, R. Nokhbeh Zaeem, and K. S. Barber, “Identity threat assessment and prediction,” *Journal of Consumer Affairs*, vol. 53, no. 1, pp. 58–70, 2019.

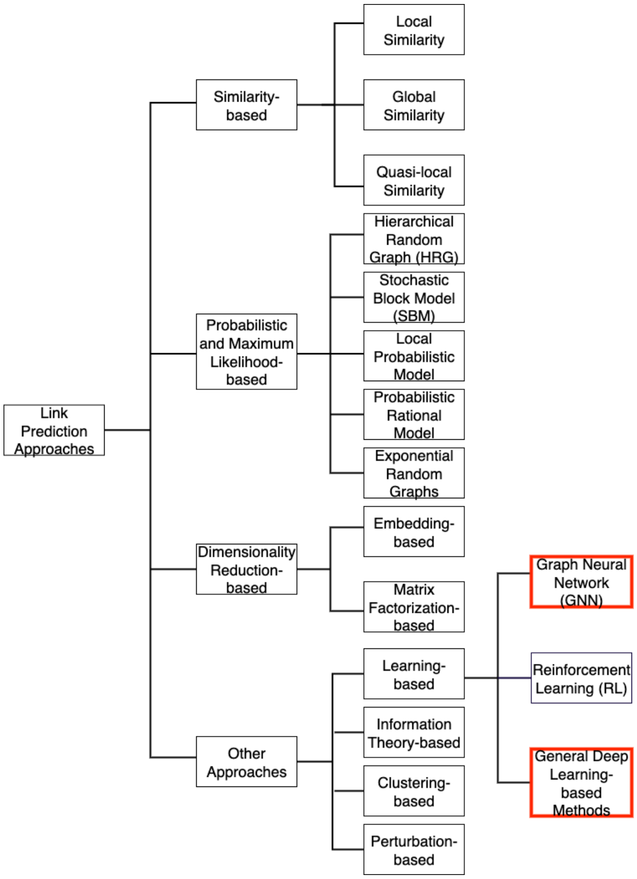

Link Prediction Approaches
Similarity-based
Local Similarity
Global Similarity
Quasi-local Similarity
Probabilistic and Maximum Likelihood-based
Hierarchical Random Graph (HRG)
Stochastic Block Model (SBM)
Local Probabilistic Model
Probabilistic Rational Model
Exponential Random Graphs
Dimensionality Reduction-based
Embedding-based
Matrix Factorization-based
Other Approaches
Learning-based
Information Theory-based
Clustering-based
Perturbation-based
Graph Neural Network (GNN)
Reinforcement Learning (RL)
General Deep Learning-based Methods

Fig. 9. Overview of Link Prediction Algorithms.

[6] A. Nippani, D. Li, H. Ju, H. Koutsopoulos, and H. Zhang, “Graph neural networks for road safety modeling: Datasets and evaluations for accident analysis,” *Advances in neural information processing systems*, vol. 36, pp. 52 009–52 032, 2023.

[7] D. Ediger, K. Jiang, J. Riedy, D. A. Bader, C. Corley, R. Farber, and W. N. Reynolds, “Massive social network analysis: Mining twitter for social good,” in *2010 39th international conference on parallel processing*. IEEE, 2010, pp. 583–593.

[8] H. A. Javaid, “Improving fraud detection and risk assessment in financial service using predictive analytics and data mining,” *Integrated Journal of Science and Technology*, vol. 1, no. 8, pp. 1–11, 2024.

[9] X. Zhang and T. Guo, “Privacy risk assessment of medical big data based on information entropy and fcm algorithm,” *IEEE Access*, 2024.

[10] A. H. Murphy, “The finley affair: A signal event in the history of forecast verification,” *Weather and forecasting*, vol. 11, no. 1, pp. 3–20, 1996.

[11] L. A. Adamic and E. Adar, “Friends and neighbors on the web,” *Social networks*, vol. 25, no. 3, pp. 211–230, 2003.

[12] C. Wang, V. Satuluri, and S. Parthasarathy, “Local probabilistic models for link prediction,” in *Seventh IEEE international conference on data mining (ICDM 2007)*. IEEE, 2007, pp. 322–331.

[13] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla, “New perspectives and methods in link prediction,” in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 243–252.

[14] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki, “Link prediction using supervised learning,” in *SDM06: workshop on link analysis, counter-terrorism and security*, vol. 30, 2006, pp. 798–805.

[15] A. Kumar, S. S. Singh, K. Singh, and B. Biswas, “Link prediction techniques, applications, and performance: A survey,” *Physica A: Statistical Mechanics and its Applications*, vol. 553, p. 124289, 2020.

[16] D. Arrar, N. Kamel, and A. Lakhfif, “A comprehensive survey of link prediction methods,” *The journal of supercomputing*, vol. 80, no. 3, pp. 3902–3942, 2024.

[17] H. Salgado, A. Santos-Zavaleta, S. Gama-Castro, D. Millán-Zárate, E. Díaz-Peredo, F. Sánchez-Solano, E. Pérez-Rueda, C. Bonavides-Martínez, and J. Collado-Vides, “Regulondb (version 3.2): transcriptional regulation and operon organization in escherichia coli k-12,” *Nucleic acids research*, vol. 29, no. 1, pp. 72–74, 2001.

[18] M. E. Newman, “Finding community structure in networks using the eigenvectors of matrices,” *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, vol. 74, no. 3, p. 036104, 2006.

[19] M. K. Islam, S. Aridhi, and M. Smail-Tabbone, “A comparative study of similarity-based and gnn-based link prediction approaches,” *arXiv preprint arXiv:2008.08879*, 2020.

[20] J. Sosa, D. Martínez, and N. Guerrero, “An unified approach to link prediction in collaboration networks,” *arXiv preprint arXiv:2411.01066*, 2024.

[21] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. ” O’Reilly Media, Inc.”, 2009.

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[23] A. F. Al Musawi, S. Roy, and P. Ghosh, “Identifying accurate link predictors based on assortativity of complex networks,” *Scientific Reports*, vol. 12, no. 1, p. 18107, 2022.

[24] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, pp. 273–297, 1995.

[25] L. C. Freeman, “A set of measures of centrality based on betweenness,” *Sociometry*, pp. 35–41, 1977.

[26] G. Sabidussi, “The centrality index of a graph,” *Psychometrika*, vol. 31, no. 4, pp. 581–603, 1966.

[27] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” *Advances in neural information processing systems*, vol. 30, 2017.

[28] M. Fey and J. E. Lenssen, “Fast graph representation learning with pytorch geometric,” *arXiv preprint arXiv:1903.02428*, 2019.

[29] A. Hagberg, P. J. Swart, and D. A. Schult, “Exploring network structure, dynamics, and function using networkx,” Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), Tech. Rep., 2008.

[30] L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web.” Stanford infolab, Tech. Rep., 1999.

[31] Z. Bar-Yossef and L.-T. Mashiach, “Local approximation of pagerank and reverse pagerank,” in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008, pp. 279–288.